Intensitätsmodulierte Bestrahlungsfelder mit geräteabhängigen Einschränkungen

Thomas Kalinowski

Institut für Mathematik
Universität Rostock
thomas.kalinowski@uni-rostock.de

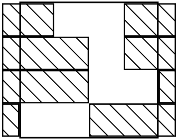
Arbeitskreis IMRT
Neuruppin
27. März 2008

Inhalt

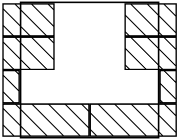
Inhalt

Ein Multileaf-Kollimator

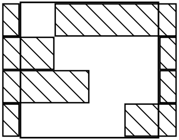
Modulierung durch Überlagerung homogener Felder



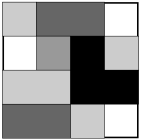
2 MU



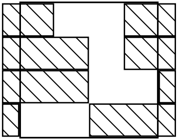
1 MU



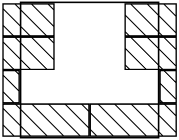
1 MU



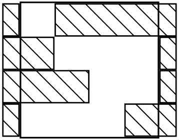
Modulierung durch Überlagerung homogener Felder



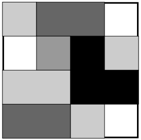
2 MU



1 MU



1 MU



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

Eine Fluenzmatrix

- Beschreibung der gewünschten Fluenz durch eine nichtnegative ganzzahlige Matrix.
- Beispiel:

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

. . . entsprechen *Segmenten:*

Die homogenen Felder

. . . entsprechen *Segmenten:*

- $0-1-$Matrizen

Die homogenen Feld
er

... entsprechen *Segmenten:*

- $0-1-$Matrizen
- In jeder Zeile gibt es genau ein Intervall von zusammenhängenden Einsen.

Die homogenen Felder

... entsprechen *Segmenten:*

- $0-1-$Matrizen
- In jeder Zeile gibt es genau ein Intervall von zusammenhängenden Einsen.
- Beispiel:

$$\begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$$

Eine Segmentierung

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

$$+1\begin{pmatrix} 0&0&0&0&0&1\\ 0&0&0&1&1&1\\ 0&0&0&0&1&0\\ 0&1&1&1&1&0 \end{pmatrix} + 1\begin{pmatrix} 0&0&0&1&1&1\\ 0&0&0&1&0&0\\ 0&1&1&0&0&0\\ 1&1&1&0&0&0 \end{pmatrix} + 1\begin{pmatrix} 0&1&0&0&0&0\\ 1&0&0&0&0&0\\ 1&1&0&0&0&0\\ 1&1&1&1&1&0 \end{pmatrix}$$

$$+1\begin{pmatrix} 1&1&0&0&0&0\\ 1&1&1&1&0&0\\ 1&1&1&1&1&1\\ 0&0&0&0&0&0 \end{pmatrix}.$$

Optimierungsziele

- Minimierung der Gesamtbestrahlungsdauer
  - Bezeichnung: DT (**D**elivery **T**ime) angegeben in MU (**M**onitor **U**nits)
  - sehr effizient exakt lösbar

Optimierungsziele

- Minimierung der Gesamtbestrahlungsdauer
  - Bezeichnung: DT (**D**elivery **T**ime) angegeben in MU (**M**onitor **U**nits)
  - sehr effizient exakt lösbar
- Minimierung der Anzahl der benötigten Felder
  - algorithmisch sehr schwierig
  - heuristische suboptimale Lösungen

Optimierungsziele

- Minimierung der Gesamtbestrahlungsdauer
  - Bezeichnung: DT (**D**elivery **T**ime) angegeben in MU (**M**onitor **U**nits)
  - sehr effizient exakt lösbar
- Minimierung der Anzahl der benötigten Felder
  - algorithmisch sehr schwierig
  - heuristische suboptimale Lösungen
- Vermeidung von dosimetrisch ungünstigen Feldern
  - kaum untersucht

Inhalt

Grundidee

- Voraussetzung: Die Lamellenpositionen in den verschiedenen Zeilen sind unabhängig voneinander.
- Es reicht einzelne Zeilen zu betrachten.
- z.B. $\begin{pmatrix} 2 & 6 & 3 & 3 & 4 & 2 & 1 & 1 & 5 & 5 \end{pmatrix}$

Grundidee

- Voraussetzung: Die Lamellenpositionen in den verschiedenen Zeilen sind unabhängig voneinander.
- Es reicht einzelne Zeilen zu betrachten.
- z.B. $\begin{pmatrix} 2 & 6 & 3 & 3 & 4 & 2 & 1 & 1 & 5 & 5 \end{pmatrix}$
- *unvermeidbare Positionen*

linke Lamelle		rechte Lamelle	
Position 1	2 MU	Position 2	3 MU
Position 2	4 MU	Position 5	2 MU
Position 5	1 MU	Position 6	1 MU
Position 9	4 MU	Position 10	5 MU

- Insgesamt braucht man also mindestens 11 MU.

links	rechts
1	2
1	2
2	2
2	5
2	5
2	6
5	10
9	10
9	10
9	10
9	10

$$\begin{pmatrix}2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{pmatrix}$$

$$\begin{pmatrix}0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\end{pmatrix}$$

$$\begin{pmatrix}0 & 2 & 2 & 2 & 2 & 0 & 0 & 0 & 0 & 0\end{pmatrix}$$

$$\begin{pmatrix}0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0\end{pmatrix}$$

$$\begin{pmatrix}0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1\end{pmatrix}$$

$$\begin{pmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 4\end{pmatrix}$$

$$\begin{pmatrix}2 & 6 & 3 & 3 & 4 & 2 & 1 & 1 & 5 & 5\end{pmatrix}$$

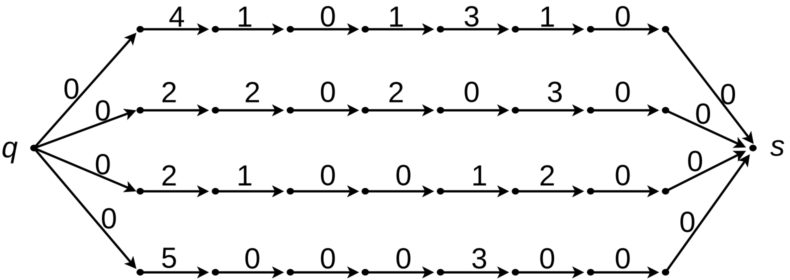
Formale Charakterisierung der minimalen DT

- $m \times n-$Fluenzmatrix $A = (a_{ij})$
- Zeilenkomplexität $c_i(A) = \sum_{j=1}^{n} \max\{0, a_{ij} - a_{i,j-1}\}$
- Gesamtkomplexität $c(A) = \max_{i \in [m]} c_i(A)$

Satz

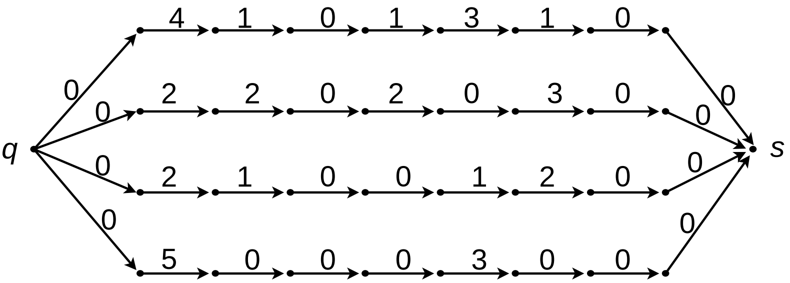
Die minimale DT zur Realisierung der Fluenzmatrix A ist gleich $c(A)$.

- Fluenzmatrix $\begin{pmatrix} 4 & 5 & 0 & 1 & 4 & 5 \\ 2 & 4 & 1 & 3 & 1 & 4 \\ 2 & 3 & 2 & 1 & 2 & 4 \\ 5 & 3 & 3 & 2 & 5 & 3 \end{pmatrix}$
- gerichteter Graph mit Gewichten
  $w((i,j-1),(i,j)) = \max\{0, a_{ij} - a_{i,j-1}\}$



Graphentheoretische Interpretation

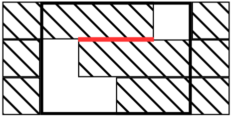
- gerichteter Graph mit Gewichten
  $w((i,j-1),(i,j)) = \max\{0, a_{ij} - a_{i,j-1}\}$



- minimale DT $=$ maximale Weglänge von q nach s.

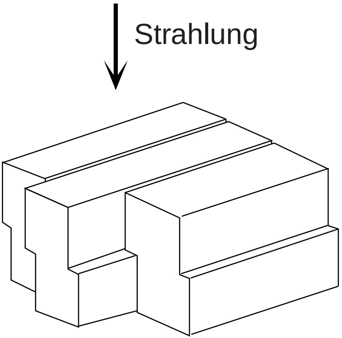
Inhalt

- Gegenüberliegende Lamellen in benachbarten Zeilen dürfen nicht überlappen.
- Beispiel:



$$\Rightarrow \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \text{ ist kein Segment.}$$

Tongue-and-Groove



Unterdosierung

$$A = \begin{pmatrix} 2 & 3 \\ 3 & 4 \end{pmatrix}$$

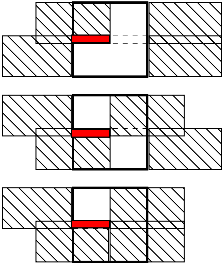
Unterdosierung

$$A = \begin{pmatrix} 2 & 3 \\ 3 & 4 \end{pmatrix}$$

3 MU

1 MU

1 MU



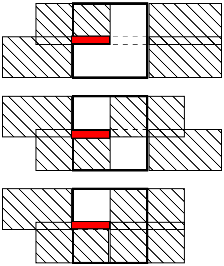
Unterdosierung

$$A = \begin{pmatrix} 2 & 3 \\ 3 & 4 \end{pmatrix}$$

3 MU

1 MU

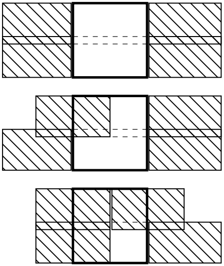
1 MU



2 MU

1 MU

1 MU



Vermeidung der Unterdosierung

$$
\begin{aligned}
a_{ij} \leq a_{i+1,j} \wedge s_{ij} = 1 \Rightarrow \; s_{i+1,j} = 1 \qquad & (i \in [m-1],\; j \in [n]),\\
a_{ij} \leq a_{i-1,j} \wedge s_{ij} = 1 \Rightarrow \; s_{i-1,j} = 1 \qquad & (i \in [2,m],\; j \in [n]).
\end{aligned}
$$

Vermeidung der Unterdosierung

$$
\begin{aligned}
a_{ij} \le a_{i+1,j} \wedge s_{ij} = 1 &\Rightarrow\ s_{i+1,j} = 1 \qquad && (i \in [m-1],\ j \in [n]),\\
a_{ij} \le a_{i-1,j} \wedge s_{ij} = 1 &\Rightarrow\ s_{i-1,j} = 1 \qquad && (i \in [2,m],\ j \in [n]).
\end{aligned}
$$

- Die Fluenz im Grenzbereich zwischen (i,j) und $(i+1,j)$ beträgt $\min\{a_{ij}, a_{i+1,j}\}$.

Definition. *A*-Segment

Ein *A–Segment* ist eine $m \times n$-Matrix $S = (s_{ij})$ mit Einträgen aus $\{0, 1\}$, so dass es ganze Zahlen $l_i,\ r_i$ ($i \in [m]$) mit folgenden Eigenschaften gibt.

Definition. A-Segment

Ein *A–Segment* ist eine $m \times n$-Matrix $S = (s_{ij})$ mit Einträgen aus $\{0, 1\}$, so dass es ganze Zahlen $l_i,\ r_i$ $(i \in [m])$ mit folgenden Eigenschaften gibt.

- $l_i \leq r_i + 1 \qquad (i \in [m])$,

Definition. *A*-Segment

Ein *A–Segment* ist eine $m \times n$-Matrix $S = (s_{ij})$ mit Einträgen aus $\{0, 1\}$, so dass es ganze Zahlen $l_i,\ r_i$ $(i \in [m])$ mit folgenden Eigenschaften gibt.

- $l_i \leq r_i + 1 \qquad (i \in [m]),$
- $s_{ij} = \begin{cases} 1 & \text{if } l_i \leq j \leq r_i \\ 0 & \text{sonst} \end{cases} \qquad (i \in [m],\ j \in [n]),$

Definition. *A*-Segment

Ein *A–Segment* ist eine $m \times n$-Matrix $S = (s_{ij})$ mit Einträgen aus $\{0, 1\}$, so dass es ganze Zahlen $l_i,\ r_i$ $(i \in [m])$ mit folgenden Eigenschaften gibt.

- $l_i \leq r_i + 1 \qquad (i \in [m]),$
- $s_{ij} = \begin{cases} 1 & \text{if } l_i \leq j \leq r_i \\ 0 & \text{sonst} \end{cases} \qquad (i \in [m],\ j \in [n]),$
- ICC: $l_i \leq r_{i+1} + 1,\ r_i \geq l_{i+1} - 1 \quad (i \in [m-1])$

Definition. A-Segment

Ein A–*Segment* ist eine $m \times n$-Matrix $S = (s_{ij})$ mit Einträgen aus $\{0, 1\}$, so dass es ganze Zahlen $l_i,\ r_i$ $(i \in [m])$ mit folgenden Eigenschaften gibt.

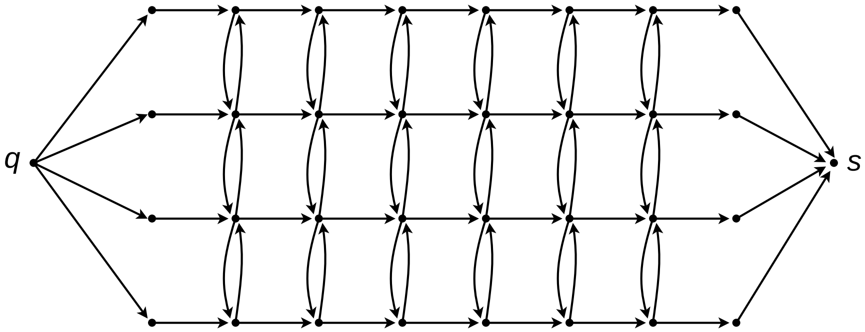
- $l_i \leq r_i + 1 \qquad (i \in [m])$,
- $s_{ij} = \begin{cases} 1 & \text{if } l_i \leq j \leq r_i \\ 0 & \text{sonst} \end{cases} \qquad (i \in [m],\ j \in [n])$,
- ICC: $l_i \leq r_{i+1} + 1,\ r_i \geq l_{i+1} - 1 \quad (i \in [m-1])$

und außerdem gilt

- Tongue-and-Groove Bedingung (TGB):

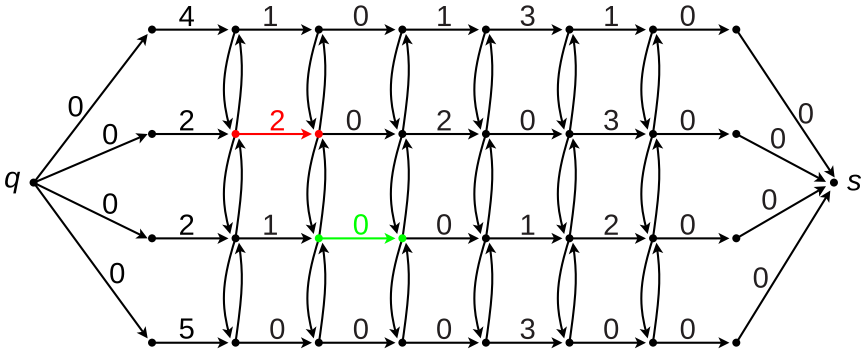
$$a_{ij} \leq a_{i+1,j} \wedge s_{ij} = 1 \ \Rightarrow\ s_{i+1,j} = 1 \quad (i \in [m-1],\ j \in [n]),$$
$$a_{ij} \leq a_{i-1,j} \wedge s_{ij} = 1 \ \Rightarrow\ s_{i-1,j} = 1 \quad (i \in [2, m],\ j \in [n]).$$

Der Segmentierungsgraph



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

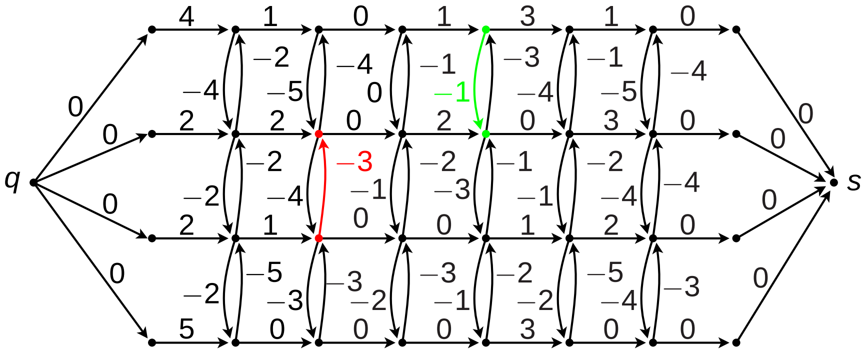
Die Gewichtsfunktion



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

$$w((i,j-1),(i,j)) = \max\{0, a_{ij} - a_{i,j-1}\}$$

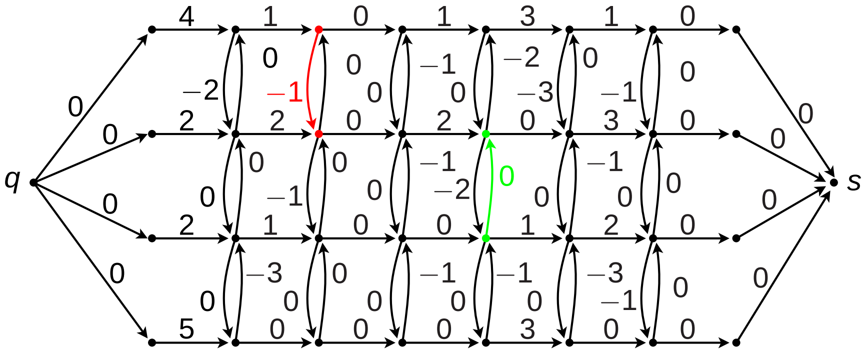
Die ICC-Gewichtsfunktion



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

$$w((i,j),(i\pm 1,j)) = -a_{ij}$$

Die ICC-TGB-Gewichtsfunktion



$$A = \begin{pmatrix} 4 & \color{red}{5} & 0 & 1 & 4 & 5 \\ 2 & \color{red}{4} & 1 & \color{green}{3} & 1 & 4 \\ 2 & 3 & 2 & \color{green}{1} & 2 & 4 \\ 5 & 3 & 3 & 2 & 5 & 3 \end{pmatrix}$$

$$w((i,j),(i \pm 1, j)) = \min\{0, a_{i\pm1,j} - a_{ij}\}$$

Konkrete Interpretation der Gewichtsfunktionen

- Wir setzen

$$\alpha_1(i,j) = \text{maximale Länge eines Weges von } q \text{ nach } (i,j).$$

- Dann ist $\alpha_1(i,j)$ eine untere Schranke für die Anzahl der Monitor Units mit $l_i \leq j$.
- Damit ist die maximale Länge eines $q-s-$Weges (oder äquivalent $\max_i \alpha_1(i,n)$) eine untere Schranke für die DT.
- Wir setzen deshalb

$$c(A) = \text{maximale Weglänge im Segmentierungsgraphen.}$$

Min-Max–Charakterisierung

Satz

Die minimale DT für eine MLC-Segmentierung mit ICC bzw. mit ICC und TGB ist gleich der maximalen Weglänge im Segmentierungsgraphen, wenn man jeweils die entsprechende Gewichtsfunktion verwendet.

- Die Zahlen $\alpha_1(i, j)$, und damit auch die minimale DT, können auch für sehr große Matrizen algorithmisch effizient bestimmt werden.
- Beweisidee: iterierte Konstruktion eines Segments S mit $c(A - S) = c(A) - 1$.

Basisversion des Algorithmus

- **Input**: Fluenzmatrix $A = A^{(0)}$, $k = 0$
- **Solange** $A^{(k)} \neq 0$
  - Bestimme bezüglich $A^{(k)}$ die Zahlen

$$\begin{aligned} \alpha_1(i,j) &= \text{maximale Länge eines Weges von } q \text{ nach } (i,j), \\ \alpha_2(i,j) &= \text{maximale Länge eines Weges von } (i,j) \text{ nach } s, \\ \alpha(i,j) &= \alpha_1(i,j) + \alpha_2(i,j). \end{aligned}$$

  - Setze $k = k + 1$
  - Definiere das Segment $S^{(k)} = \left(s_{ij}^{(k)}\right)$ durch

$$s_{ij}^{(k)} = \begin{cases} 1 & \text{falls } \alpha_1(i,j) = a_{ij} > 0, \text{ und } \alpha(i,j) = c(A), \\ 0 & \text{sonst.} \end{cases}$$

  - Setze $A^{(k)} = A^{(k-1)} - S^{(k)}$
- **Output**: $S^{(1)},\ S^{(2)}, \ldots,\ S^{(k)}$

Heuristische Reduktion der Segmentanzahl

- **Input**: Fluenzmatrix A, $k = 0$
- **Solange** $A^{(k)} \neq 0$
  - Setze $k = k + 1$
  - Bestimme ein Paar $(u_k, S^{(k)})$ mit maximalem u_k, so dass
    - $A^{(k-1)} - u_k S^{(k)}$ nichtnegativ ist,
    - $c(A^{(k-1)} - u_k S^{(k)}) = c(A^{(k-1)}) - u$
  - Setze $A^{(k)} = A^{(k-1)} - u_k S^{(k)}$
- **Output**: $(u_1, S^{(1)}),\ (u_2, S^{(2)}), \dots,\ (u_k, S^{(k)})$

Beispiel

$$\begin{pmatrix}
0 & 0 & 0 & 2 & 9 & 8 & 5 & 0 & 0 & 0 \\
0 & 0 & 0 & 3 & 4 & 0 & 2 & 4 & 0 & 0 \\
0 & 0 & 2 & 7 & 5 & 6 & 5 & 5 & 8 & 0 \\
0 & 0 & 2 & 4 & 6 & 4 & 4 & 2 & 4 & 1 \\
2 & 3 & 3 & 8 & 7 & 4 & 4 & 2 & 5 & 10 \\
3 & 3 & 4 & 4 & 7 & 5 & 6 & 3 & 7 & 9 \\
0 & 1 & 6 & 10 & 8 & 8 & 8 & 8 & 7 & 10 \\
0 & 4 & 6 & 4 & 4 & 3 & 2 & 6 & 10 & 9 \\
0 & 6 & 3 & 1 & 2 & 1 & 1 & 2 & 0 & 0 \\
0 & 0 & 2 & 4 & 1 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}$$

- Unser Algorithmus liefert eine Segmentierung mit 11 Segmenten und mit $DT = 17$.
- Die Rechenzeit ist vernachlässigbar ($\ll 1 sek$).

Tests auf Zufallsmatrizen

- $15 \times 15-$Matrizen mit zufälligen Einträgen aus $\{0, 1, \ldots, L\}$, ($L = 3, \ldots, 16$)
- Rechenzeit
  - reine DT-Minimierung: wenige Sekunden für 1000 Segmentierungen

Tests auf Zufallsmatrizen

- $15 \times 15-$Matrizen mit zufälligen Einträgen aus $\{0, 1, \ldots, L\}$, ($L = 3, \ldots, 16$)
- Rechenzeit
  - reine DT-Minimierung: wenige Sekunden für 1000 Segmentierungen
  - mit der Heuristik für die Segmentanzahl:
    - 15 Minuten für 1000 Matrizen
    - maximal 13 Sekunden für eine einzelne Matrix

Ergebnisse

L	DT	Segmente
4	21.2	18.0
6	30.3	22.6
8	39.2	25.7
10	48.2	28.3
12	57.2	30.5
14	66.0	32.2
16	74.8	33.9

Tabelle: Resultate mit Eliminierung der TG-Unterdosierung.

L	TNMU	Segmente
4	19.5	14.5
6	27.6	17.2
8	35.7	19.1
10	43.8	20.7
12	51.8	21.9
14	59.8	23.0
16	67.7	24.0

Tabelle: Resultate ohne Eliminierung der TG-Unterdosierung.

Inhalt

Zusammenfassung

- Die DT-optimale MLC-Segmentierung von Fluenzmatrizen ist für folgende Fälle sehr effizient möglich.
  - unabhängige Zeilen (Bortfeld)
  - ICC
  - ICC und TGB
- Die Anzahl der Segmente kann nur für sehr kleine Matrizen exakt minimiert werden.
- Es gibt aber leistungsfähige Heuristiken zur näherungsweisen Minimierung der Segmentanzahl.

Offene Probleme

- DT-Minimierung für MLCs ohne ICC aber mit TGB.
- dosimetrische Probleme: Streuungs- und Beugungseffekte, die für kleine Felder relevant werden
  - mathematische Modellierung dieser Effekte
  - Entwicklung eines Segmentierungsalgorithmus, der die Dosimetrie berücksichtigt

Vielen Dank für die Aufmerksamkeit!