A DUAL ALGORITHM FOR APPROXIMATING PARETO SETS IN CONVEX MULTI-CRITERIA OPTIMIZATION

Rasmus BOKRANTZ*†‡ and Anders FORSGREN†

Technical Report TRITA-MAT-2011-OS3
Department of Mathematics
Royal Institute of Technology
February 2011

Abstract

We consider the problem of approximating the Pareto set of convex multi-criteria optimization problems by a discrete set of points and their convex combinations. Finding the scalarization parameters that maximize the improvement in bound on the approximation error when generating a single Pareto optimal solution is a nonconvex optimization problem. This problem is solvable by enumerative techniques, but at a cost that increases exponentially with the number of objectives. The goal of this paper is to present a practical algorithm for solving the Pareto set approximation problem in presence of up to about ten conflicting objectives, motivated by application to radiation therapy optimization. To this end, an enumerative scheme is proposed that is in a sense dual to the algorithms in the literature. The proposed technique retains the quality of output of the best previous algorithm while solving fewer subproblems. A further improvement is provided by a procedure for discarding subproblems based on reusing information from previous solves. The combined effect of the proposed enhancements is empirically demonstrated to reduce the computational expense of solving the Pareto surface approximation problem by orders of magnitude.

1. Introduction

Multi-criteria optimization (MCO) deals with optimization problems involving multiple mutually conflicting objectives, see, e.g., the monographs [5, 20, 36]. For such problems in general, there is no feasible solution that is optimal with respect to all objectives simultaneously. Instead, a well-balanced trade-off between objectives is sought within the Pareto optimal set: the set encompassed by the feasible solutions such that an improvement in one objective can only be achieved through a sacrifice in another. One approach to identifying a suitable trade-off between objectives is to pre-compute a finite number of solutions without human interaction, and then take these as input to a navigation tool that allows feasible trade-offs to be evaluated in real-time by forming convex combinations between discrete solutions, see [16,22,38].

*E-mail: bokrantz@kth.se

†Optimization and Systems Theory, Department of Mathematics, Royal Institute of Technology (KTH), SE-100 44 Stockholm, Sweden.

‡RaySearch Laboratories, Sveavägen 25, SE-111 34 Stockholm, Sweden.

This technique is particularly suited for large-scale problems where a single optimization run is costly, and by design, imposes a restriction to problem formulations with convex constraints.

We are particularly interested in application of the described technique to treatment planning for intensity modulated radiation therapy (IMRT). There is a rich literature of methods that recognize IMRT planning as an MCO problem, see, e.g., [9, 12–14, 31, 32]. Clinical evaluations have demonstrated that such methods have the potential of improving both manual planning time and treatment quality [15, 26, 51]. In view of this application, we limit ourselves to generating Pareto optimal points by replacing the vector-valued objective function of the initial problem with a convex combination of its components. This weighting method is extensively used throughout the field of MCO [34] and is the de facto standard method for IMRT optimization [1, 28]. We further limit ourselves to consider convex objectives, so that the Pareto optimal set forms a connected surface in the boundary of a convex set [44]. Convex criteria that are commonly used in IMRT planning together with nonconvex criteria that can be reformulated into convex such are surveyed in [44].

Within this context, we consider the problem of generating a discrete set of Pareto optimal solutions so that the lower boundary of their convex hull provides a representative approximation of the Pareto surface. We focus on so-called sandwich algorithms, which maintain inner and outer approximations of the Pareto surface, see [7, 23, 45] for algorithms in the plane and [13, 30, 42, 48] for algorithms in general dimensions. The inner and outer approximations are used to steer generation of new points towards parts of the Pareto surface that currently lack accurate representation, and to provide an approximation guarantee for the current discrete representation.

The computational expense of a sandwich algorithm increases exponentially with the number of objectives. This behavior is fundamentally due to what is called "the curse of dimensionality": in direct sampling of a distribution of data, the number of samples required to maintain a given level of accuracy increases exponentially with the number of variables [2]. As a consequence, application of sandwich algorithms to problems with more than six objectives has, to the best of our knowledge, previously not been reported. The number of objectives commonly encountered in IMRT planning, on the other hand, range up to about ten [11, 49]. Current practice for high-dimensional cases is to sample weights uniformly at random. This technique is well-known to be inadequate for generating an even distribution of points from all parts of the Pareto surface [17].

Motivated by these shortcomings, we develop methods for making sandwich algorithms tractable to a wider range of problem formulations. We devote particular attention to the mathematical programming and computational geometry aspects of this problem. For many problems within these two fields, primal and dual formulations have equivalent complexity in the general case. However, as noted by Bremner [6, p. 2] with respect to polytope duality: "for a particular class of polytopes and a fixed algorithm, one transformation may be much easier than its dual." Arguing that this is the case for sandwich algorithms, we give an algorithm that in a sense is dual to algorithms in the literature. Our main contribution is a scheme that

retains the quality of output of the best previous algorithm while achieving a more benign ratio between computational effort and problem dimension. The presented algorithm also generalizes sandwich algorithms to be compatible with cone-based preference models, see e.g., [21,27,37]. For the IMRT application, such models have proven useful for excluding parts of the Pareto surface that are known a priori to not be of interest, see [47].

2. Preliminaries

2.1. Notation and terminology

We denote by e the vector of ones with dimension defined by the context. Vector inequalities are to be understood componentwise. We treat sets of points and matrices interchangeably when convenient; the rows of the matrix are the elements of the corresponding set. The shorthand $(\cdot)_+$ is used to denote $\max\{\cdot, 0\}$. We denote the optimal value of an optimization problem P by $\text{optval}(P)$. For a function f and subset S of its domain, we denote by $f(S)$ the image $\{f(s) : \ s \in S\}$. For a set S, we denote by $\text{conv}(S)$ its convex hull. For two sets S_1 and S_2, we denote by $S_1 + S_2$ their Minkowski sum. Minkowski addition between a set S and a singleton set $\{s\}$ is denoted by $s + S$. A hyperplane $\left\{z : \ a^T z = b\right\}$ with nonzero normal a and offset b is denoted by $H(a, b)$. With each hyperplane, we associate a closed positive, a closed negative, an open positive and an open negative halfspace, defined by substituting respectively "$\geq$," "$\leq$," "$>$," and "$<$" for the equality in the hyperplane equation. The intersection of a finite number of closed halfspaces is called a polyhedron. A closed and bounded polyhedron is called a polytope. The k-dimensional intersection between a polyhedron and one of its supporting hyperplanes is called a k-face. A 0-face is called a vertex, a 1-face an edge, an $(n-2)$-face a ridge, and an $(n-1)$-face a facet. Unless the contrary is stated, a normal vector to a polyhedral face is assumed to be oriented inwards.

2.2. Problem formulation

The algorithm to be described applies to a multi-objective optimization problems on the form

$$\text{(MOP)} \qquad \begin{array}{ll} \underset{x}{\text{minimize}} & (f_1(x), \ldots, f_n(x))^T \\ \text{subject to} & x \in X = \{x : \ c(x) \leq 0\}, \end{array} \tag{2.1}$$

involving $n \geq 2$ objective functions f_i to be minimized over a feasible region $X \subseteq \mathbb{R}^m$ defined by a vector c of constraint functions. We denote by f the n-vector of objective functions and by Z the image of the feasible region under the objective function mapping, i.e., $Z = f(X)$. We refer to the m-dimensional space of which X is a subset as the decision space and to the n-dimensional space of which Z is a subset as the objective space. Throughout, the feasible region X is assumed to be nonempty and the functions f and c to be convex and bounded on X. The feasible region is a convex set by virtue of that all sublevel sets of convex functions are convex. Since f and X are both convex, MOP is a convex optimization problem.

2.3. Notion of optimality

The solution set to MOP is the set of nondominated feasible points. Dominance relations between points in objective space are defined with respect to the partial order induced by some ordering cone C which we require to be closed, pointed (i.e., $0 \in C$), salient (i.e., $C \cap -C \subseteq \{0\}$), convex, and containing $\mathbb{R}^n_+$.

Definition 2.1. (Nondominance) Let x^* be feasible to MOP. Then, x^* is nondominated if there exists no x in X such that $f(x^*) \in f(x) + C \setminus \{0\}$.

In order to distinguish between the decision space and the objective space more easily, we refer to a nondominated solution x^* as efficient whereas the corresponding objective vector $f(x^*)$ is called Pareto optimal. We refer to the set of all efficient solutions as the efficient set and to the set of all Pareto optimal objective vectors as the Pareto set.

We restrict ourselves to consider polyhedral ordering cones generated by some matrix Q, i.e., $C = \{Q\mu : \ \mu \geq 0\}$. Instead of specifying Q directly, we prefer to define the set of admissible trade-off rates between objectives and let this set be the dual cone $C^* = \{z : \ y^T z \geq 0 \ \forall y \in C\}$ to C. Let T be the symmetric $n \times n$ matrix with unit diagonal and nonnegative off-diagonal elements t_{ij} such that the reciprocal of t_{ij} is the maximum acceptable increase in f_i for a unit decrease in f_j. Then, C^* is the polyhedral cone generated by T, and C the dual cone to C^*, i.e., $C = \{z : \ Tz \geq 0\}$, here using that $C = C^{**}$ by convexity and closedness of C [50, p. 53]. By construction, $C^* \subseteq \mathbb{R}^n_+$, so that $\mathbb{R}^n_+ \subseteq C$ [50, p. 56]. Taking T and Q to be the identity matrix, so that $C = C^* = \mathbb{R}^n_+$, gives dominance in the conventional Pareto sense.

2.4. The weighting method

A Pareto optimal solution can be computed by introducing an n-vector w of weights such that $w \in C^*$ and solving the scalar-valued optimization problem

$$\text{(SUM}(w)) \qquad \begin{array}{ll} \underset{x}{\text{minimize}} & \sum_{i=1}^n w_i f_i(x) \\ \text{subject to} & x \in X. \end{array} \tag{2.2}$$

This is a convex optimization problem by virtue of that $C^* \subseteq \mathbb{R}^n_+$ and since nonnegative linear combinations preserve convexity. The vector w is throughout assumed to be normalized so that $e^T w = 1$.

Problems MOP and SUM(w) are related as follows. If a point x^* is optimal to SUM(w) for some w in C^* such that $w > 0$, then x^* is efficient to MOP [36, Thm. 3.1.2]. For any x^* that is efficient to MOP, there exists w in C^* such that x^* is optimal to SUM(w) [36, Thm. 3.1.4]. The second of these results relies on convexity of MOP. As an immediate consequence, if x^* is optimal to SUM(w) for some w in C^*, the hyperplane $H(w, f(x^*))$ supports the feasible objective space Z at $f(x^*)$. To see this, observe that $f(x^*) \in Z \cap H(w, f(x^*))$ and that the intersection between Z and the open negative halfspace associated with $H(w, f(x^*))$ is empty, or otherwise x^* would not be optimal to SUM(w). Finding any point on the Pareto surface thus reduces to solving SUM(w) with w normal to the Pareto surface at the sought point.

3. The sandwich algorithm

3.1. The algorithmic idea

A generic sandwich algorithm is given in Algorithm 3.1. The goal of this algorithm is to generate a set of points such that their convex hull constitutes an approximation of the efficient set with approximation error below some tolerance $\varepsilon > 0$, in as few solves as possible. The algorithm avoids assessing the quality of the approximation of the efficient set in the typically high-dimensional decision space by a mapping to the objective space and evaluating the resulting image with respect to the Pareto surface. An upper bound on the approximation error is calculated as the distance between polyhedral inner and outer approximations of the Pareto surface. The weighting vector in the next weighted-sum problem to be solved is taken to be normal to the inner approximation at the point where the upper bound is attained. This choice corresponds to the greedy strategy of maximizing the decrease in bound on the approximation error at each iteration.

The first two steps in Algorithm 3.1 serve to normalize the range of each objective function, as to avoid bias towards objectives with a large order of magnitude. We here use the pragmatic approach of normalizing each objective function with respect to its minimum and maximum objective value during the n initial solves of SUM(w). We refer to an iteration of the while loop in Algorithm 3.1 as an iteration of the sandwich algorithm.

Algorithm 3.1. *The sandwich algorithm*

for $i = 1, \ldots, n$ **do** solve SUM(t_i) with respect to the ith extreme ray t_i of C^*;
normalize f to $[0, 1]^n$;
solve SUM(w) with $w = \frac{1}{n}e$;
construct inner and outer approximations Z_{in} and Z_{out} of the Pareto surface;
while not converged **do**
    compute an upper bound on the approximation error;
    if the upper bound is below ε **then** converged; **break**;
    solve SUM(w) with w normal to Z_{in} where the upper bound is attained;
    update Z_{in} and Z_{out};
end

3.2. Overview of previous general-dimensional sandwich algorithms

Three algorithms are of immediate interest with respect to the algorithm we propose: those of Solanki et al. [48], Craft et al. [13] and Rennen et al. [42], where the two latter algorithms can be viewed as enhanced versions of the proceeding one(s). Any general-dimensional sandwich algorithm must handle the fact that a vector normal

to the convex hull of a discrete set of Pareto optimal points can have negative components in dimension beyond two. Hence, if such a normal vector is used directly as weighting vector w in solving SUM(w), this vector may not lie in C^*, and the resulting optimal solution may not be efficient to MOP. The algorithm of Solanki et al. handles this complication by introducing bounds on the allowed deviation from the Pareto surface. In the algorithm of Craft et al., a heuristic method is instead used for transforming mixed normals into nonnegative such, as to make better use of each optimization run. The algorithm of Rennen et al. avoids mixed normals altogether by augmenting the convex hull representation of the inner approximation through setwise summation with the nonnegative orthant. Any normal to the resulting polyhedron is nonnegative [42, Lemma 2]. In [42], the three algorithms are empirically evaluated on a suite of test problems. This study indicates that the algorithm of Rennen et al. generates well-distributed points on the Pareto surface and provide a corresponding rapid improvement in bound on the approximation error, whereas this is generally not the case for the algorithms of Solanki et al. and Craft et al. Based on these findings, we use the algorithm of Rennen et al. as a single benchmark to the algorithm proposed in this paper, both in the theoretical exposition and the numerical experiments.

In the remainder of this section, we define polyhedral approximations of the Pareto surface and a quality measure on a discrete representation of the Pareto set. These definitions generalize those of Rennen et al. to dominance induced by a general ordering cone. A novel approach for performing the steps encompassing an iteration of the sandwich algorithm is proposed in Section 4.

3.3. Polyhedral approximations

The key result that makes construction of polyhedral approximations of the Pareto surface possible is that the set $Z_+ = Z + C$ is a convex set whenever MOP is a convex problem. This result is a generalization of the result that a function is convex if and only if its epigraph is a convex set to the case of epigraphs induced by convex cones [40]. Other proofs of this result for the case $C = \mathbb{R}^n_+$ can be found in [10, 44]. Convexity of Z_+ implies that the convex hull of any discrete set of points in Z_+ is an inner approximation of this set, and that the intersection of any set of closed positive halfspaces associated with supporting hyperplanes to Z_+ an outer approximation. In particular, polyhedral approximations of Z_+ can be constructed as follows.

Definition 3.1. (Inner and outer approximations) Let D be a discrete set of points that are efficient to MOP and optimal to SUM(w) with respect to some set W of weighting vectors in C^*. Then, $Z_{\text{in}} = \{P^T\lambda + Q^T\mu : \lambda, \mu \geq 0, e^T\lambda = 1\}$ where $P = f(D)$ is an inner approximation of Z_+, and $Z_{\text{out}} = \{z : Wz \geq r\}$ where r is the vector of pairwise scalar products between elements in P and W an outer approximation of Z_+, in the sense that $Z_{\text{in}} \subseteq Z_+ \subseteq Z_{\text{out}}$.

Shrinking the hypervolume between Z_{in} and Z_{out} provide a means of approximating the Pareto surface with increased accuracy as this set must confine to the lower boundary of Z_+.

3.4. Quantifying the approximation error

Quality measures of discrete representations of the Pareto set have been reviewed in [46]. The definition we give quantifies the coverage of the Pareto set in terms of a relaxation of the nondomincance criterion.

Definition 3.2. (Approximation error) Let D be a discrete set of feasible points to MOP. Then, the approximation error of D is the minimum ε such that for any efficient x^*, there exists x in $\mathrm{conv}(D)$ such that $f(x^*) \in f(x) + (C - \varepsilon e)$.

Unfortunately, explicit knowledge of the Pareto set is required to compute the above defined quantity. Since the entire Pareto set in general is not known, we will exclusively work with the upper bound on the approximation error provided by the minimum ε such that $Z_{\text{out}} \subseteq (Z_{\text{in}} - \varepsilon e)$. This upper bound is equivalent to the Hausdorff distance

$$h(Z_{\text{in}}, Z_{\text{out}}) = \max_{z \in Z_{\text{out}}} \min_{z' \in Z_{\text{in}}} d(z, z'),$$

taken with respect to the one-sided distance function

$$d(z, z') = \max_{i \in \{1,\ldots,n\}} (z'_i - z_i)_+.$$

Computing $h(Z_{\text{in}}, Z_{\text{out}})$ requires solving the linear bilevel program

$$\underset{z}{\text{maximize}} \quad \left\{ \begin{array}{ll} \underset{\eta,\ \lambda,\ \mu}{\text{minimize}} & \eta \\ \text{subject to} & \eta e \geq P^T\lambda + Q^T\mu - z, \\ & e^T\lambda = 1, \\ & \eta,\ \lambda,\ \mu \geq 0, \end{array} \right\} \tag{3.1}$$

$$\text{subject to} \quad Wz \geq r. \tag{3.2}$$

Linear bilevel problems have been shown to be NP-hard and inapproximable within any constant factor in polynomial time [18, Thm. 3.12]. Problems within this class may be exactly solvable by enumerative techniques in moderate problem dimensions, see, e.g., [39], whereas finding the optimum to large-scale instances is in general not tractable.

4. The vertex enumerative algorithm

4.1. Solution by enumerating the vertices of the outer approximation

We propose to solve (3.1) by enumerating the extreme points of its feasible region and solving a linear programming subproblem for each extreme point found. The extreme points of the outer approximation are its finitely many vertices by virtue of that this set is polyhedral. Validity of the proposed method relies on the following results.

Proposition 4.1. *At least one vertex of the outer approximation is an optimal solution to (3.1).*

Proof. The inner level linear program in (3.1) can be put on standard form by introducing slack variables for each inequality constraint. Then, since the optimal value of a linear program on standard form is a convex, piecewise affine and continuous function of its right hand side coefficients [33, Lemma 1–2], problem (3.1) by a change of sign in the objective function amounts to minimizing a concave function over a convex set. Since every global and local minimum value of a concave function is either attained at an extreme point of its feasible domain or the function is unbounded from below on a feasible ray [43, Thm. 32.3], the proof reduces to showing that the objective value of (3.1) is nonincreasing on any ray in Z_{out}. Let z and p be vectors such that $\{z + \alpha p : \alpha \geq 0\}$ is a ray in Z_{out}, i.e., $z \in Z_{\text{out}}$, $p \neq 0$, and $Wp \geq 0$. Let also (η, λ, μ) denote an optimal solution to the inner level linear program in (3.1) with respect to z. Since every row vector in W lies in C^*, we have that $p \in C$, and therefore, there exists $\bar{\mu} \geq 0$ such that $p = Q\bar{\mu}$. For such $\bar{\mu}$, $(\eta, \lambda, \mu + \alpha\bar{\mu})$ is feasible to the inner level linear program in (3.1) with respect to $z + \alpha p$ with objective value η. The objective value of (3.1) in any point on the ray is thus bounded from above by the objective value in the point z from which the ray emanates, and the proof is complete. ∎

As a direct consequence of this results, the optimal value of (3.1) is given by $\max_{v \in V} \text{optval}(\text{PLP}(v))$, where V denotes the set of vertices of the outer approximation and where

$$
\begin{aligned}
& & \underset{\eta,\ \lambda,\ \mu}{\text{minimize}} \quad & \eta && \text{(4.1a)}\\
&(\text{PLP}(v)) & \text{subject to} \quad & \eta e \geq P^T\lambda + Q^T\mu - v, && \text{(4.1b)}\\
& & & e^T\lambda = 1, && \text{(4.1c)}\\
& & & \eta,\ \lambda,\ \mu \geq 0. && \text{(4.1d)}
\end{aligned}
$$

We will for the moment postpone how to enumerate the vertices of Z_{out} and instead discuss some properties of this problem.

4.2. Identifying the next weighting vector

Having solved all instances of (4.1), we turn to identifying the weighting vector of the next weighted-sum problem that is to be solved. Similar lines of reasoning have previously been applied to a related problem in [10]. The linear programming dual to PLP(v) takes the form

$$
\begin{aligned}
& & \underset{\pi,\ \rho}{\text{maximize}} \quad & \rho - v^T\pi && \text{(4.2a)}\\
&(\text{DLP}(v)) & \text{subject to} \quad & P\pi \geq \rho e, && \text{(4.2b)}\\
& & & Q\pi \geq 0, && \text{(4.2c)}\\
& & & e^T\pi \leq 1, && \text{(4.2d)}\\
& & & \pi \geq 0. && \text{(4.2e)}
\end{aligned}
$$

It is straightforward to verify that PLP(v) is feasible and its objective value bounded from below. Therefore, by linear programming duality, DLP(v) is feasible and its

objective value bounded from above. Moreover, for any primal-dual optimal solution $(\eta, \lambda, \mu, \pi, \rho)$ to PLP(v) and DLP(v) associated with some v in V such that optval(PLP(v)) > 0, π lies in C^* and is normal to the inner approximation at $y = P^T\lambda + Q^T\mu$. This claim is made precise in Proposition 4.2. The next weighting vector is thus given by the vector of optimal dual variables π to the instance of DLP(v) with maximum optimal value.

Proposition 4.2. *Let $(\eta, \lambda, \mu, \pi, \rho)$ denote a primal-dual optimal solution to* PLP(v) *and* DLP(v) *defined by some vertex v in V such that $\eta > 0$. Then, $H(\pi, \rho)$ is a supporting hyperplane to Z_{in} at $y = P^T\lambda + Q^T\mu$ with normal vector π in $C^* \setminus \{0\}$.*

Proof. Feasibility and boundedness of PLP(v) and DLP(v) by strong duality for linear programming imply that optval(PLP(v)) = optval(DLP(v)), or equivalently

$$\eta = \rho - v^T\pi. \tag{4.3}$$

This result, the assumption that $\eta > 0$, (4.2b), and (4.2e) together imply that $\pi \neq 0$. The set $H(\pi, \rho)$ thus forms a hyperplane in objective space with normal π in $C^* \setminus \{0\}$ by feasibility with respect to (4.2c). To show that $H(\pi, \rho)$ supports Z_{in} at y, it remains to show that Z_{in} is entirely contained in the closed positive halfspace associated with $H(\pi, \rho)$ and that y is contained in $H(\pi, \rho)$. Take any $\bar{y}$ in Z_{in} parameterized by some $\bar{\mu}$ and $\bar{\lambda}$ that are feasible to (4.1), i.e., $\bar{y} = P^T\bar{\lambda} + Q^T\bar{\mu}$. Then,

$$\begin{aligned} \pi^T\bar{y} = \pi^T(P^T\bar{\lambda} + Q^T\bar{\mu}) \geq \{(4.2c),\ (4.1d)\} \geq \pi^T P^T\bar{\lambda} \geq \\ \geq \{(4.2b)\} \geq \rho\bar{\lambda}^T e = \{(4.1c)\} = \rho, \end{aligned} \tag{4.4}$$

which yields the first part of the statement. Forming the scalar product between π and y gives that

$$\begin{aligned} \pi^T y = \pi^T(P^T\lambda + Q^T\mu) \leq \{(4.1b)\} \leq \pi^T(\eta e + v) \leq \\ \leq \{(4.2d)\} \leq \eta + \pi^T v = \{(4.3)\} = \rho. \end{aligned} \tag{4.5}$$

Inserting y in (4.4) gives that $\pi^T y \geq \rho$. Therefore, all inequalities in (4.5) are tight, so that $y \in H(\pi, \rho)$. ∎

4.3. Reducing the number of subproblems to be solved

The number of linear programming subproblems on the form (4.1) that needs to be solved to compute the optimal value of (3.1) can be reduced by upper-bounding the optimal value of (4.1). Consider a sequence of solves of PLP(v) over v in V and let β^* denote the maximum optimal value obtained so far. Then, any vertex $\bar{v}$ in V that is optimal to (3.1) must satisfy optval(PLP$(\bar{v})$) $\geq \beta^*$, and therefore, for any vertex v in V and scalar β such that optval(PLP(v)) $\leq \beta \leq \beta^*$, the instance PLP(v) need not be solved. An upper bound on the optimal value of (4.1) is provided by the following result.

Proposition 4.3. *Let* $Z_{\text{out}} = \{z : Wz \geq r\}$ *denote the outer approximation,* V *its set of vertices and* $\text{PLP}(v)$ *an instance of (4.1) defined by some* v *in* V *in an iteration of the sandwich algorithm, and let the corresponding notation with superscript "+" apply to the subsequent iteration. Then, for any* v *in* V^+*, it holds that*

$$\text{optval}(\text{PLP}^+(v)) = \begin{cases} \text{optval}(\text{PLP}(v)) & \text{if} \quad v \in V^+ \\ \max_{\bar{v} \in E} \text{optval}(\text{PLP}(\bar{v})) & \text{otherwise} \end{cases},$$

where E *is extreme point set of the unique edge of* Z_{out} *that contains* v*.*

Proof. First suppose that $v \in V$ and let (η, λ, μ) be an optimal solution to $\text{PLP}(v)$. Then, $(\eta, (\lambda^T \ 0)^T, \mu)$ is a feasible solution to $\text{PLP}^+(v)$ with objective value $\text{optval}(\text{PLP}(v))$. Hence, $\text{optval}(\text{PLP}^+(v)) \leq \text{optval}(\text{PLP}(v))$. Now suppose that $v \notin V$. Since $v \in V^+$, the system $W^+v \geq r^+$ is satisfied with equality in exactly n linearly independent rows. Similarly, $v \notin V$ and $Z^+_{\text{out}} \subseteq Z_{\text{out}}$ imply that the system $Wv \geq r$ is satisfied with equality in at most $n-1$ linearly independent rows. Then, since W^+ is W augmented with one additional row, this system is satisfied with equality in exactly $n-1$ linearly independent rows. The point v is thus contained in an edge of Z_{out}. By an argument analogous to that in Proposition 4.1, the maximum optimal value of the inner level linear program in (3.1) taken over all points in this edge occurs at one of its extreme points, so that $\text{optval}(\text{PLP}^+(v)) \leq \max_{\bar{v} \in E} \text{optval}(\text{PLP}(\bar{v}))$. ∎

4.4. Enumerating the vertices of the outer approximation

We enumerate the vertices of the outer approximation by first representing this set as a polytope and then converting its halfspace representation to a vertex representation. To perform the latter of these two steps, we use the fact that vertex enumeration is equivalent to a convex hull problem under polar duality between points and hyperplanes defined by a reciprocation $H(a, b) \mapsto (a_1/b, \ldots, a_n/b)^T$ about the unit sphere [41]. We first define a duality relation between polytopes and then outline the vertex enumerative scheme.

Definition 4.1. (Polytope duality) Let A be a polytope that contains the origin in its strict interior. Then, the polytope $A^* = \{z : y^T z \leq 1, \forall y \in A\}$ is the polar dual of A.

Polar duality defines a bijection between the facets of a polytope and the vertices of its dual. This correspondence is inclusion-reversing in the sense that two facets that incident on a common ridge are in bijection with two vertices contained in a common edge [25, Thm. 3.4.4]. Polar duality is moreover a reflexive transformation, so that twice dualizing a polytope gives back the initial polytope [25, Thm. 3.4.3].

We use the above the theory to enumerate the vertices of the outer approximation by the following steps: (i) augment the outer approximation with n sufficiently large upper bounds so that the resulting set is closed and bounded; (ii) identify a point in the interior of the resulting primal polytope, e.g., the arithmetic mean of the

vertices of the inner approximation; (iii) translate the coordinate system so that this point is the origin; (iv) dualize the outwards oriented bounding hyperplanes of the primal polytope; (v) solve for the convex hull of the resulting points, thus obtaining a halfspace representation of the dual polytope; (vi) dualize the facet-inducing hyperplanes of the dual polytope (vii) translate the resulting points back into the initial coordinate system; (viii) remove any point that satisfy any of the auxiliary upper bounds with equality. The resulting set of points form the vertices of the outer approximation.

4.5. Performing the polyhedral computations on-line

The problem that arises in step (v) of the procedure described in the above section is the so-called on-line convex hull problem: we are given points one point at a time and after receiving each point, we are to compute the convex hull of the points received so far. The variant of this problem in which all input points are known in advance is called the off-line convex hull problem.

We solve the on-line convex hull problem by maintaining a graph representation of the current convex hull with facets as nodes and ridges between adjacent facets as edges. We make the mild assumption that the vertices of the dual polytope are in nondegenerate position, i.e., no $(n+1)$-tuple of points lie in a common hyperplane. By this assumption, any facet of the dual polytope is an $(n-1)$-simplex incident on exactly n ridges, and dually, exactly n edges of the primal polytope are incident on any common vertex [25]. Nondegeneracy can be simulated using standard perturbations techniques, see, e.g., [19, p. 185].

The facet graph is updated using a so-called beneath-and-beyond step, see, e.g., [19, 41]. In brief, one such step processes a new point by partitioning the facets of the current convex hull into disjoint sets of visible and obscured facets. A facet is visible if it contains the new point in its associated open negative halfspace. Obscured facets are reversibly defined. One visible facet is first identified. Remaining visible facets then found by depth-first search through adjacent visible facets, here using that the set of visible facets form a connected subgraph. A cone of new facets is created from the new point to all ridges on which one visible and one obscured facet are incident. The visible facets are finally deleted.

Efficiently identifying the first visible facet is non-trivial in the general on-line version of the convex hull problem. However, since the vertex v of the outer approximation that in a given iteration was found to be most distant from the inner approximation cannot be a vertex of the outer approximation in the subsequent iteration, a visible facet is for our problem instance immediately available as the facet dual to v. With an on-line convex hull algorithm, p iterations of the sandwich algorithm requires p number of beneath-and-beyond steps. This should be contrasted to $\sum_{k=1}^{p} k = p(p+1)/2$ steps with the straightforward solution of calling an off-line algorithm in every iteration.

The upper-bounding technique of Section 4.3 can be incorporated with the described convex hull method as follows. With each facet of the dual polytope that is dual to a vertex v of the outer approximation, we attach the current best upper

bound on optval(PLP(v)). This upper bound is updated whenever PLP(v) is solved. At the creation of a new facet, its upper bound is initialized as the maximum over upper bounds attached to any of the two facets incident on the ridge by which the new facet is induced. Validity of this update rule follows from that the ridge is dual to an edge of the outer approximation defined as in Proposition 4.3 by the incidence-reversing property of polytope duality.

5. Comparison with the facet enumerative algorithm

5.1. Solution by enumerating the facets of the inner approximation

In the algorithm of Rennen et al., problem (3.1) is solved by enumerating the facet-inducing hyperplanes of the inner approximation. Let F denote the set of facet-inducing hyperplanes of Z_{in} and take any hyperplane $H(a, b)$ in F. Then, if λ and μ in (3.1) are restricted to values such that $P^T\lambda + Q^T\mu \in H(a, b)$, the optimal value function of the inner level linear program in (3.1) decomposes by algebraic manipulations into $(b - a^T z)/(a^T e)$. The optimal value of (3.1) can thus be obtained by solving a linear program

$$(\text{LP}(a)) \qquad \begin{array}{ll} \underset{z}{\text{minimize}} & a^T z \\ \text{subject to} & Wz \geq r, \end{array} \tag{5.1}$$

for each hyperplane $H(a, b)$ in F and taking $\max_{H(a,b)\in F}(b - \text{optval}(\text{LP}(a)))/(a^T e)$. The normal a of the hyperplane at which the maximum is attained is taken as the next weighting vector in this approach. The set F is determined by computing the convex hull of the union of P and the set $\{p + \theta q : p \in P, q \in Q\}$, with θ being a fixed sufficiently large scalar. The facet enumerative algorithm can be enhanced with an upper-bounding procedure completely analogous to that outlined for the vertex enumerative algorithm.

5.2. Correspondence between algorithms

The vertex enumerative and the facet enumerative schemes are both methods for removing the nonlinearity of (3.1). The two resulting linear programs (5.1) and (4.2) have closely related geometric interpretations. Consider an instance of (5.1). This problem is to shift a facet-inducing hyperplane in its negative normal direction until this hyperplane supports the outer approximation. Take any vertex of the outer approximation contained in the face induced by the resulting supporting hyperplane. Then, if reversibly shifting a hyperplane in its positive normal direction from this vertex until the hyperplane supports the inner approximation, this corresponds to (4.2). It should be noted that the normal of the shifted hyperplane is kept fixed in (5.1) whereas it is a free variable in (4.2). Also, the hyperplane that is shifted when solving (5.1) induces an $(n-1)$-face of the inner approximation, whereas the hyperplane that is shifted when solving (4.2) induces a general k-face of the inner approximation. The dimensionality k here depends on the choice of linear programming algorithm. If (4.1) or (4.2) is solved by the simplex method that converges at vertex solutions, the solution to (4.2) will satisfy n of the components of (4.2b)

and (4.2c) with equality, so that $k = n - 1$. If instead using an interior point method that converges at the analytic center of the optimal face, the number of binding constraints and hence the dimensionality k may be lower. For a given pair of inner and outer approximation, the vertex enumerative and the facet enumerative approaches thus provide an identical upper bound on the approximation error, but the weighting vector returned by the two approaches need not be equal. We make no conjecture as to the best method with respect to generating weights, except to observe that the vertex enumerative allows for greater flexibility.

5.3. Computational complexity

Disregarding from the inevitable solves of SUM(w) problems, the computational cost of an iteration of the vertex enumerative algorithm consists of the cost of enumerating the vertices of the outer approximation and the cost of solving (3.1) by a sequence of linear programming subproblems. These two costs are directly proportional to the number of visible facets of the dual polytope [19] and the number of subproblems on the form (4.1) that is solved, respectively. Both these figures are bounded from above by the number of facets of the dual polytope. A tight upper bound on the number of facets of a convex hull of k points in an n-dimensional Euclidian space is given by the upper bound conjecture proved by McMullen [35], namely

$$\varphi(k,n) = \binom{k - \lfloor \frac{n+1}{2} \rfloor}{k-n} + \binom{k - \lfloor \frac{n+2}{2} \rfloor}{k-n}.$$

Then, since the dual polytope in the kth iteration of the vertex enumerative scheme is the convex hull of $2n + k + 1$ points, the total cost for p iterations in this scheme is bounded by $\sum_{k=2n+1}^{2n+p+1} \mathcal{O}(\varphi(k,n)) \leq \mathcal{O}(p\varphi(2n+p+1,n))$. In the kth iteration of the facet enumerative scheme, the polytope representation of the inner approximation is the convex hull of $(k+1)(n+1)$ points. By analogous reasoning, its total cost for p iterations is thus bounded by $\sum_{k=n+1}^{n+p+1} \mathcal{O}(\varphi((k+1)(n+1),n)) \leq \mathcal{O}(p\varphi((p+1)(n+1),n))$. Figure 1 illustrates the worst-case complexity of the vertex and the facet enumerative scheme as a function of problem dimension, at various fixed number of iterations p.

6. Test problems and numerical results

6.1. Test problems

We evaluate the proposed algorithm with respect to two test problems. Both problems are constructed to be scalable in the number of objectives and can be made to comply with the assumptions stated in Section 2.2 by introducing some sufficiently large upper bounds on the variables.

Problem 6.1. This is a randomly generated extension of test case 1 in [42] on the

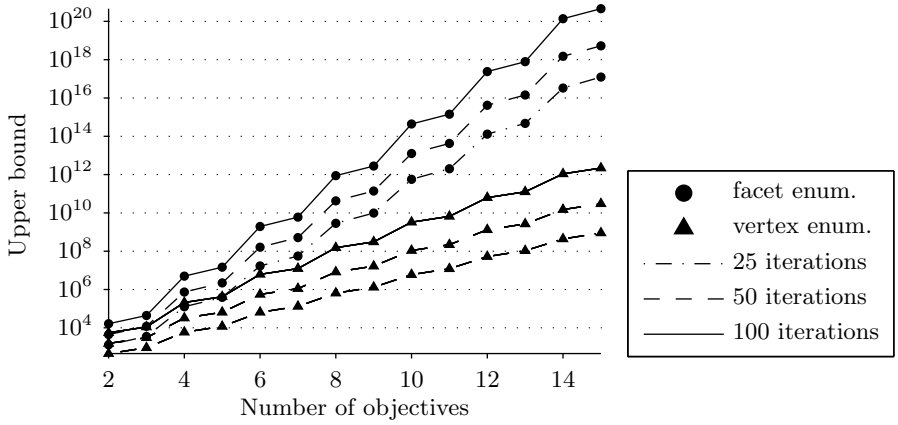


Figure 1: Upper bound on number of beneath-and-beyond steps and number of linear programming solves as a function of number of objectives and total number of sandwich algorithm iterations.

form

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \{x_1, \ldots, x_n\} \\ \text{subject to} \quad & \sum_{j \neq i} (x_j - a_j)^2 - x_i \leq 0, \quad i = 1, \ldots, n, \end{aligned}$$

where a is an n-vector of integers drawn uniformly at random from $\{1, \ldots, n\}$. No bounds on the trade-off rate between objectives were imposed for this problem.

Problem 6.2. This is an example of an IMRT optimization problem for a head and neck cancer case. Data for this problem was exported from the RayStation treatment planning system (RaySearch Laboratories, Stockholm, Sweden). The goal of IMRT is to deliver a highly conformal radiation dose to the tumor volume, as reviewed in, e.g., [1, 3]. Target coverage must be traded against sparing of radiosensitive organs in its vicinity. We consider the problem of optimizing incident energy fluence. This problem was posed on the form (2.1) by assigning objectives and constraints to each anatomical structure. All objective and constraint functions were constructed as one-sided quadratic penalties of the deviations in voxel dose from a reference dose level, as made explicit in Appendix A. A bound $t_{ij} = 10^{-2}$ on the trade-off rate between all pairs of objectives (i, j) was introduced as to zoom into the high-curvature region of the Pareto surface. A representative optimized dose distribution is illustrated in Figure 2.

6.2. Numerical results

We report the results of applying the vertex and the facet enumerative algorithms to Problems 6.1 and 6.2, in conjunction with, and without, the proposed upper-bounding procedure (called bookkeeping for short). Both algorithms were implemented in C++ using identical linear algebra routines and interfaced to Matlab.

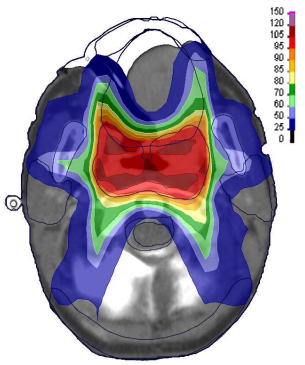


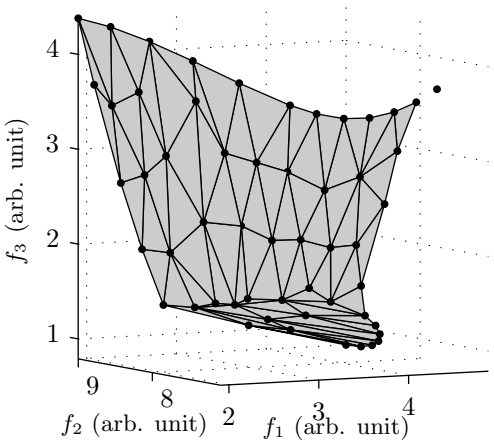
Figure 2: Transversal slice of a dose distribution associated with a Pareto optimal solution to Problem 6.2. The color table is in relative percent of the prescription level. Contours indicate borders of anatomical structures.

Nonlinear problems on the form (2.2) were solved using the barrier method of CPLEX 10.2 (ILOG, Sunnyvale, CA) with default settings. Linear programs on the form (4.1) were solved using the primal simplex method built into SNOPT 7.2 [24], with problems sorted in descending order with respect to available upper bounds. These solves are amenable to parallelization, but for ease of comparison, all computations were run under 64-bit Linux on a single Intel Xeon 3 GHz processor core with hyperthreading disabled and with 32 GB of memory. A timeout of three hours was set for all processes as to keep the overall running time reasonable.

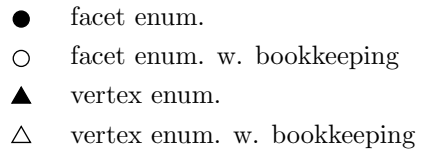
The convex hull representation of the inner approximation was empirically observed to be a degenerate polytope, manifesting as multiple faces induced by near-identical hyperplanes. Since multiple solves over such hyperplanes does not contribute considerable to the solution of (3.1), any hyperplane identified as duplicate within a tolerance of 10^{-5} was disregarded.

For each problem and algorithm, we report the number of beneath-and-beyond steps, the number of linear programming solves, and CPU time, summed over 50 iterations of the sandwich algorithm. In addition, we report the upper bound on the approximation error as a function of iteration number. The numerical results obtained for Problems 6.1 and 6.2 are summarized in Figure 3 and Figure 4, respectively. We stress that our research implementation is not optimized for speed and the reported running times given only for comparative purposes.

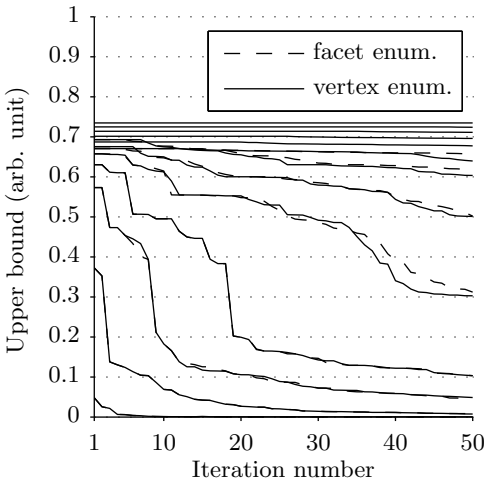
Based on the depicted results, we conclude that the vertex and the facet enumerative scheme are equivalent in terms of approximation guarantee. In terms of computational load, the combined effect of the vertex enumerative scheme and the proposed upper-bounding procedure results in an improvement that is increasing with problem dimension. For the two studied problems, the proposed enhancements translates into a reduction in the number of linear programming solves by one order of magnitude for dimensions beyond two, and a reduction by two orders of magnitude for dimension beyond five. Correspondingly, the number of dimensions tractable at computational times within the order of minutes increases from about six to eleven.



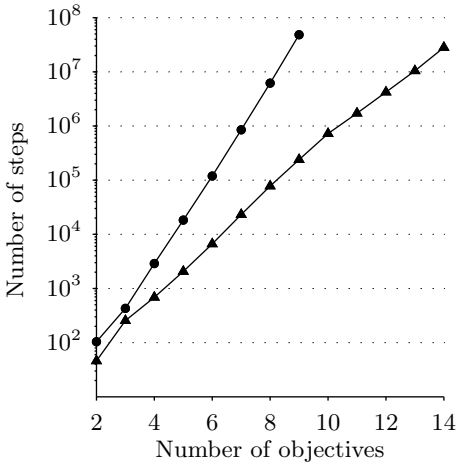
(a) Pareto surface representation.



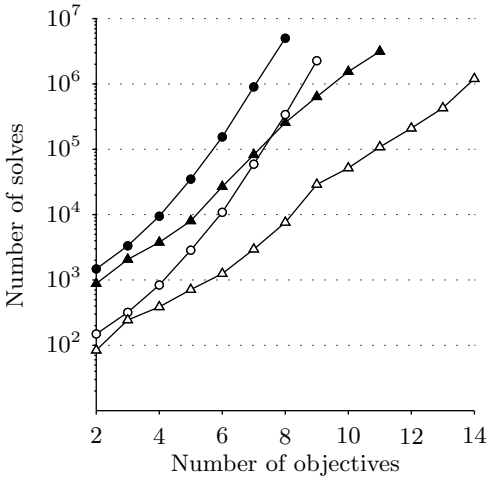
(b) Legend for graphs (d)–(f).



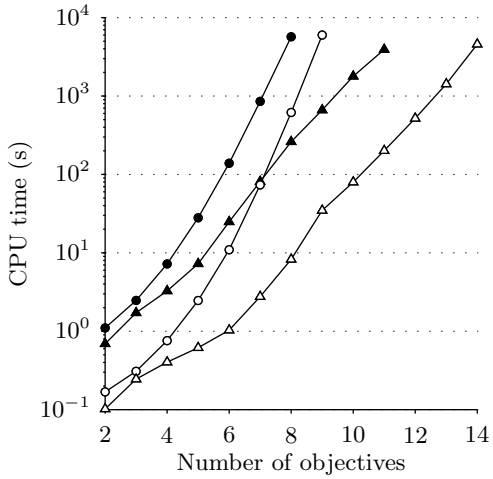
(c) Upper bound on the approximation error.



(d) Number of beneath-and-beyond steps.

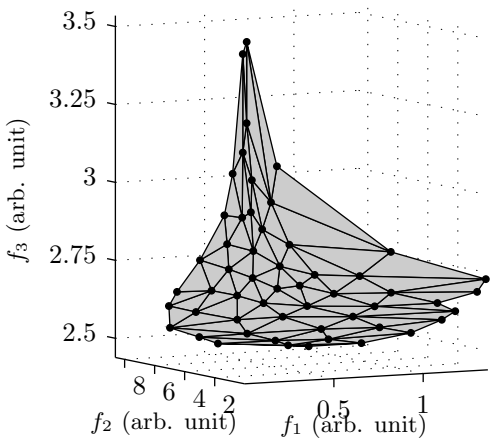


(e) Number of linear programming solves.

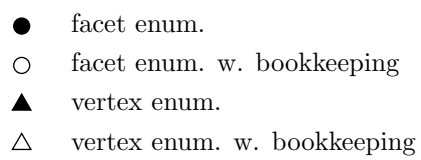


(f) CPU time.

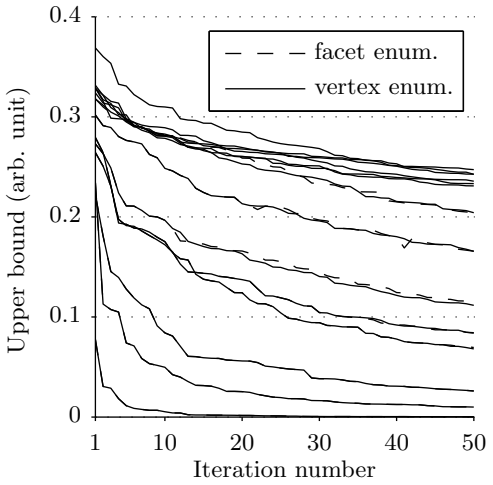
Figure 3: Numerical results of applying 50 sandwich algorithm iterations to Problem 6.1. (a): Pareto surface representation at $n = 3$. (c): upper bound on the approximation error as a function of iteration number at a fixed problem dimension n. The lowermost curve corresponds to $n = 2$ and the uppermost curve to $n = 14$. (d)–(f): all depicted quantities are summed over 50 iterations.



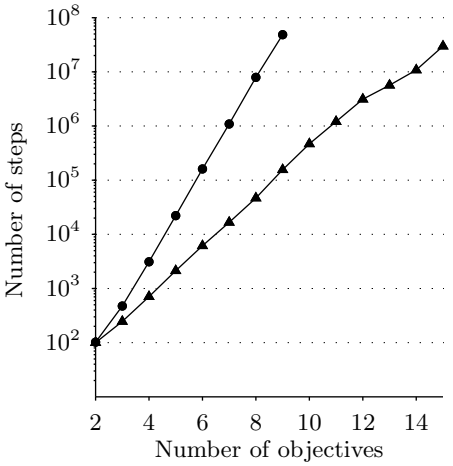
(a) Pareto surface representation.



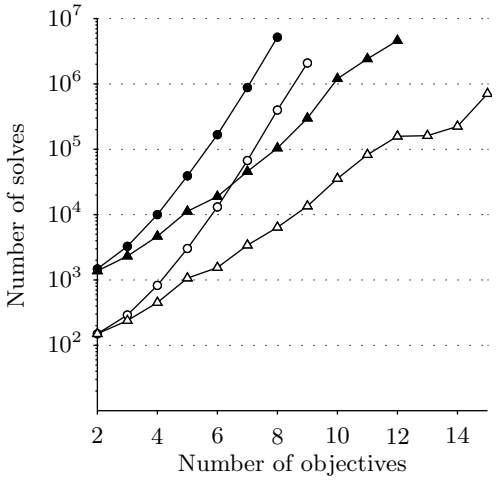
(b) Legend for graphs (d)–(f).



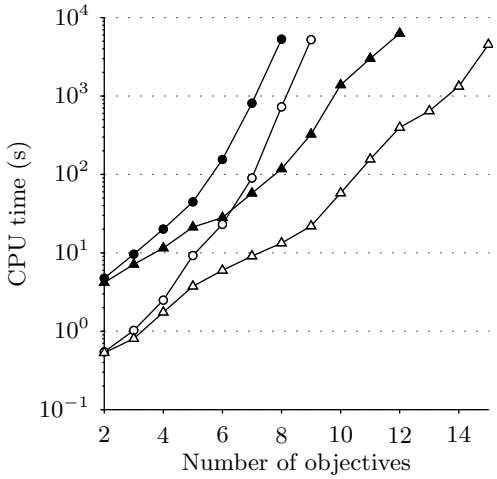
(c) Upper bound on the approximation error.



(d) Number of beneath-and-beyond steps.



(e) Number of linear programming solves.



(f) CPU time.

Figure 4: Numerical results of applying 50 sandwich algorithm iterations to Problem 6.2. (a): Pareto surface representation at $n = 3$. (c): upper bound on the approximation error as a function of iteration number at a fixed problem dimension n. The lowermost curve corresponds to $n = 2$ and the uppermost curve to $n = 15$. (d)–(f): all depicted quantities are summed over 50 iterations.

7. Summary and discussion

We have proposed a sandwich algorithm for approximating the Pareto surface of a convex multi-objective optimization problem based on enumerating the vertices of an outer polyhedral approximation of the Pareto surface. The proposed method is in a sense dual to a previously suggested algorithm based on enumerating the facets of the inner approximation. The two enumerative schemes were further enhanced with an upper-bounding procedure for reducing the number of subproblem solves required to solve a nonconvex optimization problem. This procedure was made possible by implementing the polyhedral computations in an on-line fashion.

The vertex and the facet enumerative algorithms are both exact methods for maximizing the improvement in bound on the approximation error when generating a single Pareto optimal solution. As a result, the two methods are equivalent in terms of quality of output, as was verified experimentally. The vertex enumerative scheme was shown to improve upon both worst-case complexity and practical performance of the sandwich algorithm. This improvement can be attributed to the fact that the vertex enumerative approach handles the normal vectors of the inner approximation, which is the more structurally complex polyhedron of the inner and outer approximations, as a free variable in the linear programming subproblems. In the facet enumerative approach, these normal vectors are instead explicitly given in the statement of its subproblems, leading to more costly polyhedral computations and a larger number of subproblems that needs to be solved.

We conclude by summarizing the implications for the IMRT application. There is yet no widely accepted consensus on acceptable computational time for generating a discrete representation of the Pareto surface for this application. However, judging by a recent clinical evaluation [15] where total planning time was in the order of ten minutes, running times much beyond a number minutes appears unrealistic. Based on our numerical experience, solving the Pareto surface approximation problem in presence of the up to about ten problem dimensions that are of interest in IMRT appears tractable in view of the proposed enhancements. We thus envisage that sandwich algorithms will allow for better resolved models of the viable treatment options in the form of more accurately represented Pareto surfaces throughout the spectrum of problem formulations encountered in IMRT optimization.

A. Formulation of problem 6.2

The patient volume was discretized into $5 \times 5 \times 5$ mm^3 volume elements (voxels) and the beam planes into 1×1 cm^2 surface elements (bixels). Dose kernels for five coplanar photon beams at equispaced gantry angles were computed using a pencil beam convolution technique based on singular value decomposition, similar to [4]. The problem was posed on the form (2.1) by taking the elements of x to be the energy fluence per bixel and introducing a nonnegativity bound $x \geq 0$. All objectives and constraints were modeled by minimum and a maximum dose functions on the form

$$g(x) = \sum_{i \in S} \Delta v_i \Theta(p_i^T x, d^{\text{ref}}) \left(p_i^T x - d^{\text{ref}} \right)^2,$$

where S indexes the voxels included in the anatomical structure to which the function is assigned, Δv_i denotes the relative volume of the ith voxel with respect to S, p_i is a pencil beam kernel such that $d_i = p_i^T x$, and where $\Theta(d_i, d^{\text{ref}}) = (d^{\text{ref}} - d_i)_+$ for minimum dose functions and $\Theta(d_i, d^{\text{ref}}) = (d_i - d^{\text{ref}})_+$ for maximum dose functions.

The target structure was assigned with a minimum and a maximum dose objective with $d^{\text{ref}} = 70$ Gy and a minimum dose constraint with $d^{\text{ref}} = 63$ Gy. A maximum dose objective was introduced with $d^{\text{ref}} = 0$ Gy for each healthy structure contained in the projection of the target volume onto the beam planes. The resulting total number of objectives was 15. A constraint on global maximum dose at $d^{\text{ref}} = 77$ Gy was introduced by sampling 2 % of all voxels in the patient volume uniform at random, as to keep running times reasonable. The problem was posed as an inequality constrained quadratic program with 5416 variables and 6937 linear constraints by introducing auxiliary variables, see [8].

Scaling in the number of objectives was performed by aggregating positively correlated objectives. Each objective was first optimized individually. Objectives for healthy structures where then aggregated into composite functions being the direct sum of all constituent functions by iteratively grouping together the two objectives showing maximum degree of mutual monotonocity, as determined by maximum Spearman rank correlation, see, e.g., [29].

Acknowledgements

The authors thank Björn Hårdemark and Henrik Rehbinder for valuable discussion during the course of this work. We also thank Göran Sporre for careful reading and comments that improved the manuscript.

References

[1] A. Ahnesjö, B. Hårdemark, U. Isacsson, and A. Montelius. The IMRT information process—mastering the degrees of freedom in external beam therapy. *Physics in Medicine and Biology*, 51(13):R381–R402, 2006.

[2] R. Bellman. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.

[3] T. Bortfeld. IMRT: a review and preview. *Physics in Medicine and Biology*, 51(13):R363–R379, 2006.

[4] T. Bortfeld, W. Schlegel, and B. Rhein. Decomposition of pencil beam kernels for fast dose calculations in three-dimensional treatment planning. *Medical Physics*, 20(2):311–318, 1993.

[5] J. Branke, K. Deb, K. Miettinen, and R. Slowinski. *Multiobjective Optimization—Interactive and Evolutionary Approaches*. LNCS 5252, State-of-the-Art Survey. Springer, 2008.

[6] D. Bremner, K. Fukuda, and A. Marzetta. Primal-dual methods for vertex and facet enumeration. *Discrete and Computational Geometry*, 20:333–357, 1998.

[7] R. Burkard, H. Hamacher, and G. Rote. Sandwich approximation of univariate convex functions with an application to separable convex programming. *Naval Research Logistics*, 1991.

[8] F. Carlsson, A. Forsgren, H. Rehbinder, and K. Eriksson. Using eigenstructure of the Hessian to reduce the dimension of the intensity modulated radiation therapy optimization problem. *Annals of Operations Research*, 148(1):81–94, 2006.

[9] C. Cotrutz, M. Lahanas, C. Kappas, and D. Baltas. A multiobjective gradient-based dose optimization algorithm for external beam conformal radiotherapy. *Physics in Medicine and Biology*, 46(8):2161–2175, 2001.

[10] D. Craft. Calculating and controlling the error of discrete representations of Pareto surfaces in convex multi-criteria optimization. *Physica Medica*, 26(4):184–191, 2010.

[11] D. Craft and T. Bortfeld. How many plans are needed in an IMRT multi-objective plan database? *Physics in Medicine and Biology*, 53(11):2785–2796, 2008.

[12] D. Craft, T. Halabi, and T. Bortfeld. Exploration of tradeoffs in intensity-modulated radiotherapy. *Physics in Medicine and Biology*, 50(24):5857–5868, 2005.

[13] D. Craft, T. Halabi, H. Shih, and T. Bortfeld. Approximating convex Pareto surfaces in multiobjective radiotherapy planning. *Medical Physics*, 33(9):3399–3407, 2006.

[14] D. Craft, T. Halabi, H. Shih, and T. Bortfeld. An approach for practical multiobjective IMRT treatment planning. *International Journal of Radiation Oncology, Biology, Physics*, 69(5):1600–1607, 2007.

[15] D. Craft, T. Hong, H. Shih, and T. Bortfeld. Improved planning time and plan quality through multi-criteria optimization for intensity modulated radiation therapy. *International Journal of Radiation Oncology, Biology, Physics*, Forthcoming 2011.

[16] D. Craft and M. Monz. Simultaneous navigation of multiple Pareto surfaces, with an application to multicriteria IMRT planning with multiple beam angle configurations. *Medical Physics*, 37(2):736–741, 2010.

[17] I. Das and E. Dennis. A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Structural and Multidisciplinary Optimization*, 14:63–69, 1997.

[18] S. Dempe. *Foundation of Bilevel Programming*. Nonconvex optimization and its applications. Kluwer Academic, 2002.

[19] H. Edelsbrunner. *Algorithms in Combinatorial Geometry*. Springer-Verlag, 1987.

[20] M. Ehrgott. *Multicriteria Optimization*, volume 491 of *Lecture Notes in Economics and Mathematical Systems*. Springer Verlag, 2nd edition, 2005.

[21] A. Engau. Tradeoff-based decomposition and decision-making in multiobjective programming. *European Journal of Operational Research*, 199(3):883–891, 2009.

[22] P. Eskelinen, K. Miettinen, K. Klamroth, and J. Hakanen. Pareto navigator for interactive nonlinear multiobjective optimization. *OR Spectrum*, 32:211–227, 2010.

[23] B. Fruhwirth, R. E. Bukkard, and G. Rote. Approximation of convex curves with application to the bicriterial minimum cost flow problem. *European Journal of Operational Research*, 42(3):326–338, 1989.

[24] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Review*, 47(1):99–131, 2005.

[25] B. Grünbaum. *Convex Polytopes*. Springer, 2nd edition, 2003.

[26] T. Hong, D. Craft, F. Carlsson, and T. Bortfeld. Multicriteria optimization in intensity-modulated radiation therapy treatment planning for locally advanced cancer of the pancreatic head. *International Journal of Radiation Oncology, Biology, Physics*, 72(4):1208–1214, 2008.

[27] B. Hunt and M. Wiecek. Cones to aid decision making in multicriteria programming. In T. Tanino, T. Tanaka, and M. Inuiguchi, editors, *Multi-Objective Programming and Goal Programming*, pages 153–158. Springer Verlag, 2003.

[28] M. Hunt, C. Hsiung, S. Spirou, C. Chui, H. Amols, and C. Ling. Evaluation of concave dose distributions created using an inverse planning system. *International Journal of Radiation Oncology, Biology, Physics*, 54:953–962, Nov 2002.

[29] M. Kendall. *Rank Correlation Methods*. Griffin, 1962.

[30] K. Klamroth, J. Tind, and M. Wiecek. Unbiased approximation in multicriteria optimization. *Mathematical Methods of Operations Research*, 56:413–437, 2003.

[31] K.-H. Küfer, A. Scherrer, M. Monz, F. Alonso, H. Trinkaus, T. Bortfeld, and T. Christian. Intensity-modulated radiotherapy—a large scale multi-criteria programming problem. *OR Spectrum*, 25:223–249, 2003.

[32] M. Lahanas, E. Schreibmann, and D. Baltas. Multiobjective inverse planning for intensity modulated radiotherapy with constraint-free gradient-based optimization algorithms. *Physics in Medicine and Biology*, 48(17):2843, 2003.

[33] A. Madansky. Inequalities for stochastic linear programming problems. *Mangagement Science*, 6(2):197–204, 1960.

[34] R. Marler and J. Arora. The weighted sum method for multi-objective optimization: new insights. *Structural and Multidisciplinary Optimization*, 41:853–862, 2010.

[35] P. McMullen. The maximum number of faces of a convex polytope. *Mathematika*, 17(2):179–184, 1970.

[36] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic, 1999.

[37] M. Monz. *Pareto Navigation—Interactive Multiobjective Optimisation and its Application in Radiotherapy Planning*. PhD thesis, University of Kaiserslautern, 2006.

[38] M. Monz, K.-H. Kufer, T. R. Bortfeld, and C. Thieke. Pareto navigation—algorithmic foundation of interactive multi-criteria IMRT planning. *Physics in Medicine and Biology*, 53(4):985–998, 2008.

[39] P. M. Pardalos. Enumerative techniques for solving some nonconvex global optimization problems. *OR Spectrum*, 10:29–35, 1988.

[40] T. Pennanen and J. Eckstein. Generalized Jacobians of vector-valued convex functions. Technical report, Rutgers University, 1997.

[41] F. P. Preparata and M. Shamos. *Computational Geometry—An Introduction*. Springer-Verlag, 1985.

[42] G. Rennen, E. van Dam, and D. den Hertog. Enhancement of sandwich algorithms for approximating higher-dimensional convex Pareto sets. *INFORMS Journal on Computing*, published online before print, December 2, 2010.

[43] R. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[44] E. Romeijn, J. Dempsey, and J. Li. A unifying framework for multi-criteria fluence map optimization models. *Physics in Medicine and Biology*, 49(10):1991–2013, 2004.

[45] G. Rote. The convergence rate of the sandwich algorithm for approximating convex functions. *Computing*, 48:337–361, 1992.

[46] S. Ruzika and M. Wiecek. Approximation methods in multiobjective programming. *Journal of Optimization Theory and Applications*, 126:473–501, 2005.

[47] J. Serna, M. Monz, K.-H. Küfer, and C. Thieke. Trade-off bounds for the Pareto surface approximation in multi-criteria IMRT planning. *Physics in Medicine and Biology*, 54(20):6299, 2009.

[48] R. Solanki, P. Appino, and J. Cohon. Approximating the noninferior set in multiobjective linear programming problems. *European Journal of Operational Research*, 68(3):356–373, 1993.

[49] T. Spalke, D. Craft, and T. Bortfeld. Analyzing the main trade-offs in multiobjective radiation therapy treatment planning databases. *Physics in Medicine and Biology*, 54(12):3741–3754, 2009.

[50] J. Stoer and C. Witzgall. *Convexity and Optimization in Finite Dimensions I*. Springer-Verlag, 1970.

[51] C. Thieke, K.-H. Küfer, M. Monz, A. Scherrer, F. Alonso, U. Oelfke, P. Huber, J. Debus, and T. Bortfeld. A new concept for interactive radiotherapy planning with multicriteria optimization: First clinical evaluation. *Radiotherapy and Oncology: Journal of the European Society for Therapeutic Radiology and Oncology*, 85(2):292–298, 2007.