A new algorithm for optimal multileaf collimator field segmentation

Konrad Engel
Universität Rostock
Fachbereich Mathematik
D–18051 Rostock
Germany

March, 2003

Abstract

We present a new efficient leaf sequencing algorithm for the generation of intensity maps by a nonnegative combination of segments. Intensity maps describe the intensity modulation of beams in radiotherapy. We only study the static case (stop and shoot) an optimize the total number of monitor units and the number of segments. We will present a short exact proof for a formula giving the smallest total number of monitor units and describe a class of algorithms yielding this minimal value. A special member of this class provides in addition a solution with a very small number of segments.

Key words: leaf sequencing, radiation therapy optimization, intensity modulation, multileaf collimator, IMRT

1 Introduction

Modern radiotherapy planning algorithms are composed of several procedures. There are two essential steps. In the first step, a small number of intensity modulated fields are determined with the aim that the planning target area homogenously receives a fixed dose and that critical structures are protected as well as possible. Here a field depends on several parameters like position of the isocenter, field breadth, field length, energy, collimator rotation, gantry angle, table angle, kind of wedge. Moreover, the intensity of the beam through the rectangle given by field breadth and field length is not homogenous but modulated, i.e., the fluence depends on the point of the rectangle. After discretization this intensity modulation is described by an intensity map which is mathematically an $m \times n$ matrix A with nonnegative entries. There are several methods of realizing such an intensity map, cf. Brahme [4], but a modern way is the usage of a multileaf collimator (MLC) that, by means of its leaves, opens and closes certain regions of the rectangle. Several leaf–positions, called segments, must be superposed in order to realize the intensity map. The second essential step in planning algorithms is the determination of a small number of segments (with monitor units) which realize the intensity map in short time.

In this paper we present a new algorithm for this second step. We only study the static case (stop and shoot) and optimize the total number of monitor units (the total relative fluence, the total shooting–time) and the number of segments. If the leaves can be shifted very quickly these two objectives are essential. We will present a short exact proof for a formula giving the smallest total number of monitor units (TNMU) and describe a class of algorithms yielding this minimal value. A special member of this class provides in addition a solution with a very small number of segments (NS). Starting with Galvin et al [6] and Bortfeld et al [2] several algorithms have been designed [5, 8, 10, 11, 13]. The Bortfeld–Boyer–algorithm provides the smallest possible TNMU but a large NS. Other algorithms aim to reduce the NS at the price of an increased TNMU. Like many problems in combinatorial optimization, the leaf–segmentation problem can be formulated as an integer programming problem, see Langer et al [8]. Thus, in principle, the TNMU and the NS can be optimized simultaneously. But an integer programming solver like CPLEX may help only in the case of small problem size, see [8, p. 2457].

Our algorithm is optimal for the TNMU and approximative optimal for the NS. In comparison with the other published algorithms it provides better solutions and does not essentially depend on the entries of the intensity map.

In principle, each entry may be a nonnegative real number. On the computer, we realize them as natural numbers, e.g. from $\{0, \ldots, 10,000\}$.

2 Mathematical formulation and solution of the TNMU–segmentation problem

If not stated otherwise, let all matrices be of dimension $m \times n$. Let $[n] := \{1, \ldots, n\}$. A subset I of $[n]$ is called *interval* if there are numbers $l, r \in [n]$ such that $I = \{x \in [n] : l \leq x \leq r\}$. Note that we allow $l > r$, i.e. $I = \emptyset$. We denote I by $[l, r]$. A matrix S is called *segment* if there is an m–tuple $\boldsymbol{I} = (I_1, \ldots, I_m)$ of intervals such that

$$s_{ij} = \begin{cases} 1 & \text{if } j \in I_i \\ 0 & \text{otherwise} \end{cases} \qquad i \in [m], j \in [n].$$

The interval I_i can be considered as the region which remains open by the i-th pair of leaves of the MLC. A *segmentation* of a matrix A is a decomposition of A into a nonnegative combination of segments:

$$A = \sum_k u_k S_k,$$

where $u_k \geq 0$. The *TNMU of the segmentation* is defined to be

$$U = \sum_k u_k.$$

The *TNMU–segmentation problem* is the following: Let A be a nonnegative matrix, i.e. a matrix with nonnegative entries. Find a segmentation such that its TNMU is minimum!

As an example we consider a segmentation of a benchmark–matrix A (from [3, 8]) with 6 segments and a TMNU of $U = 10$:

$$\begin{pmatrix}4&5&0&1&4&5\\2&4&1&3&1&4\\2&3&2&1&2&4\\5&3&3&2&5&3\end{pmatrix} = 4\begin{pmatrix}1&1&0&0&0&0\\0&0&0&0&0&1\\0&0&0&0&0&1\\1&0&0&0&0&0\end{pmatrix} + 2\begin{pmatrix}0&0&0&0&1&1\\0&0&0&1&0&0\\1&1&1&0&0&0\\0&1&1&1&1&0\end{pmatrix}$$
$$+1\begin{pmatrix}0&0&0&1&1&1\\1&1&1&1&1&0\\0&0&0&1&1&0\\1&1&1&0&0&0\end{pmatrix} + 1\begin{pmatrix}0&0&0&0&1&1\\1&1&0&0&0&0\\0&1&0&0&0&0\\0&0&0&0&1&1\end{pmatrix}$$
$$+1\begin{pmatrix}0&1&0&0&0&0\\0&1&0&0&0&0\\0&0&0&0&1&0\\0&0&0&0&1&1\end{pmatrix} + 1\begin{pmatrix}0&0&0&0&0&1\\0&1&0&0&0&0\\0&0&0&0&0&0\\0&0&0&0&1&1\end{pmatrix}$$

In order to avoid case distinctions, we add two zero–columns to A, i.e. let

$$a_{i,0} = a_{i,n+1} = 0 \text{ for all } i \in [m].$$

With A we associate its *difference matrix* D of dimension $m \times (n+1)$:

$$d_{ij} = a_{ij} - a_{i,j-1}, \qquad i \in [m], j \in [n+1].$$

The *TNMU–row–complexity* $c_i(A)$ of A is defined by

$$c_i(A) = \sum_{j=1}^{n+1} \max\{0, d_{ij}\}$$

and the TNMU–complexity $c(A)$ of A is the maximum TNMU–row–complexity:

$$c(A) = \max_{j\in[n]} c_i(A).$$

The following result is essential for our algorithm. Related results for the dynamic case with $m = 1$ were proved, in a more physical way and under the condition that the leaves are shifted only from left to right during the whole process, by Stein et al [12] and Ma et al [9].

Theorem 1. *The TNMU–complexity of A equals the smallest TNMU of a segmentation of A.*

Proof. We suppose that A is not the zero matrix because otherwise no segmentation is necessary. The proof consists of two parts. In the first part we show that the TNMU of a segmentation cannot be smaller than $c(A)$. In the second part we show that $c(A)$ can be realized.

For the first part, let i^* be an index of greatest TNMU–row–complexity, i.e.

$$c_{i^*}(A) = c(A).$$

Let

$$P = \{j \in [n] : d_{i^*,j} \geq 0 \text{ and } d_{i^*,j+1} < 0\},$$
$$M = \{j \in [n] : d_{i^*,j} < 0 \text{ and } d_{i^*,j+1} \geq 0\}.$$

The elements of P determine positions of local maxima (on a plateau on the right) and the elements of M determine positions of local minima (on a plateau on the left) of the sequence $(a_{i^*,1}, a_{i^*,2}, \ldots, a_{i^*,n})$. Note that, going from 1 to n, one meets in an alternating way elements of P and M, the first and the last element are elements from P. Hence, if $I \subseteq [n]$ is an interval, then $|I \cap P| - |I \cap M| \in \{-1, 0, 1\}$ and consequently, for any segment S,

$$\sum_{j \in P} s_{i^*,j} - \sum_{j \in M} s_{i^*,j} \leq 1. \tag{1}$$

Let $P = \{p_1, \ldots, p_{l+1}\}$, $M = \{m_1, \ldots, m_l\}$, and

$$1 \leq p_1 < m_1 < p_2 < m_2 < \cdots < m_l < p_{l+1} \leq n.$$

Let, in addition, $m_0 = 0$. We have

$$\begin{aligned}
\sum_{j=1}^{n} \max\{0, d_{i^*,j}\} &= \sum_{k=1}^{l+1} \sum_{j=m_{k-1}+1}^{p_k} d_{i^*,j} \\
&= \sum_{k=1}^{l+1} ((a_{i^*,m_{k-1}+1} - a_{i^*,m_{k-1}}) + \cdots + (a_{i^*,p_k} - a_{i^*,p_k-1})) \\
&= \sum_{k=1}^{l+1} (a_{i^*,p_k} - a_{i^*,m_{k-1}}),
\end{aligned}$$

i.e.

$$c(A) = \sum_{j \in P} a_{i^*,j} - \sum_{j \in M} a_{i^*,j}. \tag{2}$$

Now let

$$A = \sum_k u_k S_k$$

be any segmentation of S and let $s_{ij}^{(k)}$ be the entries of S_k. Then by (1) and (2)

$$\begin{aligned}
\sum_k u_k &= \sum_k 1 \cdot u_k \\
&\geq \sum_k \left(\sum_{j \in P} s_{i^*,j}^{(k)} - \sum_{j \in M} s_{i^*,j}^{(k)} \right) u_k \\
&= \sum_{j \in P} \sum_k u_k s_{i*,j}^{(k)} - \sum_{j \in M} \sum_k u_k s_{i*,j}^{(k)} \\
&= \sum_{j \in P} a_{i^*,j} - \sum_{j \in M} a_{i^*,j} \\
&= c(A).
\end{aligned}$$

For the second part we present an algorithm for the realization of $c(A)$ which will turn out to be almost optimal concerning the NS. First we describe one step of the algorithm. Let

$$u_1 := \min\{a_{ij} : a_{ij} > 0, i \in [m], j \in [n]\}$$

be the minimum of the nonzero elements of A. Moreover, let $\boldsymbol{I} = (I_1, \ldots, I_m)$ (with $I_i = [l_i, r_i]$ for all i) be such an m-tuple of intervals such that

$$\begin{aligned}
I_i &\neq \emptyset && \text{if } c_i(A) > 0, \\
a_{ij} &> 0 && \text{for } j \in I_i, \\
a_{i,l_i-1} = a_{i,r_i+1} &= 0 && \text{if } I_i \neq \emptyset, \\
a_{ij} &= u_1 && \text{for some } i \in [m], j \in I_i.
\end{aligned}$$

Let S_1 be the segment associated with $\boldsymbol{I}$ and let

$$A' := A - u_1 S_1$$

(again with $a'_{i,0} = a'_{i,n+1} = 0$ for all i). By construction, all entries of A' are nonnegative. It is easy to see that, for the differences

$$d'_{ij} := a'_{ij} - a'_{i,j-1}, \qquad i \in [m], j \in [n+1],$$

we have

$$d'_{ij} = \begin{cases} d_{ij} - u_1 \geq 0 & \text{if } j = l_i \text{ and } I_i \neq \emptyset, \\ d_{ij} + u_1 \leq 0 & \text{if } j = r_i + 1 \text{ and } I_i \neq \emptyset, \\ d_{ij} & \text{otherwise.} \end{cases} \tag{3}$$

Hence, for all $i \in [m]$,

$$c_i(A') = \begin{cases} c_i(A) - u_1 & \text{if } I_i \neq \emptyset, \\ c_i(A) = 0 & \text{otherwise,} \end{cases}$$

and consequently,

$$c(A') = c(A) - u_1 \geq 0 \qquad \text{if } c(A) > 0.$$

Here the first step is finished and we may continue in the same way. The whole algorithm terminates if the zero matrix $\boldsymbol{O}$ is obtained. The algorithm provides a sequence $(u_1, S_1), (u_2, S_2), \ldots$ such that

$$\begin{aligned} \boldsymbol{O} &= A - u_1 S_1 - u_2 S_2 - \cdots , \\ 0 = c(\boldsymbol{O}) &= c(A) - u_1 - u_2 - \cdots , \end{aligned}$$

i.e. a segmentation with the TNMU $c(A)$ is obtained.

By construction, at least one non–zero entry becomes zero in each step. Hence, after at most mn steps the zero matrix is obtained which proves in particular that termination is after finite time. □

It is easy to see that each step can be realized in algorithmic time–complexity $O(mn)$, hence the whole algorithm has time complexity $O(m^2n^2)$. The algorithm (not requiring the integrality of the entries of A) is near to the algorithm of Bortfeld et al which also yields the TNMU $c(A)$ (the proof is similar). One may choose the intervals I_i a little bit more precisely: Choose I_i such that the set $\{a_{ij} : j \in I_i\}$ contains as much as possible elements equal to the minimal nonzero element of the actual matrix. Then often not only one, but several nonzero elements become zero in one step.

Now we show that the algorithm is almost optimal w.r.t. the number of segments.

Theorem 2. *Let the elements a_{ij} of A be realizations of the coordinates of a continuous mn–dimensional random vector $(X_{11}, \ldots, X_{1n}, X_{21}, \ldots, X_{mn})$. Then the probability that there is a segmentation of A with fewer than mn segments equals 0.*

Proof. There is a huge, but finite number of choices of less than mn segments. Fix some choice $\{S_1, \ldots, S_l\}$, $l < mn$. These segments span a vector space $\langle S_1, \ldots, S_l \rangle$ of dimension at most $mn - 1$ which has zero measure in the mn–dimensional space. Hence, the probability that A belongs to $\langle S_1, \ldots, S_l \rangle$

equals zero which implies that the probability of a segmentation

$$A = \sum_{k=1}^{l} u_k S_k$$

also equals zero. The probability that A can be segmented with less than mn segments is not greater than the sum of the probabilities described above, extended over all choices of less than mn–segments, i.e. not greater than a finite sum of zeros. □

3 Minimizing the number of segments

In practice, the elements of A are not continuously distributed numbers, but numbers from a discrete distribution which can be encoded e.g. by natural numbers from $\{0, \ldots, 10,000\}$. Then the single probabilities from the proof of Theorem 2 are not zero, but a small positive number. By the huge number of choices of segments the probability that A can be segmented with essentially fewer than mn segments dramatically increases. So, in practice, there is still need to have algorithms providing a small NS and a minimal TNMU. As in the proof of Theorem 1, one can see that the following class of algorithms always leads to the minimum TNMU:

General minimum TNMU–algorithm:

while $A \neq \boldsymbol{O}$

find a coefficient $u > 0$ and a segment S such that

$$A' = A - uS \text{ is nonnegative,} \tag{4}$$

$$c(A') = c(A) - u; \tag{5}$$

output (u, S);

$A := A - uS$.

We call a pair (u, S) of a positive number $u > 0$ and a segment S an *admissible segmentation pair* if conditions (4) and (5) are satisfied. Note that, in the proof of Theorem 1, we constructed an admissible segmentation pair (u, S) where $u = u_1$ is the smallest positive entry of A, i.e. $u > 0$.

First we study the more difficult condition (5). Let S be given by the m–tuple $\boldsymbol{I} = (I_1, \ldots, I_m)$ and let $I_i = [l_i, r_i]$.

Lemma 3. *Let i^* be such an index for which the maximum TNMU–row–complexity is attained, i.e. $c_{i^*}(A) = c(A)$. Then $c_{i^*}(A') \geq c(A) - u$.*

Proof. If $I_{i^*} = \emptyset$ we have $c_{i^*}(A') = c_{i^*}(A) \geq c(A) - u$. Let $I_{i^*} \neq \emptyset$. Recall that by (3) the differences d_{i,l_i} and d_{i,r_i+1} are the only differences which alter. Hence we have

$$\begin{aligned} c_{i^*}(A') = c_{i^*}(A) &+ (\max\{0, d_{i^*,l_{i^*}} - u\} - \max\{0, d_{i^*,l_{i^*}}\}) \\ &+ (\max\{0, d_{i^*,r_{i^*+1}} + u\} - \max\{0, d_{i^*,r_{i^*+1}}\}) \\ \geq c_{i^*}(A) &- u + 0 \\ = \quad c(A) &- u. \end{aligned}$$

□

Lemma 4. *We have $c(A') = c(A) - u$ iff for all $i \in [m]$:*

$$u \leq c(A) - c_i(A) \text{ if } I_i = \emptyset, \tag{6}$$

$$\begin{aligned} &\max\{0, d_{i,l_i} - u\} + \max\{0, d_{i,r_i+1} + u\} + u \\ &\qquad \leq c(A) - c_i(A) + \max\{0, d_{i,l_i}\} + \max\{0, d_{i,r_i+1}\} \text{ if } I_i \neq \emptyset. \end{aligned} \tag{7}$$

Proof. If $I_i = \emptyset$, $c_i(A') = c_i(A)$ and if $I_i \neq \emptyset$, again by (3),

$$\begin{aligned} c_i(A') = c_i(A) - \max\{0, d_{i,l_i}\} &- \max\{0, d_{i,r_i+1}\} \\ &+ \max\{0, d_{i,l_i} - u\} + \max\{0, d_{i,r_i+1} + u\}. \end{aligned}$$

So both conditions (6) and (7) are equivalent to

$$c_i(A') \leq c(A) - u. \tag{8}$$

Clearly, $c(A') = c(A) - u$ implies $c_i(A') \leq c(A) - u$ for all $i \in [m]$. Conversely, if $c_i(A') \leq c(A) - u$ for all $i \in [m]$, then $c(A') \leq c(A) - u$. The inequality $c(A') \geq c(A) - u$ follows from Lemma 3. □

Now we show that u and the intervals I_i may be maximal in a specific way.

Lemma 5. *If (6) resp. (7) are satisfied for I_i and u, and if $u' \leq u$, then these inequalities are satisfied for I_i and u', too.*

Proof. For (6) the proof is trivial and for (7) one has to observe that the functions $\max\{0, d_{i,r_i+1} + u\}$ as well as $\max\{0, d_{i,l_i} - u\} + u$ are both non–decreasing in u. □

Lemma 6. *Let (7) be satisfied for the interval $I_i = [l_i, r_i]$ and for u.*
a) If $d_{i,l_i} \leq 0$, then (7) is also satisfied for the interval $I'_i = [l_i - 1, r_i]$ and for u.
b) If $d_{i,r_i+1} \geq 0$, then (7) is also satisfied for the interval $I'_i = [l_i, r_i + 1]$ and for u.

Proof. a) The proof follows from the simple relations

$$\max\{0, d_{i,l_i} - u\} - \max\{0, d_{i,l_i}\} = 0,$$
$$\max\{0, d_{i,l_i-1} - u\} - \max\{0, d_{i,l_i-1}\} \leq 0.$$

b) analogously. □

Hence it makes sense to choose segments S in such a way that $d_{i,l_i} > 0$ and $d_{i,r_i+1} < 0$ for all i. We call an interval $I_i = [l_i, r_i]$ *essential* if $I_i = \emptyset$ or if $I_i \neq \emptyset$ and $d_{i,l_i} > 0$ and $d_{i,r_i+1} < 0$. Now we specify the choice of u:

Main strategy for the choice of u:

Take the largest possible u, i.e. the greatest number $u = u_{max}$ for which there exists a segment S such that (u, S) is an admissible segmentation pair.

We call this number u_{max} the *maximum MU–number*. Note that u_{max} indeed exists (and that $u_{max} > 0$) since there is an admissible segmentation pair (u, S) (with $u > 0$), see again the proof of Theorem 1. In order to find u_{max} we first determine for each essential interval $I_i = [l_i, r_i]$ the greatest number v_{I_i} such that the inequality (6) resp. (7) is satisfied with $u = v_{I_i}$. For brevity, we define the *row–complexity–gap* $g_i(A)$ by

$$g_i(A) = c(A) - c_i(A), i \in [m].$$

Lemma 7. *a) If $I_i = \emptyset$ then $v_{I_i} = g_i(A)$.*
b) If $I_i \neq \emptyset$ then

$$v_{I_i} = \begin{cases} \min\{d_{i,l_i}, -d_{i,r_i+1}\} + g_i(A) & \text{if } g_i(A) \leq |d_{i,l_i} + d_{i,r_i+1}|, \\ (d_{i,l_i} - d_{i,r_i+1} + g_i(A))/2 & \text{otherwise.} \end{cases}$$

Proof. a) is trivial, thus we only prove b). Let h_{I_i} be the RHS of the equation in the lemma. We have to prove that $v_{I_i} = h_{I_i}$.

Case 1. $g_i(A) \leq |d_{i,l_i} + d_{i,r_i+1}|$. Using the equality $|\alpha - \beta| = \max\{\alpha, \beta\} - \min\{\alpha, \beta\}$ we obtain

$$h_{I_i} \leq \min\{d_{i,l_i}, -d_{i,r_i+1}\} + |d_{i,l_i} + d_{i,r_i+1}| = \max\{d_{i,l_i}, -d_{i,r_i+1}\},$$

i.e.

$$\min\{d_{i,l_i}, -d_{i,r_i+1}\} \le h_{I_i} \le \max\{d_{i,l_i}, -d_{i,r_i+1}\}.$$

A straightforward computation with case distinction $d_{i,l_i} \le -d_{i,r_i+1}$ and $d_{i,l_i} > -d_{i,r_i+1}$ shows that (7) holds with equality for $u = h_{I_i}$: For example, if $d_{i,l_i} \le -d_{i,r_i+1}$, we have

$$\begin{aligned}\max\{0, d_{i,l_i} - h_{I_i}\} &+ \max\{0, d_{i,r_i+1} + h_{I_i}\} + h_{I_i} \\ &= 0 + 0 + h_{I_i} = d_{i,l_i} + g_i(A) \\ &= g_i(A) + \max\{0, d_{i,l_i}\} + \max\{0, d_{i,r_i+1}\}.\end{aligned}$$

Again a straightforward computation shows that (7) does not hold for $u > h_{I_i}$: For example, if $d_{i,l_i} > -d_{i,r_i+1}$, we have

$$\begin{aligned}\max\{0, d_{i,l_i} - u\} &+ \max\{0, d_{i,r_i+1} + u\} + u \\ &\ge (d_{i,l_i} - u) + (d_{i,r_i+1} + u) + u > d_{i,l_i} + d_{i,r_i+1} + (-d_{i,r_i+1}) + g_i(A) \\ &= g_i(A) + \max\{0, d_{i,l_i}\} + \max\{0, d_{i,r_i+1}\}.\end{aligned}$$

Case 2. $g_i(A) > |d_{i,l_i} + d_{i,r_i+1}|$. Using the equality $(\alpha - \beta + |\alpha + \beta|)/2 = \max\{\alpha, -\beta\}$ we obtain

$$h_{I_i} > (d_{i,l_i} - d_{i,r_i+1} + |d_{i,l_i} + d_{i,r_i+1}|)/2 = \max\{d_{i,l_i}, -d_{i,r_i+1}\}.$$

Consequently, for $u \ge h_{I_i}$,

$$\begin{aligned}\max\{0, d_{i,l_i} - u\} &+ \max\{0, d_{i,r_i+1} + u\} + u \\ &= 0 + d_{i,r_i+1} + 2u \ge d_{i,l_i} + g_i(A) \\ &= g_i(A) + \max\{0, d_{i,l_i}\} + \max\{0, d_{i,r_i+1}\},\end{aligned}$$

and (7) holds with equality for $u = h_{I_i}$ and does not hold for $u > h_{I_i}$. $\square$

We emphasize that (6) resp. (7) are satisfied for $u = v_{I_i}$ as equalities. Since these conditions are also equivalent to (8) in case of equality, for essential intervals

$$c_i(A') = c(A) - v_{I_i}. \tag{9}$$

For an interval $I_i = [l_i, r_i]$ let

$$w_{I_i} := \begin{cases} \infty & \text{if } I_i = \emptyset, \\ \min\{a_{ij} : l_i \le j \le r_i\} & \text{if } I_i \ne \emptyset, \end{cases} \tag{10}$$

and

$$u_{I_i} := \min\{v_{I_i}, w_{I_i}\}. \tag{11}$$

Let $\mathcal{I}$ be the set of all essential intervals in $[n]$, $i \in [m]$.

Theorem 8. *Let, for $i \in [m]$, $\hat{I}_i \in \mathcal{I}$ be an interval of maximum value u_{I_i}, i.e. $u_{\hat{I}_i} \geq u_{I_i}$ for all $I_i \in \mathcal{I}$. Then the maximum MU–number is given by*

$$u_{max} = \min_{i \in [m]} u_{\hat{I}_i}.$$

Proof. Let $\hat{u} = \min_{i \in [m]} u_{\hat{I}_i}$. We have to prove that $\hat{u} = u_{max}$.

First we show that $\hat{u} \geq u_{max}$. Assume, that there is an admissible segmentation pair (u, S) such that $u > \hat{u}$. Let S be associated with $(I_1, \dots, I_m)$ and let $A' = A - uS$. By Lemma 6 we may suppose that all intervals $I_i, i \in [m]$, are essential. Since $u > \hat{u}$ there is some index $i \in [m]$ such that $u > u_{\hat{I}_i}$, i.e. $u > u_{I_i}$ and hence $u > v_{I_i}$ or $u > w_{I_i}$. In the first case, by definition of v_{I_i}, the inequality (6) resp. (7) cannot be satisfied, and thus, by Lemma 4, $c(A') \neq c(A) - u$. In the second case, A' has negative elements. Both cases lead to a contradiction, i.e. the assumption was false.

Now we show that $\hat{u} \leq u_{max}$. Let $\hat{S}$ be the segment which is associated with $(\hat{I}_1, \dots, \hat{I}_m)$. It is enough to verify that $(\hat{u}, \hat{S})$ is an admissible segmentation pair. We already know that $\hat{u} \geq u_{max} > 0$. Let again $A' = A - \hat{u}\hat{S}$. By construction,

$$\hat{u} \leq u_{\hat{I}_i} \leq w_{\hat{I}_i} \leq a_{ij} \text{ for all } i \in [m] \text{ and for all } j \text{ with } l_i \leq j \leq r_i.$$

Consequently, A' is nonnegative. Moreover,

$$\hat{u} \leq u_{\hat{I}_i} \leq v_{\hat{I}_i} \text{ for all } i \in [m].$$

By definition of $v_{\hat{I}_i}$, the inequality (6) resp. (7) is satisfied for $v_{\hat{I}_i}$ and hence by Lemma 5 also for $\hat{u}$. From Lemma 4 it follows that $c(A') = c(A) - \hat{u}$. $\square$

With the matrix A, we associate the number

$$q(A) := |\{(i,j) : \min\{a_{ij}, |d_{ij}|\} > 0\}| + |\{i : c_i(A) < c(A)\}|.$$

Clearly, $0 \leq q(A) \leq mn + n - 1$. We call $q(A)$ the *NS–complexity* of A.

Theorem 9. *If $u = u_{max}$ and $S = \hat{S}$ are chosen as in the proof of Theorem 8, then the general minimum TNMU–algorithm needs at most $mn + n - 1$ steps.*

Proof. We will show that $q(A') \leq q(A) - 1$ if $A \neq \boldsymbol{O}$. Then, in the algorithm, after at most $mn + n - 1$ steps $q(A) = 0$ which implies $A = \boldsymbol{O}$.

First note that

$$\min\{a_{ij}, |d_{ij}|\} = 0 \text{ implies } \min\{a'_{ij}, |d'_{ij}|\} = 0$$

because $a'_{ij} \leq a_{ij}$ and d_{i,l_i} and d_{i,r_i+1} are the only altering differences (see (3)) and $d_{i,l_i}, d_{i,r_i+1} \neq 0, i \in [m]$. Moreover, from Lemma 3 it follows that

$$c_i(A) = c(A) \text{ implies } c_i(A') = c(A').$$

Consequently, it is enough to find a pair (i,j) with

$$\min\{a_{ij}, |d_{ij}|\} > 0 \text{ and } \min\{a'_{ij}, |d'_{ij}|\} = 0$$

or an index i with

$$c_i(A) < c(A) \text{ and } c_i(A') = c(A').$$

Let i be that index for which $u_{max} = u_{\hat{I}_i}$.

Case 1. $u_{\hat{I}_i} = w_{\hat{I}_i}$. By (10) clearly $I_i \neq \emptyset$. Let $j \in \hat{I}_i$ be that index for which $w_{\hat{I}_i} = a_{ij}$. Then $a_{ij} = u_{max} > 0$. Now let λ be the smallest index such that

$$a_{i,\lambda} = a_{i,\lambda+1} = \cdots = a_{ij}.$$

Then $\lambda \in \hat{I}_i$ and $\min\{a_{i,\lambda}, |d_{i,\lambda}|\} > 0$, but $\min\{a'_{i,\lambda}, |d'_{i,\lambda}|\} = a'_{i,\lambda} = 0$.

Case 2. $u_{\hat{I}_i} = v_{\hat{I}_i}$. By (9)

$$c_i(A') = c(A) - v_{\hat{I}_i} = c(A) - u_{max} = c(A').$$

Thus we are done if $c_i(A) < c(A)$. If $c_i(A) = c(A)$, then (by Lemma 7)

$$u_{\hat{I}_i} = \min\{d_{i,l_i}, -d_{i,r_i+1}\} > 0$$

which implies

$$\min\{a_{i,l_i}, |d_{i,l_i}|\} > 0 \text{ and } d_{i,r_i+1} \neq 0.$$

If $u_{\hat{I}_i} = d_{i,l_i}$, then $d'_{i,l_i} = 0$, i.e. $\min\{a'_{i,l_i}, |d'_{i,l_i}|\} = 0$. Thus let $u_{\hat{I}_i} = -d_{i,r_i+1}$. If $a_{i,r_i+1} \neq 0$, then $\min\{a_{i,r_i+1}, |d_{i,r_i+1}|\} > 0$ but $\min\{a'_{i,r_i+1}, |d'_{i,r_i+1}|\} = d'_{i,r_i+1} = 0$. If $a_{i,r_i+1} = 0$, then $a_{i,r_i} > 0$ and $u_{\hat{I}_i} = a_{i,r_i} = w_{\hat{I}_i}$, and we are in Case 1. □

The algorithm presented so far is good, but it can be slightly improved. Up to now we have taken that segment which is given by intervals I_i having the largest possible u_{I_i}. These intervals can be computed simultaneously with the determination of u_{max}. Now we present a better way for the construction of S after the determination of u_{max}. This method is suggested by the proof of Theorem 9, but does not completely correspond to the proof.

Main strategy for the choice of S:

For given A and given u_{max} take such a segment $\tilde{S}$ such that $(u_{max}, \tilde{S})$ is admissible and, for $A' = A - u_{max}\tilde{S}$, the NS–complexity $q(A')$ is rather small.

Clearly, one could replace "rather small" by "minimum", but numerical tests have shown that the variant presented below is in general slightly better. For each $i \in [m]$, we consider all essential intervals $I_i = [l_i, r_i]$. First recall that in order to get an admissible pair $(u_{max}, \tilde{S})$ we must have by (6) and (7)

$$u_{max} \leq c(A) - c_i(A) \text{ if } I_i = \emptyset, \tag{12}$$

$$\max\{0, d_{i,l_i} - u_{max}\} + \max\{d_{i,r_i+1} + u_{max}\} + u_{max} \\ \leq c(A) - c_i(A) + d_{i,l_i} \text{ if } I_i \neq \emptyset \tag{13}$$

(note that $d_{i,l_i} > 0$ and $d_{i,r_i+1} < 0$). For each such interval we define its *potential* p_{I_i}

$$p_{I_i} := 0 \text{ if } I_i = \emptyset, \tag{14}$$

$$p_{I_i} := p_{I_i}^{(1)} + p_{I_i}^{(2)} + p_{I_i}^{(3)} \text{ if } I_i \neq \emptyset, \tag{15}$$

where

$$p_{I_i}^{(1)} := \begin{cases} 1 & \text{if } u_{max} = d_{i,l_i} \text{ and } a_{i,l_i} \neq u_{max}, \\ 0 & \text{otherwise}, \end{cases} \tag{16}$$

$$p_{I_i}^{(2)} := \begin{cases} 1 & \text{if } u_{max} = d_{i,r_i+1} \text{ and } a_{i,r_i+1} \neq u_{max}, \\ 0 & \text{otherwise}, \end{cases} \tag{17}$$

$$p_{I_i}^{(3)} := |\{j \in [l_i, r_i] : a_{ij} = u_{max}\}| \tag{18}$$

and its *length* l_{I_i} by

$$l_{I_i} := \begin{cases} 0 & \text{if } I_i = \emptyset, \\ r_i - l_i + 1 & \text{if } I_i = [l_i, r_i] \neq \emptyset. \end{cases} \tag{19}$$

For the construction of $\tilde{S}$ we take for each $i \in [m]$ such an interval $\tilde{I}_i$ which satisfies (12) resp. (13) and has in first instance maximum potential and (if there are several of them) in second instance maximum length. Then the searched–after segment $\tilde{S}$ is determined by $(\tilde{I}_1, \ldots, \tilde{I}_m)$.

Now we summarize the whole algorithm:

Special minimum TNMU–algorithm, the TNMU–NS–algorithm:
while $A \neq \boldsymbol{O}$

- Determine for each i and each essential interval I_i the number v_{I_i} according to Lemma 7, the number w_{I_i} according to (10) and the number u_{I_i} according to (11).
- Put $u_{max} := \min_{i \in [m]} \max\{u_{I_i} : I_i \text{ is an essential interval}\}$.
- Find for each i an essential interval $\tilde{I}_i$ such that (12) resp. (13) holds and $\tilde{I}_i$ has in first instance maximum potential and in second instance maximum length where the potential is computed by (14)–(18) and the length is given by (19).
- Let $\tilde{S}$ be the segment associated with $(\tilde{I}_1, \ldots, \tilde{I}_m)$.
- output $(u_{max}, \tilde{S})$.
- $A := A - u_{max}\tilde{S}$.

It is not difficult to find an implementation having for each step time-complexity $O(mn^2)$, so that the whole algorithm has complexity $O(m^2n^3)$.

4 Results

The output of our algorithm applied to the benchmark–matrix from Section 2 is the segmentation presented there. It is a small exercise to show that it does not exist a segmentation with 5 segments (already for the first two rows 6 segments are necessary). We know from Theorem 1 that our algorithm leads to a segmentation with minimum TNMU, so it is optimal in this way. But our algorithm provides also for many other examples a very small number of segments. The best known algorithm concerning the average number of segments which is able to work with 1,000 random matrices in reasonable time is to our knowledge, the algorithm of Xia and Verhey [13], but note that the time–consuming algorithms of Dai, Zhu [5] and Langer et al [8] provide in general for single matrices somewhat better results. The Xia–Verhey–algorithm is far away from being optimal w.r.t. to the TNMU, in contrast to the algorithm of Bortfeld et al and our algorithm. The following table contains the average number of segments for the case that the entries of the matrix A are randomly chosen elements from $\{0, 1, \ldots, L\}$ (uniformly distributed) and 1,000 or 10,000 matrices of dimension 15×15 have been generated. In addition, we compare the average total number of monitor units.

L	NS Bortfeld	NS Xia	NS new	TNMU Bortfeld	TNMU Xia	TNMU new
3	14.0	11.1	9.9	14.0	16.6	14.0
4	17.9	14.1	11.2	17.9	22.4	17.9
5	21.8	15.1	12.0	21.8	25.0	21.7
6	25.6	17.9	12.8	25.6	37.7	25.6
7	29.5	16.2	13.5	29.5	38.8	29.4
8	33.3	20.2	14.1	33.3	46.3	33.2
9	37.1	20.2	14.5	37.1	51.0	37.0
10	40.9	20.5	15.0	40.9	53.9	40.9
11	44.8	21.6	15.5	44.8	55.7	44.7
12	48.6	21.8	15.8	48.6	81.1	48.5
13	52.4	22.4	16.2	52.4	83.3	52.3
14	56.2	22.8	16.5	56.2	83.5	56.2
15	60.1	23.5	16.8	60.1	83.5	59.8
16	63.8	23.9	17.1	63.8	93.6	63.6

The results for the algorithm of Bortfeld et al and of Xia, Verhey are taken from [13] (with in each case 1,000 matrices). To obtain the whole columns for our algorithm (with in each case 10,000 matrices) a 1.8 GHz PC needs 125 seconds (i.e. treating 140,000 matrices). The small differences in the TMNU–Bortfeld–column and TMNU–new–column have their reason in the random choice. But with 10,000 matrices the estimate seems to be sufficiently stable. We mention that for $L = 10,000$ (and 15×15 matrices) our algorithm provides in the average 48.9 segments and a TNMU of 37,880.2. For the segmentation of one 100×100 matrix with $L = 10,000$ the PC needs 3 seconds, so our algorithm is completely practicable.

5 Concluding remarks

At the moment we do not have a well–founded explanation why our algorithm provides this small number of segments. This should be part of further research. But we mention that we also tested several other criteria for the choice of an admissible segmentation pair (u, S). Let S be associated with $(I_1, \ldots, I_m)$. The *area* of S is defined to be the sum of the lengths of the m intervals. For example, the following methods are plausible. Choose under all admissible pairs (u, S) such a pair for which:

1. method: $area(S)$ is maximum and in second instance u is maximum.

2. method: $u \cdot area(S)$ is maximum.

3. method: The NS–complexity $q(A - uS)$ is minimum (without first maximizing u).

But all these methods are not better than the method presented in the paper and moreover have greater time–complexity.

In this paper, we did not consider additional constraints like the interleaf collision constraint (machine–dependent) and tongue and groove constraints. Concerning the TNMU–problem with interleaf collision constraint, there have been obtained two important results recently: Boland et al [1] designed an algorithm in a network–flow–framework and Kalinowski [7] generalized Theorem 1 to the case of interleaf collision constraint and, as a by–product, obtained a very efficient leaf–sequencing algorithm. Present research is directed to minimizing simultaneously the number of segments.

Acknowledgement. I would like to thank Thomas Kalinowski for a helpful discussion.

References

[1] N. Boland, H.W. Hamacher, F. Lenzen, Minimizing beam–on–time in cancer radiation treatment using multileaf collimators, submitted, 2002.

[2] T.R. Bortfeld, D.L. Kahler, T.J. Waldron, A.L. Boyer, X–ray field compensation with multileaf collimators, Int. J. Radiat. Oncol. Biol. Phys. 28 (1994) 723–730.

[3] A.L. Boyer, C.Y. Yu, Intensity–modulated radiation therapy with dynamic multileaf collimators, Semin. Radiat. Oncol. 9 (1999) 48–59.

[4] A. Brahme, Treatment optimization using physical and radiobiological objective functions, in: A.R. Smith, ed., Radiation Therapy Physics (Springer–Verlag, Berlin, 1995) 209–246.

[5] J. Dai, Y. Zhu, Minimizing the number of segments in a delivery sequence for intensity–modulated radiation therapy with a multileaf collimator, Med. Phys. 28 (2001) 2113–2120.

[6] J.M. Galvin, X.G. Chen, R.M. Smith, Combining multileaf fields to modulate fluence distributions, Int. J. Radiat. Oncol. Biol. Phys. 27 (1993) 697–705.

[7] T. Kalinowski, An algorithm for optimal mulileaf collimator field segmentation with interleaf collision constraint, Preprint 03/2, University of Rostock (2003).

[8] M. Langer, V. Thai, L. Papiez, Improved leaf sequencing reduces segments of monitor units needed to deliver IMRT using multileaf collimators, Med. Phys. 28 (2001) 2450–2458.

[9] L. Ma, A.L. Boyer, L. Xing, C.M. Ma, An optimized leaf setting algorithm for beam intensity modulation using dynamic multileaf collimators, Phys. Med. Biol. 43 (1998) 1629–1643.

[10] W. Que, Comparison of algorithms for multileaf collimator field segmentation, Med. Phys. 26 (1999) 2390–2396.

[11] R.A.C. Siochi, Minimizing static intensity modulation delivery time using an intensity solid paradigm, Int. J. Radiat. Oncol. Biol. Phys. 43 (1999) 671–680.

[12] J. Stein, T. Bortfeld, B. Dörschel, W. Schlegel, Dynamic X–ray compensation for conformal radiotherapy by means of multi–leaf collimation, Radiother. Oncol. 32 (1994) 163–173.

[13] P. Xia, L. Verhey, Multileaf collimator leaf–sequencing algorithm for intensity modulated beams with multiple static segments, Med. Phys. 25 (1998) 1424–1434.