Annals of Operations Research 119, 183–203, 2003

The Least-Intensity Feasible Solution for Aperture-Based Inverse Planning in Radiation Therapy

Y. XIAO, D. MICHALSKI and J.M. GALVIN {ying.xiao, darek.michalski, james.galvin}@mail.tju.edu
Medical Physics Division, Radiation Oncology Department, Thomas Jefferson University Hospital, 10th and Locust Streets, Philadelphia, PA 19107, USA

Y. CENSOR yair@math.haifa.ac.il
Department of Mathematics, University of Haifa, Mt. Carmel, Haifa 31905, Israel

Abstract. Aperture-based inverse planning (ABIP) for intensity modulated radiation therapy (IMRT) treatment planning starts with external radiation fields (beams) that fully conform to the target(s) and then superimposes sub-fields called segments to achieve complex shaping of 3D dose distributions. The segments' intensities are determined by solving a feasibility problem. The least-intensity feasible (LIF) solution, proposed and studied here, seeks a feasible solution closest to the origin, thus being of least intensity or least energy. We present a new iterative, primal–dual, algorithm for finding the LIF solution and explain our experimental observation that Cimmino's algorithm for feasibility actually converges to a close approximation of the LIF solution. Comparison with linear programming shows that Cimmino's algorithm has the additional advantage of generating much smoother solutions.

1. Introduction

A recently proposed technique for intensity modulated radiation therapy (IMRT) treatment planning has been investigated by Xiao et al. in [45]. This is a rule-based method that starts by choosing beams that conform to the entire target or set of targets. The orientation or direction of the beams are user-selected as are any beam modifiers such as wedges that can slant the dose distribution relative to a perpendicular entry surface.

The rules add sub-fields (called *segments*) to these fully conformal beams to create dose distributions that follow irregular surfaces of a target and allow for different dose levels for a combination of targets. Successful implementation of this approach requires solving the inverse problem to determine the intensities of these segments. (We intentionally use here the term "intensity" instead of "weight", which is commonly used in this field, because "weights of importance" appear later on, with a different meaning, in our algorithms.) This approach, which we call here *aperture-based inverse planning* (ABIP), simplifies the computational burden associated with the *inverse problem* of a *fully discretized model* where the radiation field is represented by beamlets (i.e., pencil-beams or rays resulting from the disceretization of beams). The inverse problem of a fully discretized model in IMRT has been formulated and solved as a mathematical *feasibility problem* by Altschuler and Censor [2] and further developed by Altschuler,

VTEX(P) PIPS No:5115484 artty:res (Kluwer BO v.2002/10/03)
a5115484.tex; 24/02/2003; 16:03; p. 1

Censor and Powlis in [4,15,16,20,41] and [42]. See also [3], Censor [14] and Censor and Zenios [21, chapter 11]. For a recent review see, e.g., Shepard et al. [43].

The ABIP approach ameliorates many of the shortcomings of the inverse planning with a fully discretized model and may well satisfy the critics who claim that beamlets models are unrealistic, that the computational burden of inverse planning is too severe, or that the number of beams required is too great. The encouraging results with the ABIP problem, presented in this paper, constitute an advance toward clinical acceptance of inverse planning.

In either ABIP or the fully discretized model the situation is modelled as a problem of solving a system of linear interval inequalities, referred to as the *linear interval feasibility problem*. One approach to finding a solution of such systems, that has been proven efficacious in solving the fully discretized inverse problem of radiation therapy treatment planning (RTTP), is to use iterative *projection algorithms*, preferably of the *simultaneous* type, to generate a sequence of approximate vectors of intensities converging to a solution of the system, e.g., Censor, Altschuler and Powlis [15,16], Lee et al. [38], Cho and Marks II [22], see also Censor and Zenios [21, chapter 11] and references therein.

In this paper we investigate algorithms for finding the *least-intensity feasible* (LIF) solution of the linear interval feasibility problem, i.e., a solution of the model's linear interval constraints which has the least possible intensities of the segments, in the sense of being as close as possible to the origin in the intensity vectors' space. While aiming at such a solution is, from the mathematical point of view, a mathematical *optimization* problem, the feasibility approach flavor is preserved because no exogenous merit (cost) function – whose merit might be debateable – is imposed on the constraints. Rather we are just doing the very reasonable thing of searching for a feasible solution for the model's constraints with least possible intensities because there is really no reason to settle for an arbitrary feasible solution if there is one available which uses less intensities for the segments.

It turns out that on the algorithmic side such a least-intensity feasible (LIF) solution can be achieved with a simultaneous projections algorithm whose computational demands are not much higher then those of the well-known Cimmino algorithm which finds an arbitrary feasible solution. We present this newly developed algorithm and discuss its origins and merits. Then we demonstrate the viability of all the proposed ingredients: (i) the aperture-based inverse planning method, (ii) the linear interval feasibility problem modeling, and (iii) the new algorithm for finding a LIF solution.

Our computational experimental work revealed a surprise which went unnoticed in the literature about the use of Cimmino's algorithm in RTTP until now. Namely, that the well-known iterative algorithm of Cimmino, when initialized at the origin, yields a solution which repeatedly is very close to a LIF solution. We give a mathematical justification for this phenomenon in the appendix. Finally, comparing the results obtained by Cimmino's algorithm with those obtained with the SIMPLEX method of Linear Programming (LP) shows the greater smoothness of the results obtained by Cimmino's algorithm.

The paper is laid out as follows. The ABIP method for IMRT is presented in section 2. Section 3 is a brief introductory overview on projection algorithms in general while in section 4 the linear feasibility model for the ABIP approach is formulated and the fully simultaneous Cimmino algorithm for its solution is given. The LIF solution and the new primal–dual algorithm for finding it are presented in section 5. Representative results from our experimental computational work are given in section 6 and some concluding remarks are contained in section 7. At the end we give an appendix where the similar behavior of Cimmino's and the LIF algorithms is mathematically explained.

2. Aperture-based inverse planning (ABIP) for IMRT

The basic difference between the *inverse problem of a fully discretized model* and the *aperture-based inverse planning* approaches in IMRT lies in the way the external radiation is accounted for in the model. In the fully discretized approach many small elemental beams (called *beamlets*) projecting to various points covering the target(s) are considered in the model. In fact, for the beam directions (different treatment unit orientations) and elemental beam size chosen to solve the problem, all beamlets that completely or partially impinge on the target(s) are included in the model. The model attaches an unknown value x_i to the intensity of each beamlet and, because of the dense coverage of beams and their fine discretization into beamlets, the total number I of unknowns is very large.

In contrast, ABIP restricts the number of total beams used for the intensity optimization by first considering a single aperture that conforms to all the targets as a single beam. This is done for each orientation of the treatment unit and results in just a few fields (about 4–12). A relatively small number of additional aperture shapes are selected to define field segments of reduced size that superimpose on the fields defined by the initial conformal apertures. This is done in a non-automatic manner in which human judgement is involved to place the beams and define the segments. The feasibility or optimization problem then becomes one of finding the intensities of the conformal fields and the superimposed segments. This approach significantly reduces the size of the intensities vector x compared to the fully discretized method. It also simplifies the overall problem since a second step must be added to the beamlet-based approach. This involves consolidation of beamlets of various intensities into a smaller number of deliverable apertures of larger size. The radiation field segments are defined according to the following guidelines.

Definition 1. Guidelines for segment selection in ABIP:

1. Select a field that conforms to the combined outline of all targets projected back to the radiation point source, and repeat for all orientations of the treatment unit.

2. Select a field that conforms to the projection of the `boost volume` back to the radiation point source, and repeat for all orientations of the treatment unit. The boost

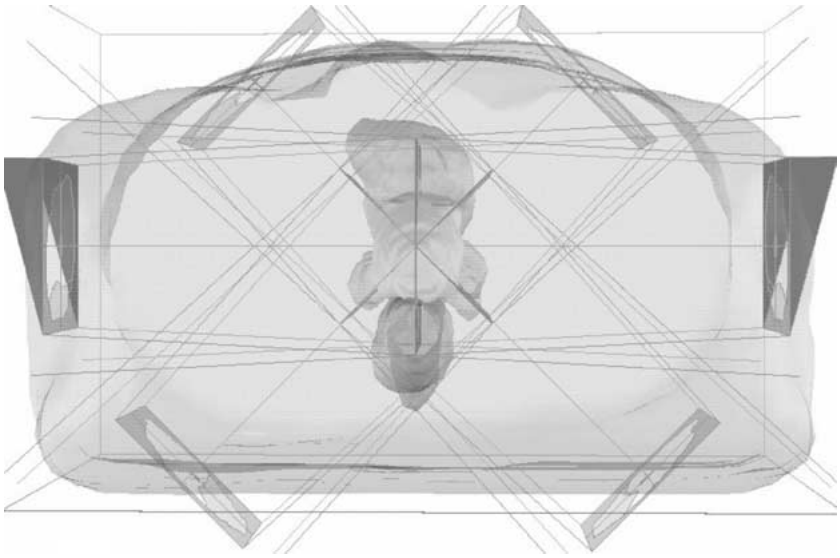


Figure 1. A six beams arrangement for treatment of prostate cancer.

volume is the portion of the target that is to receive a higher dose compared to regions of the target taken to the lowest dose.

3. Repeat step 2 for regions of the target that are to receive the next higher level of dose.
4. For every critical structure lying in the path of the conformal beam of the complete target, select a field segment that conforms to the target but fully shields the critical structure.
5. Repeat the previous step for the field segments that encompass each of the volumes receiving higher dose.
6. Select, for the complete target, extra segments to adjust for the dose inhomogeneity that results from shielding critical structures that do not run along the whole length of the target.

The first step in the procedure produces a number of apertures that is equal to the number of pre-selected beam orientations. This is also true for steps 2 and 3. The fourth step adds many more field segments. This number is at least equal to the number of orientations of the treatment unit multiplied by the number of critical structures falling within the aperture and requiring protection. Other factors can increase this number even more. For treatment planning of an *oropharyngeal cancer* case, with two critical structures (the right parotid, and the spinal cord), the number of segments we obtained by using the guidelines of definition 1 was 99, i.e., 9 superimposed segments per each of the pre-chosen 11 gantry angles, see Galvin et al. [31]. This number of segments was needed because the computer driven device used to define field shapes (called *multileaf*

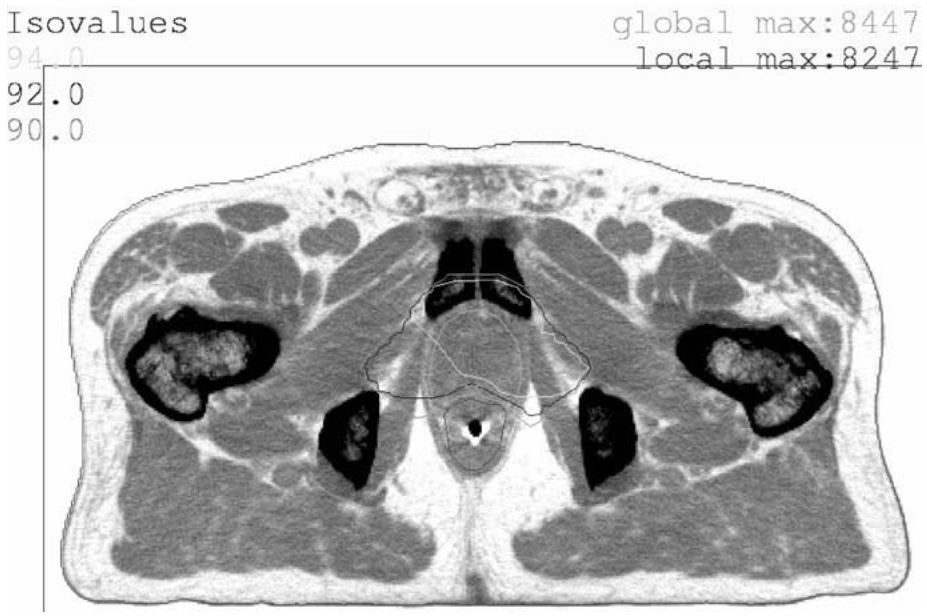


Figure 2. Dose distribution represented by isodose curves for treatment of prostate cancer case, as calculated by using the ABIP model and Cimmino's algorithm.

collimator, MLC) was limited in its ability to adapt to some aperture shapes. For treatment plans of prostate cancer, with bladder and rectum as the critical organs that have to be avoided, the total number of segments is usually in the range of 50–60.

We use here a case of prostate cancer treatment as an illustration of the complete process. We start with six beam angles at 45°, 90°, 135°, 225°, 270°, and 315°, as shown in figure 1. The tumor that needs to be treated is in the prostate (center organ) and the seminal vesicles (the two protruding structures on left and right sides). The organs to be avoided are the rectum (underneath) and the bladder (above). For each of the beam angles, beam apertures are chosen by applying the guidelines of definition 1. The dose distribution due to a unit intensity radiation from each segment is calculated by a state-of-the-art *dose calculation algorithm* and the segment's intensity is computed by the various optimization techniques described in this study. The aim is to create treatment plans that conform to the tumor yet spare critical structures as much as possible. Figure 2 shows a CT (Computed Tomography) cross-section with the resulting isodose curves superimposed. Notice the concavity of the isodose that closely avoids the rectum from underneath.

3. Projection algorithms: A brief overview

As an introduction to the algorithmic part of this paper we give a brief overview of projection algorithms for the convex feasibility problem. *Projection algorithms* employ projections onto convex sets in various ways. They may use different kinds of projections

and, sometimes, even use different projections within the same algorithm. They serve to solve a variety of problems which are either of the feasibility or the optimization types. They have different algorithmic structures, of which some are particularly suitable for parallel computing, and they demonstrate nice convergence properties and/or good initial behavior patterns. This class of algorithms has witnessed great progress in recent years and its member algorithms have been applied with success to fully discretized models of problems in image reconstruction and image processing, see, e.g., Stark and Yang [44] and Censor and Zenios [21].

The *convex feasibility problem* is to find a point (any point) in the non-empty intersection $C := \bigcap_{i=1}^{m} C_i \neq \emptyset$ of a family of closed convex subsets $C_i \subseteq R^n$, $1 \leqslant i \leqslant m$, of the n-dimensional Euclidean space. It is a fundamental problem in many areas of mathematics and the physical sciences, see, e.g., Combettes [24,26] and references therein. It has been used to model significant real-world problems in image reconstruction from projections, see, e.g., Herman [33], in radiation therapy treatment planning, see Censor, Altschuler and Powlis [16] and Censor [14], and in crystallography, see Marks, Sinkler and Landree [40], to name but a few, and has been used under additional names such as *set theoretic estimation* or the *feasible set approach*. A common approach to such problems is to use projection algorithms, see, e.g., Bauschke and Borwein [6], which employ *orthogonal projections* (i.e., nearest point mappings) onto the individual sets C_i. The orthogonal projection $P_\Omega(z)$ of a point $z \in R^n$ onto a closed convex set $\Omega \subseteq R^n$ is defined by

$$P_\Omega(z) := \operatorname{argmin}\big\{\|z - x\|_2 \mid x \in \Omega\big\}, \tag{1}$$

where $\|\cdot\|_2$ is the Euclidean norm in R^n. Frequently a *relaxation parameter* is introduced so that

$$P_{\Omega,\lambda}(z) := (1 - \lambda)z + \lambda P_\Omega(z) \tag{2}$$

is the *relaxed projection* of z onto Ω with relaxation λ.

Another problem that is related to the convex feasibility problem is the *best approximation problem* of finding the projection of a given point $y \in R^n$ onto the non-empty intersection $C := \bigcap_{i=1}^{m} C_i \neq \emptyset$ of a family of closed convex subsets $C_i \subseteq R^n$, $1 \leqslant i \leqslant m$, see, e.g., Deutsch's recent book [27]. In both problems the convex sets $\{C_i\}_{i=1}^{m}$ represent mathematical constraints obtained from the modeling of the real-world problem. In the convex feasibility approach any point in the intersection is an acceptable solution to the real-world problem whereas the best approximation formulation is usually appropriate if some point $y \in R^n$ is given and one wishes to find the point in the intersection of the convex sets which is closest to the point y.

Iterative projection algorithms for finding a projection of a point onto the intersection of sets are more complicated then algorithms for finding just any feasible point in the intersection. This is so because they must have, in their iterative steps, some built-in "memory" mechanism to remember the original point whose projection is sought after. The sequential or parallel algorithms of Dykstra, see, e.g., Bregman, Censor and Reich [11], Haugazeau, see, e.g., Bauschke and Combettes [8], Bauschke [5] and others

and their modifications employ different such memory mechanisms. We will not deal with these algorithms here although many of them share the same algorithmic structural features described below.

Projection algorithmic schemes for the convex feasibility problem or for the best approximation problem are, in general, either *sequential* or *simultaneous* or *block-iterative* (see, e.g., Censor and Zenios [21] for a classification of projection algorithms into such classes, and the review paper of Bauschke and Borwein [6] for a variety of specific algorithms of these kinds). In the following subsections we explain and demonstrate these structures along with the recently proposed *string-averaging* structure. The philosophy behind these algorithms is that it is easier to calculate projections onto the individual sets C_i then onto the whole intersection of sets. Thus, these algorithms call for projections onto individual sets as they proceed sequentially, simultaneously or in the block-iterative or the string-averaging algorithmic modes.

3.1. Sequential projections

The well-known "Projections Onto Convex Sets" (POCS) algorithm for the convex feasibility problem is a *sequential* projection algorithm, see Bregman [9], Gubin, Polyak and Raik [32], Youla [46] and the review papers by Combettes [24,26]. Starting from an arbitrary initial point $x^0 \in R^n$, the POCS algorithm's iterative step is

$$x^{k+1} = x^k + \lambda_k\big(P_{C_{i(k)}}\big(x^k\big) - x^k\big), \tag{3}$$

where $\{\lambda_k\}_{k\geqslant 0}$ are relaxation parameters and $\{i(k)\}_{k\geqslant 0}$ is a *control sequence*, $1 \leqslant i(k) \leqslant m$, for all $k \geqslant 0$, which determines the individual set $C_{i(k)}$ onto which the current iterate x^k is projected. A commonly used control is the *cyclic control* in which $i(k) = k \bmod(m+1)$, but other controls are also available [21]. Bregman's projection algorithm [10], allowed originally only unrelaxed projections, i.e., its iterative step is of the form

$$x^{k+1} = P_{C_{i(k)}}\big(x^k\big), \quad \text{for all } k \geqslant 0. \tag{4}$$

For unity relaxation, i.e., when $\lambda_k = 1$, for all $k \geqslant 0$, (4) coincides with (3).

3.2. The string averaging algorithmic structure

The *string-averaging* algorithmic scheme was proposed by Censor, Elfving and Herman [17], from where the contents of this subsection is taken. For $t = 1, 2, \ldots, M$, let the *string* I_t be an ordered subset of $\{1, 2, \ldots, m\}$ of the form

$$I_t = \big(i_1^t, i_2^t, \ldots, i_{m(t)}^t\big), \tag{5}$$

with $m(t)$ denoting the number of elements in I_t. Suppose that there is a set $S \subseteq R^n$ such that there are operators $R_1, R_2, \ldots, R_m$ mapping S into S and an operator R which maps $S^M = S \times S \times \cdots \times S$ (M times) into S. Initializing the algorithm at an arbitrary

$x^0 \in S$, the iterative step of the string-averaging algorithmic scheme is as follows. Given the current iterate x^k, calculate, for all $t = 1, 2, \ldots, M$,

$$T_t x^k = R_{i^t_{m(t)}} \cdots R_{i^t_2} R_{i^t_1} x^k, \tag{6}$$

and then calculate

$$x^{k+1} = R\big(T_1 x^k, T_2 x^k, \ldots, T_M x^k\big). \tag{7}$$

For every $t = 1, 2, \ldots, M$, this algorithmic scheme applies to x^k successively the operators whose indices belong to the tth string. This can be done in parallel for all strings and then the operator R maps all end-points onto the next iterate x^{k+1}. This is indeed an algorithm provided that the operators $\{R_i\}_{i=1}^m$ and R all have algorithmic implementations. In this framework we get a *sequential* algorithm by the choice $M = 1$ and $I_1 = (1, 2, \ldots, m)$ and a *simultaneous* algorithm by the choice $M = m$ and $I_t = (t),\ t = 1, 2, \ldots, M$.

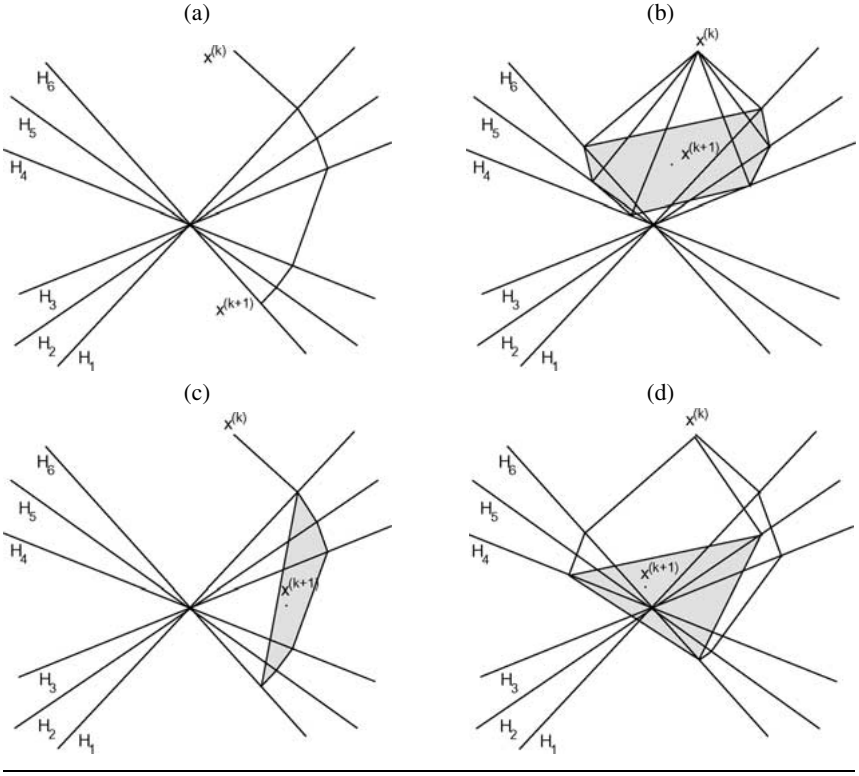
We demonstrate the underlying idea of the string-averaging algorithmic scheme with the aid of the drawings in table 1. For simplicity, we take the convex sets to be hyperplanes, denoted by H_1, H_2, H_3, H_4, H_5, and H_6, and assume all operators R_i to be orthogonal projections onto the hyperplanes. The operator R is taken as a convex combination

$$R\big(x^1, x^2, \ldots, x^M\big) = \sum_{t=1}^{M} \omega_t x^t, \tag{8}$$

with $\omega_t > 0$, for all $t = 1, 2, \ldots, M$, and $\sum_{t=1}^{M} \omega_t = 1$.

Table 1 contains four figures that help to explain the different methods. Table 1(a) depicts the purely sequential algorithm. This is the so-called POCS (Projections Onto Convex Sets) algorithm which coincides, for the case of hyperplanes, with the Kaczmarz algorithm, see, e.g., algorithms 5.2.1 and 5.4.3, respectively, in [21]. The fully simultaneous algorithm appears in table 1(b). With orthogonal reflections instead of orthogonal projections it was first proposed, by Cimmino [23], for solving linear equations. Here the current iterate x^k is projected on all sets simultaneously and the next iterate x^{k+1} is a convex combination of the projected points. In table 1(c) we show how a simple averaging of *successive* projections (as opposed to averaging of parallel projections in table 1(b)) works. In this case $M = m$ and $I_t = (1, 2, \ldots, t)$, for $t = 1, 2, \ldots, M$. This scheme, appearing in Bauschke and Borwein [6], inspired the formulation of the general string-averaging algorithmic scheme whose action is demonstrated in table 1(d). It averages, via convex combinations, the end-points obtained from strings of sequential projections and in this figure the strings are $I_1 = (1, 3, 5, 6)$, $I_2 = (2)$, $I_3 = (6, 4)$. Such schemes offer a variety of options for steering the iterates towards a solution of the convex feasibility problem. It is an *inherently parallel* scheme in that its mathematical formulation is parallel (like the fully simultaneous method mentioned above). We use this term to contrast such algorithms with others which are sequential in their mathematical formulation but can, sometimes, be implemented in a parallel fashion based

Table 1
(a) Sequential projections. (b) Fully simultaneous projections. (c) Averaging of sequential projections. (d) String-averaging. (Reproduced from Censor, Elfving and Herman [17].)



on appropriate model decomposition (i.e., depending on the structure of the underlying problem). Being inherently parallel, this algorithmic scheme enables flexibility in the actual manner of implementation on a parallel machine. At the extremes of the "spectrum" of possible specific algorithms, derivable from the string-averaging algorithmic scheme, are the generically sequential method, which uses one set at a time, and the fully simultaneous algorithm, which employs all sets at each iteration.

The "block-iterative projections" (BIP) scheme of Aharoni and Censor [1] also has the sequential and the fully simultaneous methods as its extremes in terms of block structures (see also Butnariu and Censor [12], Bauschke and Borwein [6], Bauschke, Borwein and Lewis [7], Elfving [30] and Eggermont, Herman and Lent [29]). The string-averaging algorithmic structure gives users a tool to design many new inherently parallel computational schemes.

The behavior of the string-averaging algorithmic scheme, or special instances of it, in the inconsistent case when the intersection $C = \bigcap_{i=1}^{m} C_i$ is empty is not known at this time. For results on the behavior of the fully simultaneous algorithm with orthogonal projections in the inconsistent case see, e.g., Combettes [25] or Iusem and De Pierro [36].

4. The linear feasibility model and projection algorithms for ABIP

We describe next how the aperture-based inverse problem leads to a linear feasibility problem and review the formulations of the *sequential row-action relaxation method* of Agmon, Motzkin and Schoenberg, called the AMS method, and the *simultaneous projections method* of Cimmino, called the CIM method, see, e.g., Censor [14] or Censor and Zenios [21, chapter 11]. Assume that the three-dimensional (3D) volume of interest includes Q pre-identified target regions, denoted by $\{T_q \mid q = 1, 2, \ldots, Q\}$, for radiation treatment and that the lower bounds for the required dose to be deposited in target region T_q is t_q. The volume of interest also includes S pre-identified critical organs, denoted by $\{C_s \mid s = 1, 2, \ldots, S\}$, that should be spared by observing upper bounds of permissible doses c_s in organs C_s. The reminder of the volume constitutes the complimentary tissue, denoted by M, which is allowed to absorb not more then m dose units. This volume of interest is discretized into a Cartesian grid of J voxels (pixels – in two-dimensions) and the voxels are numbered (in an agreed manner) by $j = 1, 2, \ldots, J$. Depending on whether a voxel is inside a target (tumor) or inside a critical organ the total dose absorbed in it must lie above or below the lower or upper prescribed dose bounds, respectively.

The modelling also assumes that the radiation, delivered from outside sources, propagates along lines and that the whole volume of interest is covered by I segments, from which radiation emanates, chosen according to the guidelines, as explained in section 2. The segments are indexed (in an agreed manner) by $i = 1, 2, \ldots, I$, and their intensities x_i, arranged in a vector $x = (x_i)_{i=1}^I \in R^I$ in the I-dimensional Euclidean space R^I, are the unknowns of the problem. Next we assume that a state-of-the-art *forward calculation* program is available which calculates, based on the geometry of the segments and on the known geometry and biology of the structures in the volume of interest, for each segment i and every voxel j, the quantity a_{ij} which is the dose absorbed (uniformly) in voxel j due to radiation of unit intensity emanating from the ith segment.

Then the basic *linear feasibility problem* associated with recovering the segments intensities vector $x = (x_i)_{i=1}^I$ is

$$\sum_{i=1}^{I} a_{ij} x_i \leqslant c_s, \quad \text{for all } j \in C_s,\ s = 1, 2, \ldots, S, \tag{9}$$

$$t_q \leqslant \sum_{i=1}^{I} a_{ij} x_i, \quad \text{for all } j \in T_q,\ q = 1, 2, \ldots, Q, \tag{10}$$

$$\sum_{i=1}^{I} a_{ij} x_i \leqslant m, \quad \text{for all } j \in M, \tag{11}$$

$$x_i \geqslant 0, \quad \text{for all } i = 1, 2, \ldots, I. \tag{12}$$

Such a linear feasibility problem can be rearranged to the general form

$$\sum_{i=1}^{I} a_{ij} x_i \leqslant d_j, \quad \text{for all } j = 1, 2, \ldots, J, \tag{13}$$

which can also be rewritten as

$$\left\langle a^j, x \right\rangle \leqslant d_j, \quad \text{for all } j = 1, 2, \ldots, J, \tag{14}$$

where $a^j = (a_{ij})_{i=1}^{I} \in R^I$ is an I-dimensional vector and $\langle a^j, x \rangle := \sum_{i=1}^{I} a_{ij} x_i$ is the inner product in R^I. The nonnegativity constraints (12) can be either subsumed in the system (14) or handled separately by any iterative projections algorithm applied to the problem.

The sequential iterative projections method of AMS for this problem (14) is as follows, see, e.g., [21, algorithm 5.4.2]. It can be viewed also as a special case (for half-spaces) of the well-known method of Projections Onto Convex Sets (POCS) (see, e.g., Stark and Yang [44]) which was used in RTTP by Censor, Altschuler and Powlis [15] and by Lee et al. [38]. The POCS method is in itself a sequential realization of the block-iterative projection (BIP) method of Aharoni and Censor [1] which allows variable blocks of constraints to be processed as the algorithm proceeds.

Algorithm 1. The AMS algorithm.

Initialization: $x^0 \in R^I$ is arbitrary.
Iterative step: Given x^k, calculate the next iterate x^{k+1} by the formula

$$x^{k+1} = x^k + \sigma_k a^{j(k)}, \tag{15}$$

where

$$\sigma_k = \min\left(0, \lambda_k \frac{d_{j(k)} - \langle a^{j(k)}, x^k \rangle}{\|a^{j(k)}\|^2}\right), \tag{16}$$

and go back to the beginning of the Iterative step.
Relaxation parameters: λ_k are real numbers such that $\varepsilon \leqslant \lambda_k \leqslant 2 - \varepsilon$, for all $k \geqslant 0$, with some arbitrarily small but fixed $\varepsilon > 0$.
Control sequence: The sequence of indices $\{j(k)\}_{k \geqslant 0}$ is `cyclic`, meaning that $j(k) = k \bmod(J + 1)$, for all $k \geqslant 0$.

The simultaneous projections method of Cimmino for problem (14) is as follows, see, e.g., Bauschke and Borwein [6, example 6.30], or consult [21, algorithm 5.6.2].

Algorithm 2. Cimmino's algorithm (CIM).

Initialization: $x^0 \in R^I$ is arbitrary.
Weights of importance: These are user-chosen positive real numbers $\omega_j > 0$, for all $j = 1, 2, \ldots, J$, with $\sum_{j=1}^{J} \omega_j = 1$.

Iterative step: Given x^k, calculate the next iterate x^{k+1} by the formula

$$x^{k+1} = x^k + \lambda_k \sum_{j=1}^{J} \omega_j \sigma_j\big(x^k\big) a^j, \tag{17}$$

where

$$\sigma_j\big(x^k\big) = \min\left(0, \frac{d_j - \langle a^j, x^k \rangle}{\|a^j\|^2}\right), \tag{18}$$

and go back to the beginning of the Iterative step.

Relaxation parameters: λ_k are real numbers such that $\varepsilon \leqslant \lambda_k \leqslant 2 - \varepsilon$, for all $k \geqslant 0$, with some arbitrarily small but fixed $\varepsilon > 0$.

The main advantage of Cimmino's algorithm (CIM) over the sequential AMS method is that it converges regardless of the consistency of the system of inequalities (14). In the inconsistent case, when there is no solution to the system, the CIM algorithm still generates convergent sequences $\{x^k\}_{k\geqslant 0}$ of segment intensities which converge to a minimum value of a *proximity function* which measures the weighted (according to the chosen weights of importance) sum of the squares of the distances to all violated inequalities of the system. In addition, CIM is an *inherently parallel algorithm,* see, e.g., Butnariu, Censor and Reich [13], whose operations can be performed on a parallel computer. Acceleration of CIM can also be achieved by using its recent modification called *Component Averaging* (CAV) of Censor, Gordon and Gordon [18], see also [19].

5. The algorithm for the least-intensity feasible (LIF) solution

A single linear inequality in (14) (related to a single voxel in the cross-section) defines a set of points in the space R^I of all segments intensities vectors, namely, the half-space

$$L_j = \big\{x \mid \langle a^j, x \rangle \leqslant d_j\big\} \tag{19}$$

and equation (14) calls for finding a feasible point in the intersection $L = \bigcap_{j=1}^{J} L_j$. We propose to steer the solution process of the linear feasibility problem not to just *any* feasible point but to a feasible point which is *closest* to the origin of the space. This means that while not imposing an exogenous merit function, whose use might be debateable, we want to find a feasible vector x of segments' intensities whose Euclidean norm $\|x\|$ will be the smallest possible from amongst all feasible vectors. The physical meaning is that the segments' intensities (energies) will be the smallest possible – subject to the constraints of the problem, and we call such a vector the least-intensity feasible (LIF) solution of the problem. A graphical illustration of such a vector is depicted in figure 3.

We reformulate the problem slightly and put equations (9)–(12) in the form of an *interval linear feasibility problem*

$$l_j \leqslant \langle a^j, x \rangle \leqslant u_j, \quad \text{for all } j = 1, 2, \ldots, J, \tag{20}$$

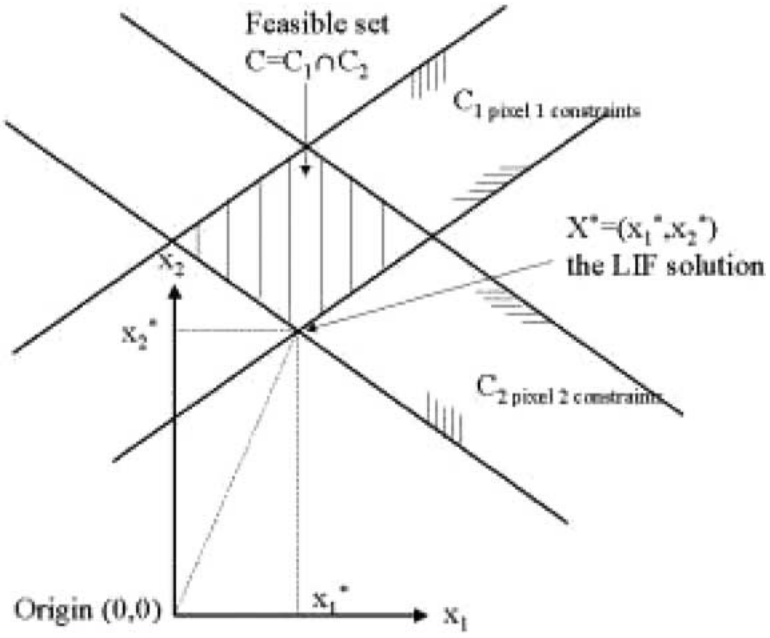


Figure 3. Graphical illustration of the least-intensity feasible (LIF) solution.

where l_j and u_j are lower and upper bounds on the value of the jth inner product (i.e., on the dose absorbed in the jth voxel). We propose to use a novel development of the *primal–dual simultaneous Hildreth projections method* which we now describe. Along with the segments intensities vectors $\{x^k\}_{k\geqslant 0}$ the proposed algorithm keeps track and updates a sequence of *dual vectors* $\{z^k\}_{k\geqslant 0}$, where $z^k = (z_j^k)_{j=1}^J \in R^J$. Before stating the algorithm we define an operation called "mid". Given any triplet of real numbers a, b and c, their mid is that member of the triplet which lies in-between the other two, i.e.,

$$\operatorname{mid}(a,b,c) := \begin{cases} a, & \text{if } b \leqslant a \leqslant c, \text{ or if } c \leqslant a \leqslant b, \\ b, & \text{if } a \leqslant b \leqslant c, \text{ or if } c \leqslant b \leqslant a, \\ c, & \text{if } a \leqslant c \leqslant b, \text{ or if } b \leqslant c \leqslant a. \end{cases} \tag{21}$$

Algorithm 3. The primal–dual algorithm for the LIF solution (LIF).

Initialization: $x^0 \in R^I$ is arbitrary and $z^0 = 0$ is the zero vector in R^J.

Weights of importance: These are user-chosen positive real numbers $\omega_j > 0$, for all $j = 1, 2, \ldots, J$, with $\sum_{j=1}^J \omega_j = 1$.

Iterative step: Given x^k and z^k, calculate the next primal iterate x^{k+1} and the next dual iterate z^{k+1} by doing the following two steps:

(i) Correction factor calculation:

For all $j = 1, 2, \ldots, J$, compute the numbers

$$\alpha_j^k = \frac{u_j - \langle a^j, x^k \rangle}{\|a^j\|^2}, \tag{22}$$

$$\beta_j^k = \frac{l_j - \langle a^j, x^k \rangle}{\|a^j\|^2}, \tag{23}$$

and

$$\gamma_j^k = \text{mid}\left(\frac{z_j^k}{\omega_j}, \alpha_j^k, \beta_j^k\right). \tag{24}$$

(ii) Update of primal and dual vectors:

$$x^{k+1} = x^k + \lambda_k \sum_{j=1}^{J} \omega_j \gamma_j^k a^j, \tag{25}$$

$$z_j^{k+1} = z_j^k - \omega_j \gamma_j^k, \quad \text{for all } j = 1, 2, \ldots, J, \tag{26}$$

and go back to the beginning of the Iterative step.

Relaxation parameters: λ_k are real numbers such that $\varepsilon \leqslant \lambda_k \leqslant 2 - \varepsilon$, for all $k \geqslant 0$, with some arbitrarily small but fixed $\varepsilon > 0$.

This primal–dual algorithm, which we nickname LIF, is simultaneous in nature (like Cimmino's algorithm) and we construct it by merging together two algorithmic structures. The first is the algorithmic structure of Iusem and De Pierro [37] for computing projections on polyhedra which is basically a simultaneous version of Hildreth's sequential algorithm for norm minimization over linear inequalities, see Hildreth [35], Lent and Censor [39] or Censor and Zenios [21, algorithm 6.5.2]. The second structure is that of the sequential, norm-minimizing, image reconstruction algorithm of Herman and Lent [34] called ART4 (Algebraic Reconstruction Technique 4) which handles in a special effective manner interval inequalities. A formal proof of convergence of algorithm 4 will be published elsewhere.

6. Experimental results

We applied both LIF and CIM algorithms to several clinical problems, including treatment planning of prostate cancer and head and neck cancer cases. All algorithmic runs were initiated at zero intensities, i.e., $x^0 = 0$. The total grid numbers for the prostate case presented here are 487 points for the prostate PTV, 1558 and 153 for the adjacent portions of bladder and rectum, respectively. The resolution for the dose calculation is 5 mm/grid. We found that both algorithms converge in approximately the same manner as shown by the convergence plots of norm values versus iteration index. Figure 4 shows the convergence patterns of the algorithms for a typical treatment plan of prostate cancer.

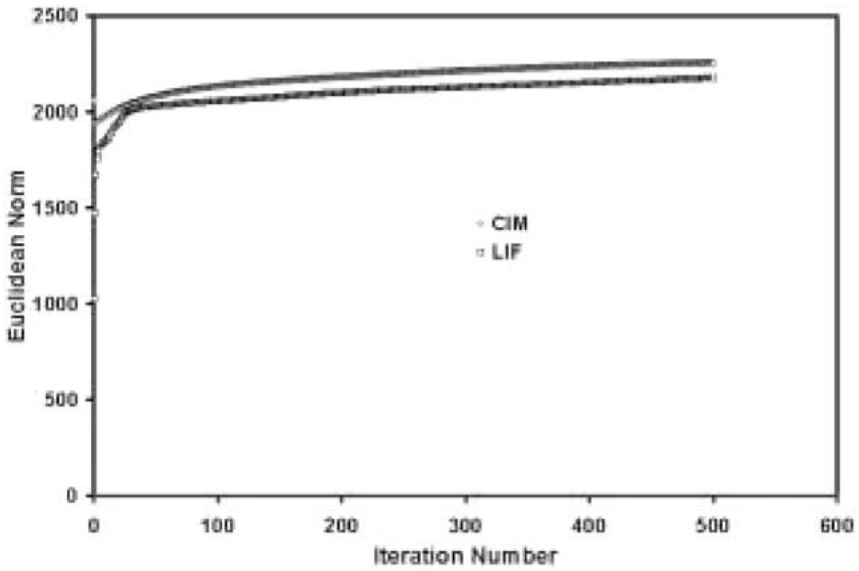


Figure 4. Comparison of Euclidean norm values versus iteration number for the CIM and LIF algorithms in a prostate cancer treatment planning with the ABIP model.

The resulting dose distributions for both algorithms are very similar and the small differences seen are considered clinically insignificant. Figure 5 shows the cumulative dose–volume histogram (% of the volume of a structure receiving a particular dose or less) resulting from the treatment plans with the segments' intensities obtained from these algorithms. For target coverage listed as either the Gross Tumor Volume (GTV) alone or the Planning Target Volume (PTV) that includes various margins to allow for uncertainties in the location of disease and/or daily setup variations, the resulting curves are basically superimposed. For critical healthy structures like the bladder and rectum that are positioned near the target, the dose–volume histograms included in the figure show that doses well below the target dose are achievable. This is the case for both of the algorithms.

7. Conclusion

In this paper we have demonstrated the viability of the ABIP method for IMRT. By starting with external radiation fields that fully conform to the targets and then superimposing some, carefully chosen, sub-fields, the method enables us to use iterative projection algorithms to solve the resulting inverse problem. With such pre-set segments we do not need to apply any further computational process such as is required when the fully discretized beamlet oriented inverse planning approach is taken.

The connection between algorithm 4 and Cimmino's algorithm reveals that the latter's solution (if initialized at the zero vector) may be construed in terms of approaching the intensity vector with the smallest norm (energy) if the problem is feasible. In the

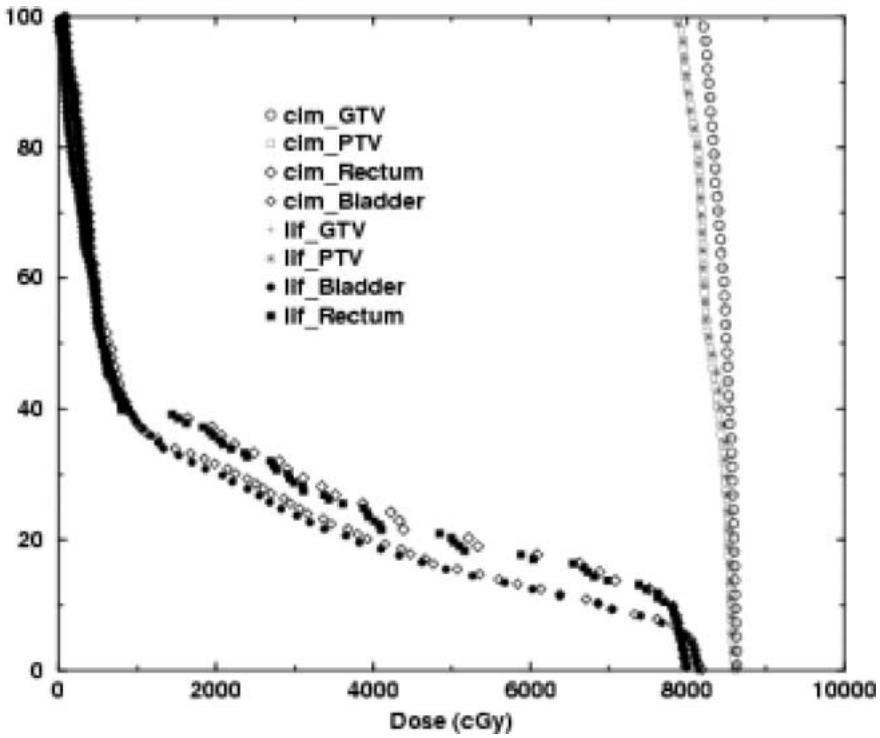


Figure 5. Dose–volume histograms (DVH) comparing LIF and CIM solutions for a treatment plan of a prostate cancer case.

infeasible case the solution approaches the intensity vector with smallest norm amongst the members of the set G of minimizers of the proximity function of (A.8). So, in a way, the solution complies with a least intensity criterion that does not manifest itself but remains hidden in the computations. It turns out that this feature is salutary for the radiotherapy treatment planning problem since it reduces the total beam-on time, thus reducing leakage radiation hazards. Another advantage of Cimmino's algorithm is its inherent parallelism that allows one to implement it on multi-processor systems either in multithreaded environment or on computer clusters.

The modelling with the ABIP method allows the use of a relatively small number of segments and this benefits the treatment itself by improving the accuracy and quality of the treatment plan delivery.

Appendix: Cimmino's algorithm as an approximate LIF solver

The discovery that Cimmino's algorithm 2 always generated in our experimental computational work, when initialized at zero, solutions that were surprisingly good approximations of the LIF solution can be explained and put on firm mathematical ground. To do so we need to look at a class of algorithms which find the projection of a given point onto

the intersection of finitely many given closed and convex sets in the Euclidean space R^I. Such algorithms have recently attracted much attention, see, e.g., Bregman, Censor and Reich [11] and references therein. Let us look at the algorithm studied by Combettes in [25, theorem 5]. The projection of a point $z \in R^I$ onto the half-space L_j is given by

$$P_{L_j}(z) = z + a^j \cdot \min\left(0, \frac{d_j - \langle a^j, z\rangle}{\|a^j\|^2}\right). \tag{A.1}$$

Letting the convex sets be half-spaces of the form (19) and rewriting the algorithm studied by Combettes in our notations yields the following algorithm.

Algorithm 4.

Initialization: $x^0 \in R^I$ is arbitrary.

Weights of importance: These are user-chosen nonnegative real numbers $\omega_j \geqslant 0$, for all $j = 1, 2, \ldots, J$, with $\sum_{j=1}^{J} \omega_j = 1$.

Iterative step: Given x^k, calculate the next iterate x^{k+1} by the formula

$$x^{k+1} = (1 - \alpha_k)x^0 + \alpha_k\left(\lambda \sum_{j=1}^{J} \omega_j \sigma_j\left(x^k\right)a^j + (1 - \lambda)x^k\right), \tag{A.2}$$

where

$$\sigma_j\left(x^k\right) = \min\left(0, \frac{d_j - \langle a^j, x^k\rangle}{\|a^j\|^2}\right), \tag{A.3}$$

and go back to the beginning of the Iterative step.

Relaxation parameter: λ is a real constant such that $0 < \lambda \leqslant 2$, for all $k \geqslant 0$.

Steering parameters: α_k are real numbers such that $\alpha_k \in [0, 1)$ for all $k \geqslant 0$, for which the following conditions hold:

$$\lim_{k\to\infty} \alpha_k = 1, \tag{A.4}$$

$$\sum_{k=0}^{\infty} (1 - \alpha_k) = +\infty, \tag{A.5}$$

and

$$\lim_{k\to\infty} (\alpha_{k+1} - \alpha_k)(1 - \alpha_{k+1})^{-2} = 0. \tag{A.6}$$

We introduce here the term *steering parameters* for the elements of the sequence $\{\alpha_k\}_{k\geqslant 0}$ (the term was not used in [25] or elsewhere in this context) to distinguish them from the relaxation parameters and to reflect the effect of these parameters on the behavior of the sequence of iterates $\{x^k\}_{k\geqslant 0}$ generated by this algorithm. Although the user is free to choose them in the interval [0, 1), they are not otherwise free (like relaxation parameters are) but have to fulfill the three conditions (A.4)–(A.6). Condition (A.4), in

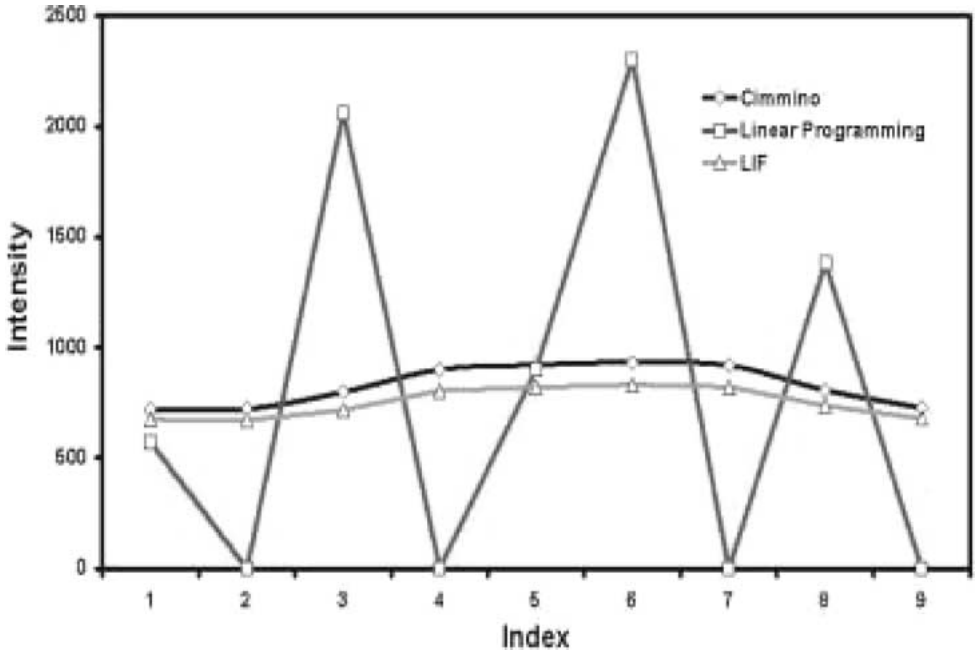


Figure 6. Comparison of intensities for 9 segments (indexed from 1 to 9) of one beam obtained by Ciminno's algorithm, by the LIF algorithm and by a linear programming package demonstrate the greater smoothness of Cimmino's solution and the LIF solution.

particular, steers the next iterate further away from x^0 as the iterations proceed. Algorithm 4 is a simultaneous projection algorithm which, according to Combettes [25, theorem 5], has the property that every sequence $\{x^k\}_{k\geqslant 0}$ generated by it converges (in R^I) to the projection of x^0 onto the intersection $\bigcap_{j=1}^{J} L_j$ if this intersection is nonempty. Otherwise, it converges to the projection of x^0 onto the set

$$G = \left\{x \mid \Phi(x) \leqslant \Phi(y), \text{ for all } y \in R^I\right\}, \tag{A.7}$$

of minimizers of the *proximity function*

$$\Phi(y) := \frac{1}{2}\sum_{j=1}^{J} \omega_j d(y, L_j)^2 \tag{A.8}$$

where the Euclidean distance between a point y and the half-space L_j is given by $d(y, L_j) = \|y - P_{L_j}(y)\|$ and ω_j are the weights of importance used in the algorithm. This set G is actually the set of all weighted least-squares solutions in case the intersection $\bigcap_{j=1}^{J} L_j$ is empty.

The technical condition in Combettes' theorem 5 requiring that one of the closed convex sets to which the algorithm is applied is bounded, is present only to guarantee that G is not empty (see [25, proposition 7]). In our case of half-spaces no set is bounded (by the very nature of half-spaces), so we have to assume that G is non-empty.

The surprising fact that the well-known (and often-used in this field) iterative algorithm of Cimmino generated in our experiments a solution that was repeatedly a very close approximation to the LIF solution can be explained in the light of algorithm 4 as follows. When we solve the fully-discretized model or the ABIP problem we invariably initialize the algorithm with $x^0 = 0$. Using this initial point in algorithm 4 and a relaxation parameter $\lambda = 2$, and applying algorithm 4 with any permissible sequence $\{\alpha_k\}_{k\geqslant 0}$ shows that in the iterative step (A.2) only the second summand remains (because $x^0 = 0$) and the closer α_k gets to the value one (by (A.4)) the closer the whole expression of (A.2) gets to the formula (17) of Cimmino's algorithm 2. For the history of algorithm 4 and further references consult Deutsch and Yamada [28].

The fact that Cimmino's algorithm yields a good approximation of the least-intensity feasible vector is a very desirable feature. Another desirable property is demonstrated by the observation that the Cimmino algorithm also generates smooth intensity distributions in the process of solving inverse planning problems. Irregularity of the intensity vector, characterized by large variations of neighboring segments' intensities has the disadvantage of being generally more difficult to deliver in practice with sufficient accuracy and that it is more vulnerable to patient positioning inaccuracy and other positioning uncertainties. Smooth segments' intensity patterns are closer to the conventional open or wedged beams, therefore, more easily acceptable and implementable in the clinical environment. Figure 6 shows the smooth pattern of the vector obtained from Cimmino's algorithm as compared with that from the SIMPLEX method of linear programming approach using the "phase one" of the algorithm for finding a feasible point.

Acknowledgments

We are indebted to two anonymous referees whose comments and suggestions helped us to improve the first version of this paper. The results presented here were announced at the 43rd Annual Meeting of the American Association of Physicists in Medicine (AAPM), July 2001, Salt Lake City, Utah, USA. This work was supported in part by Elekta Oncology Systems Inc. The work of Y. Censor was supported by research grant 592/00 from the Israel Science Foundation founded by the Israel Academy of Sciences and Humanities.

References

[1] R. Aharoni and Y. Censor, Block-iterative projection methods for parallel computation of solutions to convex feasibility problems, Linear Algebra and Its Applications 120 (1989) 165–175.

[2] M.D. Altschuler and Y. Censor, Feasibility solutions in radiation therapy treatment planning, in: *Proceedings of the Eighth International Conference on the Use of Computers in Radiation Therapy* (IEEE Computer Society Press, Silver Spring, MD, 1984) pp. 220–224.

[3] M.D. Altschuler, Y. Censor and W.D. Powlis, Feasibility and optimization methods in teletherapy planning, in: *Advances in Radiation Oncology Physics*, ed. J.A. Purdy, Medical Physics Monograph No. 19 (American Association of Physicists in Medicine (AAPM), American Institute of Physics, New York, 1992) pp. 1022–1057.

[4] M.D. Altschuler, W.D. Powlis and Y. Censor, Teletherapy treatment planning with physician requirements included in the calculation: I. Concepts and methodology, in: *Optimization of Cancer Radiotherapy*, eds. B.R. Paliwal, D.E. Herbert and C.G. Orton (American Institute of Physics, New York, 1985) pp. 443–452.
[5] H.H. Bauschke, The approximation of fixed points of compositions of nonexpansive mappings in Hilbert space, Journal of Mathematical Analysis and Applications 202 (1996) 150–159.
[6] H.H. Bauschke and J.M. Borwein, On projection algorithms for solving convex feasibility problems, SIAM Review 38 (1996) 367–426.
[7] H.H. Bauschke, J.M. Borwein and A.S. Lewis, The method of cyclic projections for closed convex sets in Hilbert space, Contemporary Mathematics 204 (1997) 1–38.
[8] H.H. Bauschke and P.L. Combettes, A weak-to-strong convergence principle for Fejér-monotone methods in Hilbert spaces, Mathematics of Operations Research 26 (2001) 248–264.
[9] L.M. Bregman, The method of successive projections for finding a common point of convex sets, Soviet Mathematics Doklady 6 (1965) 688–692.
[10] L.M. Bregman, The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming, USSR Computational Mathematics and Mathematical Physics 7 (1967) 200–217.
[11] L.M. Bregman, Y. Censor and S. Reich, Dykstra's algorithm as the nonlinear extension of Bregman's optimization method, Journal of Convex Analysis 6 (1999) 319–333.
[12] D. Butnariu and Y. Censor, Strong convergence of almost simultaneous block-iterative projection methods in Hilbert spaces, Journal of Computational and Applied Mathematics 53 (1994) 33–42.
[13] D. Butnariu, Y. Censor and S. Reich, eds., *Inherently Parallel Algorithms in Feasibility and Optimization and Their Applications*, Studies in Computational Mathematics, Vol. 8 (Elsevier Science, Amsterdam, 2001).
[14] Y. Censor, Mathematical aspects of radiation therapy treatment planning: continuous inversion versus full discretization and optimization versus feasibility, in: *Computational Radiology and Imaging: Therapy and Diagnostics*, eds. C. Börgers and F. Natterer, The IMA Volumes in Mathematics and Its Applications, Vol. 110 (Springer, New York, 1999) pp. 101–112.
[15] Y. Censor, M.D. Altschuler and W.D. Powlis, A computational solution of the inverse problem in radiation therapy treatment planning, Applied Mathematics and Computation 25 (1988) 57–87.
[16] Y. Censor, M.D. Altschuler and W.D. Powlis, On the use of Cimmino's simultaneous projections method for computing a solution of the inverse problem in radiation therapy treatment planning, Inverse Problems 4 (1988) 607–623.
[17] Y. Censor, T. Elfving and G.T. Herman, Averaging strings of sequential iterations for convex feasibility problems, in: *Inherently Parallel Algorithms in Feasibility and Optimization and Their Applications*, eds. D. Butnariu, Y. Censor and S. Reich (Elsevier Science, Amsterdam, 2001) pp. 101–114.
[18] Y. Censor, D. Gordon and R. Gordon, Component averaging: An efficient iterative parallel algorithm for large and sparse unstructured problems, Parallel Computing 27 (2001) 777–808.
[19] Y. Censor, D. Gordon and R. Gordon, BICAV: A block-iterative, parallel algorithm for sparse systems with pixel-dependent weighting, IEEE Transactions on Medical Imaging 20 (2001) 1050–1060.
[20] Y. Censor, W.D. Powlis and M.D. Altschuler, On the fully discretized model for the inverse problem of radiation therapy treatment planning, in: *Proceedings of the Thirteenth Annual Northeast Bioengineering Conference*, Vol. 1, ed. K.R. Foster (IEEE, New York, 1987) pp. 211–214.
[21] Y. Censor and S.A. Zenios, *Parallel Optimization: Theory, Algorithms, and Applications* (Oxford University Press, New York, 1997).
[22] P.S. Cho and R.J. Marks II, Hardware-sensitive optimization for intensity modulated radiotherapy, Physics in Medicine and Biology 45 (2000) 429–440.
[23] G. Cimmino, Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari, La Ricerca Scientifica XVI Series II 9(1) (1938) 326–333.

[24] P.L. Combettes, The foundations of set-theoretic estimation, Proceedings of the IEEE 81 (1993) 182–208.
[25] P.L. Combettes, Inconsistent signal feasibility problems: Least-squares solutions in a product space, IEEE Transactions on Signal Processing 42 (1994) 2955–2966.
[26] P.L. Combettes, The convex feasibility problem in image recovery, Advances in Imaging and Electron Physics 95 (1996) 155–270.
[27] F. Deutsch, *Best Approximation in Inner Product Spaces* (Springer, New York, 2001).
[28] F. Deutsch and I. Yamada, Minimizing certain convex functions over the intersection of the fixed point sets of nonexpansive mappings, Numerical Functional Analysis and Optimization 19 (1998) 33–56.
[29] P.P.B. Eggermont, G.T. Herman and A. Lent, Iterative algorithms for large partitioned linear systems, with applications to image reconstruction, Linear Algebra and Its Applications 40 (1981) 37–67.
[30] T. Elfving, Block-iterative methods for consistent and inconsistent linear equations, Numerische Mathematik 35 (1980) 1–12.
[31] J.M. Galvin, Y. Xiao, D. Michalski, Y. Censor, C. Houser, Y. Chen, G. Bednarz, P.R. Anni, S. Huq and W.J. Curran, Treating oropharyngeal cancer with an aperture-based inverse planning method, Technical Report (September 27, 2001).
[32] L. Gubin, B. Polyak and E. Raik, The method of projections for finding the common point of convex sets, USSR Computational Mathematics and Mathematical Physics 7 (1967) 1–24.
[33] G.T. Herman, *Image Reconstruction From Projections: The Fundamentals of Computerized Tomography* (Academic Press, New York, 1980).
[34] G.T. Herman and A. Lent, A family of iterative quadratic optimization algorithms for pairs of inequalities, with applications in diagnostic radiology, Mathematical Programming Study 9 (1978) 15–29.
[35] C. Hildreth, A quadratic programming procedure, Naval Research Logistics Quarterly 4 (1957) 79–85. Erratum, ibid., p. 361.
[36] A.N. Iusem and A.R. De Pierro, Convergence results for an accelerated nonlinear Cimmino algorithm, Numerische Mathematik 49 (1986) 367–378.
[37] A.N. Iusem and A.R. De Pierro, A simultaneous iterative method for computing projections on polyhedra, SIAM Journal on Control and Optimization 25 (1987) 231–243.
[38] S. Lee, P.S. Cho, R.J. Marks II and S. Oh, Conformal radiotherapy computation by the method of alternating projections onto convex sets, Physics in Medicine and Biology 42 (1997) 1065–1086.
[39] A. Lent and Y. Censor, Extensions of Hildreth's row-action method for quadratic programming, SIAM Journal on Control and Optimization 18 (1980) 444–454.
[40] L.D. Marks, W. Sinkler and E. Landree, A feasible set approach to the crystallographic phase problem, Acta Crystallographica A 55 (1999) 601–612.
[41] W.D. Powlis, M.D. Altschuler and Y. Censor, Teletherapy treatment planning with physician requirements included in the calculation: II. Clinical applications, in: *Optimization of Cancer Radiotherapy*, eds. B.R. Paliwal, D.E. Herbert and C.G. Orton (American Institute of Physics, New York, 1985) pp. 453–461.
[42] W.D. Powlis, M.D. Altschuler, Y. Censor and E.L. Buhle, Semi-automated radiotherapy treatment planning with a mathematical model to satisfy treatment goals, International Journal of Radiation Oncology, Biology and Physics 16 (1989) 271–276.
[43] D.M. Shepard, M.C. Ferris, G.H. Olivera and T.R. Mackie, Optimizing the delivery of radiation therapy to cancer patients, SIAM Review 41 (1999) 721–744.
[44] H. Stark and Y. Yang, *Vector Space Projections: A Numerical Approach to Signal and Image Processing, Neural Nets, and Optics* (Wiley, New York, 1998).
[45] Y. Xiao, J. Galvin, M. Hossain and R. Valicenti, An optimized forward-planning technique for intensity modulated radiation therapy, Medical Physics 27 (2000) 2093–2099.
[46] D.C. Youla, Mathematical theory of image restoration by the method of convex projections, in: *Image Recovery: Theory and Applications*, ed. H. Stark (Academic Press, Orlando, FL, 1987) pp. 29–77.