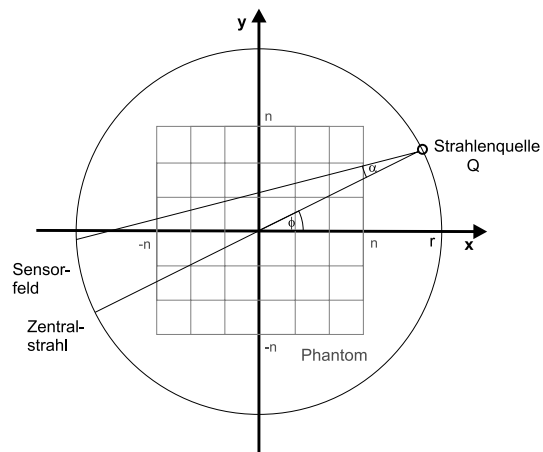
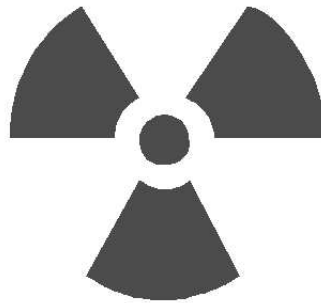


Computertomographie - CT

Alexander Croy*, Ivo Kabadshow†

14.1.1999

Betreuer: Prof. Michael Eiermann
Institut für Angewandte Mathematik II
TU Bergakademie Freiberg



*alexander.croy@physik.tu-chemnitz.de

†ivo.kabadshow@e-technik.tu-chemnitz.de

Einleitung

Archäologen, Mediziner und Techniker sind häufig mit dem Problem konfrontiert, die Inhalte einer „uneinsehbaren Kammer“ festzustellen. In der medizinischen Diagnostik zum Beispiel ist diese Kammer der Körper eines Patienten. Es wurde lange Zeit nach Verfahren gesucht, die präzise sind und für den Patienten ohne weitere Belastungen bleiben. Dieses Ziel sahen viele durch Conrad Röntgens Entdeckung der „X-Strahlen“ erreicht. Jedoch zeigte sich, daß die Röntgenbilder nicht für alle Anwendungen geeignet waren. Neue Wege mußten beschritten werden. Einen bis heute erfolgreichen Ansatz stellt die Verbindung von Röntgen- und Computertechnik in Form der Computertomographie dar. Die Zusammensetzung aus dem griechischen Wörtern *Tomo* und *Graph*, was soviel wie „Schicht“ und „Aufzeichnen“ bedeutet, beschreibt schon die Funktionsweise und die Vorteile dieses Verfahrens.

Die mathematischen Grundlagen für diese Methode wurden bereits 1918 von dem österreichischen Mathematiker Johann Radon mit den Radon-Transformationen gelegt. Diese theoretischen Betrachtungen konnten allerdings erst 50 Jahre später mit Hilfe moderner Rechensysteme praktisch angewandt werden. Der Brite Godfrey N. Hounsfield konstruierte 1968 das erste Computertomographiesystem, das auch in der Diagnostik eingesetzt werden konnte. Für diesen Beitrag zum Fortschritt der Medizin erhielt er 1979 zusammen mit dem amerikanischen Physiker Allan Cormack, der sich schon vorher theoretisch mit dem Problem beschäftigt hatte, den Nobelpreis für Physiologie und Medizin. Zu Ehren Hounsfields wird auch heute noch der Dichtewert in der Hounsfieldeinheit HU angegeben.

Das Ziel der vorliegenden Arbeit soll die mathematische Erfassung der Computertomographie als numerisches Problem und die Erarbeitung einer Computersimulation sein. Dazu werden die physikalischen Grundlagen im Vorfeld diskutiert und schließlich heutige Anwendungsbeispiele dargestellt. Der mathematische Schwerpunkt liegt bei der Rekonstruktion eines zweidimensionalen Bildes aus eindimensionalen Projektionen. Diese Aufgabe nimmt in der numerischen Mathematik einen hohen Stellenwert ein, da sich ein breites Anwendungsspektrum aufzeigt. Beispiele dafür sind die Untersuchung der molekularen Struktur von Bakterien mit Hilfe der Elektronenmikroskopie oder die Erstellung der Röntgenstrahlstruktur aus den Überresten einer Supernova. Das Gebiet der Computertomographie ermöglicht einen interessanten und praktisch bedeutsamen Zugang zu diesen numerischen Problemen.

Inhaltsverzeichnis

1	Physik	1
1.1	Zur Entstehung der Röntgenstrahlung	1
1.2	Eigenschaften der Röntgenstrahlung	2
1.3	Schwächungseffekte	2
1.4	Probleme	5
2	Mathematik	6
2.1	Das mathematische Modell	6
2.2	Formelle Vereinbarungen	8
2.3	Erste Berechnungen	8
2.4	Das direkte Problem	10
2.5	Das inverse Problem	12
2.5.1	Lösen von überbestimmten Gleichungssystemen	13
2.5.2	Iterationsverfahren	15
2.5.3	Simulation	16
2.5.4	Probleme	16
2.6	Bedienung der graphischen Oberfläche	18
2.6.1	Interface - direktes Problem	18
2.6.2	Interface - inverses Problem	20
3	Anwendung	21
4	Schlußbetrachtungen	25
A	Anhang	27
A.1	Berechnung der Schnittpunkte - » <i>spunkt.m</i> «	27
A.2	Erstellen einer eigenen Schwächungsmatrix - » <i>make_object.m</i> «	28
A.3	Berechnung der Schwächung - » <i>ct_dcalc.m</i> «	29
A.4	Erzeugung des Schweizer-Kreuz - » <i>make_scross.m</i> «	30

Abbildungsverzeichnis

1	Modell der Röntgenröhre	1
2	Wechselwirkungen im Atom	2
3	Effekte bei kleiner Spannung	3
4	Effekte bei großer Spannung	3
5	Aufbau des mathematischen Modells	7
6	Schema zur Indexberechnung	9
7	Schweizer Kreuz Phantom	11
8	Schwächungskurve	11
9	Beispiel für ein überbestimmtes Gleichungssystem	13
10	Schema für die beste Näherungslösung	14
11	Besetzung des Gleichungssystems	15
12	Gegebenes Phantom	16
13	Rekonstruiertes Bild	16
14	Rekonstruiertes Bild bei 540 Gleichungen	17
15	Rekonstruiertes Bild bei 810 Gleichungen	17
16	Unterschiede bei 540 Gleichungen	18
17	Unterschiede bei 810 Gleichungen	18
18	Benutzeroberfläche des direkten Problems	19
19	Benutzeroberfläche des inversen Problems	21
20	Allan M. Cormack	21
21	Godfrey N. Hounsfield	21
22	Funktionsprinzip eines Computertomographen	22
23	Typisches CT Bild (Schädel-CT)	23
24	Das von Siemens entwickelte Gerät SOMOTOM Plus 4	24

1 Physik

Die unsichtbaren Strahlen. Es ist ja offensichtlich, daß man einem Menschen nicht ansehen kann, wie es in seinem Inneren aussieht. Dies ist aber zum Beispiel bei medizinischen Untersuchungen unerlässlich. Genau hier, wo das menschliche Auge nichts mehr erfassen kann, beginnt die Arbeit der Computertomographen. Sie verwenden dabei nicht das sichtbare Licht, sondern Röntgenstrahlung. Diese besitzt eine höhere Wellenlänge und liegt in einem Bereich von 10^{-8} bis 10^{-13} m, was einer Frequenz von 10^{16} bis 10^{22} Hz entspricht. Daher läßt sich die Röntgenstrahlung im Lichtspektrum nach der Ultraviolettstrahlung einordnen. Was aber macht diese Strahlung so interessant?

1.1 Zur Entstehung der Röntgenstrahlung

Röntgenstrahlung entsteht beim Abbremsen energiereicher Elektronen. Diese werden aus einer Kathode emittiert, beschleunigt und prallen schließlich mit hoher Geschwindigkeit $\approx c$ auf die Anode. Die Energie der Strahlung kann nach der Gleichung $E = h \cdot f$ berechnet werden. Dabei ist $h = 6.626\,18 \cdot 10^{-34}$ Js als Planck'sches Wirkungsquantum definiert. Die Energiewerte werden in der Einheit eV (Elektronenvolt) angegeben; 1 eV entspricht der Energie, die ein Elektron bei einer Beschleunigungsspannung von 1 Volt erhält.

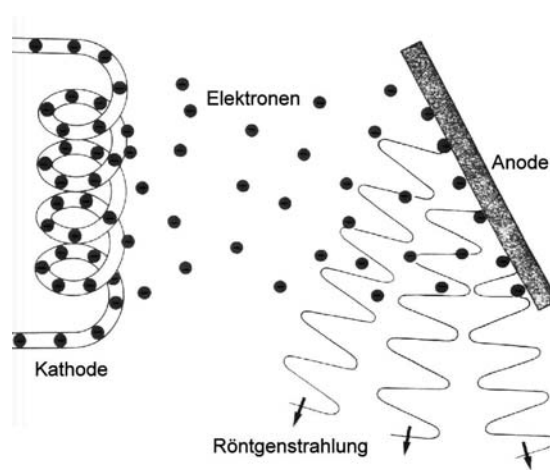


Abbildung 1: Modell einer Röntgenröhre (aus [1])

Die in der Röntgendiagnostik auftretenden Energiewerte liegen bei:

$$10^3 \text{eV} = 1 \text{keV (Kilo-eVolt)} \quad 10^6 \text{eV} = 1 \text{MeV (Mega-eVolt)}$$

1.2 Eigenschaften der Röntgenstrahlung

- Röntgenstrahlen durchdringen Stoffe und werden dabei geschwächt.
- Röntgenstrahlen regen bestimmte Stoffe zur Lichtemission an.
- Röntgenstrahlen ionisieren Gase.
- Röntgenstrahlen können Veränderungen an lebendem Gewebe verursachen.
- Röntgenstrahlen können photographische Filme Schwarzfärben.
- Röntgenstrahlen ändern die Leitfähigkeit und Ladung von Halbleitern.

1.3 Schwächungseffekte

Für unsere Arbeit ist die Schwächung der Strahlung von primärer Bedeutung. Erst dieser Effekt macht es möglich einen Querschnitt zu rekonstruieren. An dieser Stelle soll noch einmal kurz auf die verschiedenen Schwächungseffekte eingegangen und deren Entstehung geklärt werden.

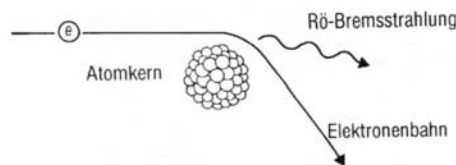


Abbildung 2: Wechselwirkung von Elektronen mit dem Atomkern (aus [1])

Absorption. Ein Teil der Energie des Röntgenquants wird dazu genutzt, ein Atom zu ionisieren, das heißt ein Elektron aus dem Kraftfeld des Atoms zu entfernen. Die verbleibende Energie wird als kinetische Energie mitgegeben. Das emittierte Elektron wird als **Photoelektron** bezeichnet. Dabei sind folgende Besonderheiten zu beobachten:

- Die Absorption nimmt mit größerer Wellenlänge zu.
- Die Absorption nimmt mit höherer Ordnungszahl des durchstrahlten Atoms zu.
- Die Absorption nimmt mit höherer Dichte des Materials zu.
- Die Absorption nimmt bei größer werdender Dicke zu.

Streuung. Hier unterscheidet man grundlegend zwei Arten: die **klassische** und die **Compton-Streuung**. Erstere tritt vorwiegend im langwelligen, letztere dagegen vorwiegend im kurzwelligen Bereich auf.

Bei der klassischen Streuung findet eine Richtungsänderung des Röntgenquants ohne Energieverlust statt. Dabei wird die Energie durch ein Elektron aufgenommen und in Form eines neuen Röntgenquants gleicher Energie wieder abgegeben. Bei der Compton-Streuung verlieren einige der Photonen beim vollkommen unelastischen Stoß mit Elektronen einen Teil ihrer Energie, die den Elektronen als kinetische Energie zugeführt wird. Das Röntgenquant wird bei diesem Vorgang abgelenkt. So ist es möglich, daß neben der gestreuten Strahlung gleicher Wellenlänge auch Strahlung größerer Wellenlänge auftritt.

Die verschiedenen Schwächungseffekte sind von der angelegten Röhrenspannung der Röntgenröhre abhängig.

- Bei niedriger Röhrenspannung überwiegt die Absorption. Die Streuung erfolgt in alle Richtungen.

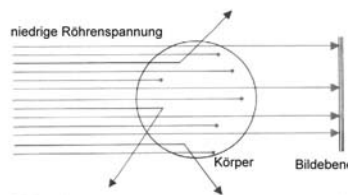


Abbildung 3: Niedrige Röhrenspannung (aus [1])

- Bei hoher Röhrenspannung nimmt die Absorption ab. Gleichzeitig ist eine Richtungsänderung der Streustrahlung in Richtung der Primärstrahlung zu beobachten.

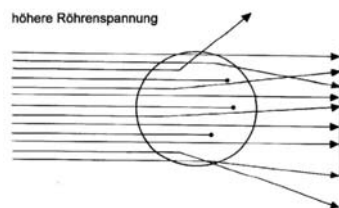


Abbildung 4: Hohe Röhrenspannung (aus [1])

Die Schwächungsgleichung. Die Grundlage für unsere Arbeit bildet nur eine Gleichung. Sie stellt den Zusammenhang zwischen Eintritts- und Austrittsintensität her.

$$I_{\text{out}} = I_{\text{in}} \cdot e^{-\mu \cdot L} \quad (1)$$

Die in der Gleichung auftretenden Größen sind folgendermaßen definiert:

- μ = linearer Schwächungskoeffizient des durchstrahlten Körpers.
- L = Strecke, die der Strahl in einem homogenen Material durchläuft
- e = Eulersche Zahl $e = 2,718\,281$
- I_{in} = Intensität der eintretenden Strahlen
- I_{out} = Intensität der austretenden Strahlen

Der Schwächungskoeffizient μ ist materialabhängig. Er kann aus den verschiedenen Koeffizienten der einzelnen Schwächungseffekte berechnet werden.

$$\mu = \tau + \sigma_{\text{klassisch}} + \sigma_{\text{compton}} + \chi$$

- τ = Koeffizient der Photoabsorption
- $\sigma_{\text{klassisch}}$ = Koeffizient der klassischen Streuung
- σ_{compton} = Koeffizient der Compton-Streuung
- χ = Koeffizient der Paarbildung

Die Gleichung (1) gilt allerdings nur für homogene Körper, das heißt Körper die aus einem Material bestehen. Der menschliche Körper ist aber aus vielen verschiedenen Materialien zusammengesetzt. Die unterschiedliche Schwächung in den einzelnen Schichten muß daher berücksichtigt werden. Ein Röntgenquant, das die erste Schicht mit μ_1 durchläuft, wird gemäß Gleichung (1) geschwächt. Danach durchquert es die zweite Schicht mit μ_2 und wird abermals geschwächt. Dabei ergibt sich für die Austrittsintensität $I_{\text{out}}^{(1)}$ nach der ersten Schicht die Gleichung

$$I_{\text{out}}^{(1)} = I_{\text{in}} \cdot e^{-\mu_1 L_1}.$$

Die Eintrittsintensität der zweiten Schicht ist die Austrittsintensität der ersten Schicht.

$$I_{\text{out}}^{(2)} = I_{\text{out}}^{(1)} \cdot e^{-\mu_2 L_2}$$

Die Gleichung der ersten Schicht wird in die Gleichung der Zweiten eingesetzt. Die Austrittsintensität $I_{\text{out}}^{(2)}$ berechnet sich nach

$$I_{\text{out}}^{(2)} = (I_{\text{in}} \cdot e^{-\mu_1 L_1}) \cdot e^{-\mu_2 L_2} = I_{\text{in}} \cdot e^{-\mu_1 L_1 - \mu_2 L_2}.$$

Für n Schichten mit den Durchlaufstrecken L_1, L_2, \dots, L_n und den Schwächungskoeffizienten $\mu_1, \mu_2, \dots, \mu_n$ ergibt sich die Gleichung

$$I_{\text{out}} = I_{\text{in}} \cdot e^{-\mu_1 L_1 - \mu_2 L_2 - \dots - \mu_n L_n}. \quad (2)$$

1.4 Probleme

In der Praxis treten Probleme auf, die die Qualität eines gescannten Bildes reduzieren. Eines der Hauptprobleme dabei ist das **Bildrauschen**. Dieses hat wiederum mehrere Ursachen. Der Hauptanteil wird durch das **Quantenrauschen** erzeugt.

- Die Röntgenstrahlung, die für die Untersuchung genutzt wird, besteht naturgemäß aus einer endlichen Anzahl von Quanten. Die Anzahl der gemessenen Quanten schwankt allerdings um einen Mittelwert, so daß ein Rauschen entsteht. Zur Lösung des Problems kann bei lebendem Gewebe im Rahmen der Strahlenschutzverordnung die Dosis erhöht werden.
- Ein weiteres Problem stellt die Röntgenröhre selbst dar. Die Strahlung, die sie erzeugt ist nicht monochromatisch. Eine exakte Messung ist jedoch nur möglich, wenn sie diese Eigenschaft besitzt, das heißt von einer Wellenlänge ist. (vgl. Abschnitt 1.3 auf Seite 2) Der Fehler läßt sich durch den Einsatz von Filtern vermindern, damit wird jedoch auch die Bildqualität reduziert, da sich hierdurch die Signalintensität verringert. Für ein genaues Bild ist aber ein hoher Signal-Rauschabstand nötig, was bedeutet, daß der Unterschied zwischen physikalisch bedingtem Rauschen und dem Signal groß sein muß.
- Durch die verschiedenen Wechselwirkungen eines Photons in der Elektronenhülle, zum Beispiel dem Comptoneffekt, werden die Bilder verrauscht. Eine übliche Gegenmaßnahme zur Beseitigung der Streustrahlung ist der Einsatz von Kollimatoren. Vereinfacht handelt es sich hierbei um Filter, die nur Photonen einer Richtung passieren lassen.

Neben dem Bildrauschen existieren verschiedene **Artefakte**, die im rekonstruierten Bild störend sind.

- Zum Beispiel treten Partialvoluminaeffekte auf, da durch Diskretisierung in endlich viele Schichten zwei Volumina mit unterschiedlichen Absorptionskoeffizienten in einem Voxel - dem Volumelement oder 3-D-Pixel - zusammengefaßt werden. Es wird mit einem gemittelten Koeffizienten gerechnet. Optisch stellen sich dann fließende Konturen statt den tatsächlich harten Übergängen dar.

2 Mathematik

Aus den physikalischen Grundlagen ergeben sich zwei unterschiedliche Problemstellungen, die auch einzeln erläutert werden sollen:

1. die Berechnung der Austrittsintensität eines Röntgenstrahls bei bekanntem Strahlengang und gegebenen Schwächungskoeffizienten der einzelnen Schichten. Nach dem Ermitteln der Durchlaufstrecken in den verschiedenen Materialien sind alle Größen der rechten Seite von Gleichung (2) bekannt, und die Austrittsintensität kann berechnet werden. Dieser Fall wird als *direktes Problem* bezeichnet.
2. die Berechnung der Schwächungskoeffizienten der durchquerten Schichten, welche Rückschlüsse auf die Materialart zulassen. Kennt man den Strahlengang und mißt die Ein- und Austrittsintensität eines Röntgenstrahls, dann sind in Gleichung (2) nur die Schwächungskoeffizienten unbekannt. Aus verschiedenen Strahlen, die alle eine solche Gleichung ergeben, versucht man die unbekannt Faktoren zu ermitteln. Dieser Fall, auch als *inverses Problem* benannt, ist das eigentliche Grundproblem der Computertomographie.

In diesem Abschnitt wird zunächst ein Modell erarbeitet, daß es ermöglicht, die Computertomographie mathematisch zu betrachten. Das Modell beinhaltet ein Schema, das heißt eine vereinfachte Form des tatsächlichen Vorgangs. Außerdem soll es helfen die physikalischen Prozesse mit der Mathematik zu verbinden. Anschließend werden formelle Inhalte geklärt, zum Beispiel verwendete Formelzeichen und sonstige Vereinbarungen, die für das Verständnis notwendig sind. Danach werden mit dem Modell einige Berechnungen durchgeführt, die als Grundlagen für weitere Problemlösungen dienen. Die nachfolgenden Betrachtungen beschäftigen sich zunächst mit dem *direkten* und dann mit dem *inversen Problem*. Der Schwerpunkt liegt beim *inversen Problem* und dem damit verbundenen Lösungsansatz.

2.1 Das mathematische Modell

Da die schichtweise Durchdringung eines Körpers durch die Röntgenstrahlen nur die Betrachtung eines Querschnittes erlaubt, erscheint es sinnvoll diesen als Grundlage des Modells zu wählen. Man legt den Körperquerschnitt in den Ursprung eines Koordinatensystems, wobei das Phantom selbst quadratisch sein soll und sich auf der x-Achse von $-n$ bis $+n$ erstreckt, was analog auf der y-Achse gelten soll. Die quadratische Form ermöglicht eine einfache Abstrahierung durch Matrizen und stellt sich auch weiterhin als praktikabler Ansatz heraus. Bei der Projektion eines Gitters mit der Maschenweite 1 wird der Querschnitt in Quadrate, insgesamt $2n \cdot 2n = 4 \cdot n^2$, eingeteilt. Diese sogenannten Pixel stellen die feinsten Strukturen unseres Körpers dar. Man geht davon aus, daß sie homogen

sind, also daß sich der Schwächungskoeffizient in ihnen nicht ändert. Diese Annahme ist eine notwendige Bedingung für die weiteren Berechnungen. Sie stimmt mit der Wirklichkeit, zum Beispiel verschiedener Zellgewebe als fast homogene Einheiten, gut überein. Die Anzahl der Pixel mit $4 \cdot n^2$ läßt erkennen, daß eine bessere Auflösung durch die Wahl eines größeren Parameters n erreicht werden kann. Es muß beachtet werden, daß damit auch der Rechenaufwand enorm ansteigt (vgl. Abschnitt 2.5.4 auf Seite 16).

Die Röntgenstrahlen sollen von einer Röntgenquelle Q ausgesandt werden, die sich im Abstand r zum Ursprung bewegt. Bei jeder Messung wird ein Bündel an Röntgenstrahlen emittiert, wobei die Auslenkung eines Strahles zum Zentralstrahl $\alpha \in \mathbf{R}\{-\pi/2; \pi/2\}$ beträgt. Dieser verläuft unter dem Winkel ϕ vom Punkt Q aus durch den Koordinatenursprung. Die Strahlen treten auf der einen Seite in das Phantom ein und treffen auf das Sensorfeld, welches der Röntgenquelle gegenüberliegt. Das Material außerhalb des Körpers soll den Röntgenstrahl nicht schwächen, was bei geeigneter Eichung auch praktisch umsetzbar ist.

Das folgende Schema zeigt noch einmal den grundlegenden Aufbau und relevante Größen:

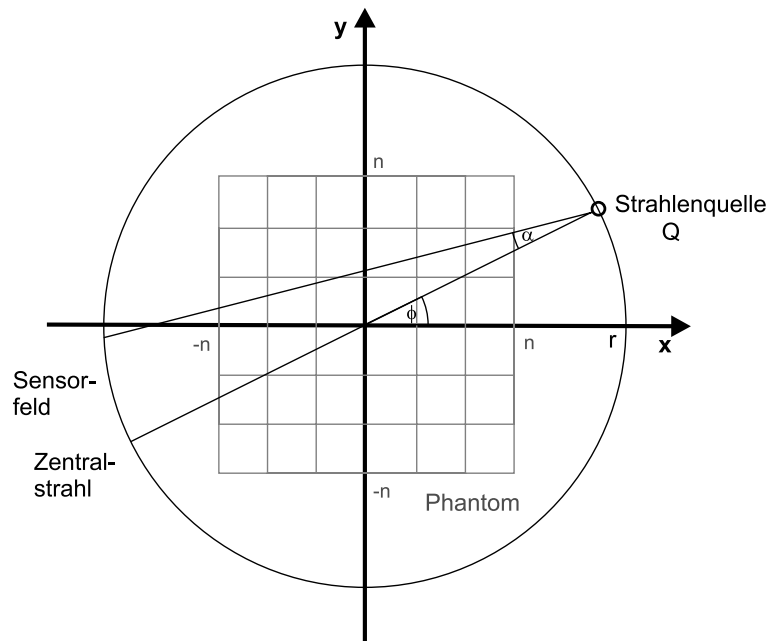


Abbildung 5: Aufbau des Modells

Pro Durchlauf sollen $step$ Messungen durchgeführt werden, das heißt die Quelle und die gegenüberliegenden Sensoren werden um den Winkel $\Delta\phi = \frac{2 \cdot \pi}{step}$ weiterbewegt. Besteht ein Strahlenbündel aus a Strahlen, dann werden insgesamt $a \cdot step$ Messungen erfaßt.

2.2 Formelle Vereinbarungen

Dieser Abschnitt soll in der Arbeit verwendete Formatierungen und Abkürzungen erläutern.

Formatierung

monospaced Text z.B. <code>Matlab</code>	...	gibt Eigennamen an
Kleinbuchstaben z.B. x	...	bezeichnen reellwertige Variablen
fettgedruckte Kleinbuchstaben z.B. wie \mathbf{v}	...	stehen für Vektoren
Großbuchstaben z.B. B	...	repräsentieren Matrizen
<code>>>dateiname.m<<</code>	...	<code>Matlab</code> -Programmdateien
Befehl	...	direkte <code>Matlab</code> -Befehle im laufenden Text

2.3 Erste Berechnungen

Die Aufgabenstellungen für das direkte und das inverse Problem erfordern beide zunächst die Berechnung der folgenden Größen:

- der Index der getroffenen Pixel, das heißt ihre Position im Phantom,
- die Länge der durchlaufenen Strecke in jedem Pixel.

Der Algorithmus zur Berechnung wurde im Programm `>>spunkt.m<<` (vgl. A.1) umgesetzt und soll im Folgenden beschrieben werden.

1. Für die weiteren Betrachtungen werden zunächst die Schnittpunkte der Strahlen mit dem Gitternetz benötigt. Hierzu wird für jeden Strahl, der durch $Q(\phi, r)$ und α gegeben ist, eine Punkt-Richtungs-Gleichung aufgestellt:

$$\frac{y - y_P}{x - x_P} = m \quad (3)$$

Wobei x_P und y_P die Koordinaten eines durchlaufenen Punktes und m der Anstieg der Geraden ist. Diese Steigung m ergibt sich aus $m = \tan(\phi + \alpha)$. Nach dem Umstellen von (3) erhält man für y :

$$y = m(x - x_P) + y_P \quad (4)$$

Und für x :

$$x = \frac{y - y_P}{m} + x_P \quad (5)$$

Nach dem Einsetzen von $y = -n, \dots, n$ in (5) ergeben sich die jeweiligen Schnittpunkte mit den vertikalen Achsenparallelen.

Diese Schnittpunkte werden in der Matrix X_S abgelegt. Die Spalten in der Matrix enthalten die x-Koordinate des Schnittpunktes eines Strahls. Um die Längen effizient berechnen zu können, ist eine aufsteigende Sortierung nach den x-Werten (X_S) und eine entsprechende Ermittlung der zugehörigen y-Werte (Y_S) nötig. Dazu werden die x-Koordinaten in eine weitere Matrix P_X übertragen und die Werte $x = -n, \dots, n$ ergänzt. Anschließend kann P_X aufsteigend in die Matrix S_X sortiert werden. Durch Einsetzen dieser Werte in (4) erhält man in der Matrix S_Y die entsprechenden y-Koordinaten. **Matlab** macht es möglich die Schnittpunktberechnungen simultan für das gesamte Strahlenbündel durchzuführen, weshalb die Spaltenanzahl der Matrizen S_X und S_Y mit der Anzahl der Strahlen im Bündel übereinstimmt.

- Die Indizes durchlaufener Pixel, das heißt die Positionen im Phantom, sind wichtig um später auf die richtigen Schwächungskoeffizienten zugreifen zu können. Zuerst werden für die getroffenen Pixel „Phantomkoordinaten“ ausgerechnet. Dieses gedachte Koordinatensystem beginnt in der linken unteren Ecke mit dem Pixel (1, 1); siehe dazu Abbildung 6 auf Seite 9. Um die Koordinaten zu erhalten wird der Mittelwert von zwei aufeinanderfolgenden Schnittpunkten der jeweiligen x- bzw. y-Werte gebildet, n addiert und anschließend auf die nächstgrößere ganze Zahl gerundet. Daraus kann nun der Index mit der Formel $i = (x + y \cdot 2 \cdot n) - 2 \cdot n$ berechnet werden. Allerdings ist es möglich, daß Pixel auch außerhalb des Phantoms liegen, nämlich dann wenn $x < 1$ oder $x > 2 \cdot n$ ist (analoges gilt für y). Dieser Effekt entsteht durch das Schneiden von $-n$ bis n . Sinnvollerweise werden diese Pixel bei den weiteren Betrachtungen ignoriert. Daher speichern wir die Koordinaten der entsprechenden Pixel und setzen die jeweiligen Indizes auf einen „harmlosen“ Wert, wie zum Beispiel 1.

(1;2n) $4n^2-2n$	(2;2n) ...	(2n;2n) $4n^2$
(1;2) $2n$	(2;2) $2n+1$	(2n;2) $4n-1$
(1;1) 0	(2;1) 1	(2n;1) $2n-1$

Abbildung 6: Schema zur Indexberechnung

3. Eine Strecke zwischen zwei Punkten A und B läßt sich mit Hilfe des Satzes des Pythagoras errechnen, es gilt: $\overline{AB} = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$. Nach Einsetzen der Schnittpunktkoordinaten in die obige Formel und Aussortieren außenliegender Pixel durch Nullsetzen der Länge, liegen die durchlaufenen Strecken in den einzelnen Pixeln des Phantoms von links nach rechts vor.

2.4 Das direkte Problem

Das *direkte Problem* stellt uns vor die Aufgabe Austrittsintensitäten bei gegebenen Eingangsgrößen, das heißt Eingangsintensität und Schwächungskoeffizienten, zu berechnen. Da die Ermittlung der Größen Länge und Index schon im vorigen Abschnitt geklärt wurde, soll es jetzt um konkrete Datenorganisation und die Erstellung einer Computersimulation mit Hilfe des Mathematikpakets `Matlab`¹ gehen.

Datenorganisation

Als erstes muß eine effiziente Abstrahierung der vorhandenen Daten diskutiert werden. Besonders die Fähigkeiten der Matrizenmanipulation des Programmes `Matlab` sollten dabei zum Zug kommen. Da für das Phantom eine quadratische Form gewählt wurde, kann man die Schwächungskoeffizienten in einer $2n \times 2n$ Matrix anordnen:

$$SM = \begin{pmatrix} \mu_{1,1} & \mu_{2,1} & \dots & \mu_{2n,1} \\ \mu_{1,2} & \mu_{2,2} & \dots & \mu_{2n,2} \\ \vdots & \vdots & \ddots & \\ \mu_{1,2n} & \mu_{2,2n} & \dots & \mu_{2n,2n} \end{pmatrix}$$

Die Indizes an den Koeffizienten geben die Phantomkoordinaten an. Um aber nicht beide Koordinaten speichern zu müssen, wandeln wir die Matrix in eine $1 \times 4n$ Matrix um, indem die Zeilen aneinandergehängt werden. Jetzt kann man über den errechneten Index auf den Schwächungskoeffizienten im entsprechenden Pixel zugreifen: `si = SM(i)`. Wenn die Indizes in einem Vektor, zum Beispiel \mathbf{i} abgelegt werden und damit auf die Schwächungsmatrix zugegriffen wird, dann liefert der obige Befehl auch einen Vektor mit den zugehörigen Koeffizienten zurück. Somit kann die Berechnung nach der Formel (2) für einen Strahl, dessen durchlaufene Pixel bzw. deren Index im Vektor \mathbf{b} und die entsprechenden Längen im Vektor \mathbf{l} liegt, folgendermaßen implementiert werden: `Iout = exp(sum(-SM(b).*l))*Iin`. Der angegebene Befehl funktioniert auch, wenn \mathbf{b} und \mathbf{l} Matrizen sind. Jede Zeile der Matrix entspricht dann dem Index- bzw. Längenvektor für einen Strahl aus dem Bündel. Das erhaltene I_{out} ist nun keine reelle Zahl mehr, sondern ein Spaltenvektor, der die Austrittsintensitäten für jeden Stahl enthält.

¹Matlab ist ein eingetragenes Warenzeichen der The MathWorks Inc.

Für die weitere Implementation ist die Vorgabe einer Schwächungsmatrix notwendig. Dazu wurden die Hilfsprogramme »*make_cross.m*« und »*make_object.m*« (vgl. A.4 und A.2) entwickelt. Ersteres erzeugt eine quadratische Matrix angegebener Größe und besetzt sie so, daß der entstehende Querschnitt wie ein Schweizer Kreuz aussieht.

$$SC = \begin{pmatrix} 0 & \mu & \mu & 0 \\ \mu & \mu & \mu & \mu \\ \mu & \mu & \mu & \mu \\ 0 & \mu & \mu & 0 \end{pmatrix}$$

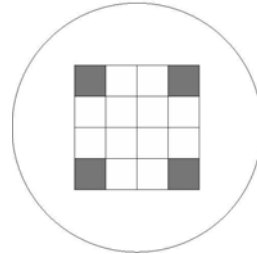


Abbildung 7: Schweizer Kreuz in der Simulation

Das zweite Programm bietet die Möglichkeit ein Phantom mit Hilfe einer „graphischen Oberfläche“ zu erzeugen. Damit stehen verschiedenste Testobjekte zur Verfügung.

Die Simulation beginnt mit der Darstellung des Phantoms und der Strahlenquelle (vgl. A.3). Anschließend wird das Modul »*spunkt.m*« ausgeführt. Die Berechnung der Austrittsintensitäten für das aktuelle Bündel erfolgt wie beschrieben. Die Strahlenquelle wird um den Winkel $\frac{2 \cdot \pi}{step}$ auf dem Kreis weiterbewegt, so daß sich für $\phi = \phi + \frac{2 \cdot \pi}{step}$ ergibt. Der Algorithmus wird solange wiederholt, bis die Quelle einen vollen Kreis beschrieben hat. Die errechneten Intensitäten werden zwischengespeichert und können so später weiterverwendet werden.

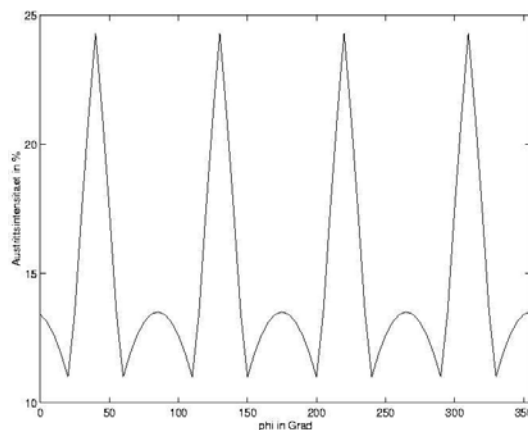


Abbildung 8: Schwächungskurve

Betrachtet man Abbildung 8 auf Seite 11 so zeigt sich die Abhängigkeit der Austrittsintensität von der Auslenkung des Zentralstrahls, wobei hier die Schwächung desselben ausgewertet wurde. Das benutzte Phantom war ein „Schweizer Kreuz“ mit $n = 2$ (vgl. Abbildung 7). Man erkennt deutlich vier „Spitzen“, das heißt es gibt Winkel unter denen die Abschwächung am geringsten ist. An diesen Stellen hat der Strahl die kürzeste Strecke unter Schwächung zurückgelegt. Diese „Spitzen“ repräsentieren sozusagen die vier fehlenden Ecken des Phantoms. Die Kurve beschreibt den Aufbau des untersuchten Objekts. Die Rekonstruktion eines Bildes aus diesen Meßdaten ist Thema des nächsten Abschnittes.

Das *direkte Problem* hat uns bei der Einarbeitung in die Materie geholfen. Wir erhielten so einen konkreten Einstieg in `Matlab` und schufen gute Voraussetzungen für das anstehende *inverse Problem*.

2.5 Das inverse Problem

Das *inverse Problem* stellt die eigentliche Grundfrage der Computertomographie dar. Es sollen aus mehreren Strahlenverläufen, den eindimensionalen Projektionen, Rückschlüsse auf die Zusammensetzung eines Körpers gezogen werden, die für die Erzeugung eines zweidimensionalen Querschnittbildes notwendig sind. Um das zu erreichen, wird auf Gleichung (2) zurückgegriffen, die alle Schwächungskoeffizienten der von einem Strahl durchdrungenen Pixel enthält. Nach dem Umstellen ergibt sich folgende Formel:

$$\ln \frac{I_{\text{out}}}{I_{\text{in}}} = -\mu_1 L_1 - \mu_2 L_2 - \dots - \mu_n L_n \quad (6)$$

Die Faktoren $\mu_1 \dots \mu_n$ kommen darin nur linear vor. Man könnte also aus n solchen Gleichungen ein lineares Gleichungssystem bilden, dessen Lösung die gesuchte Zusammensetzung liefert. Bei einer Auflösung von 20×20 ergeben sich 400 Unbekannte; es sind also 400 Gleichungen notwendig. Doch schon bei einer Schrittweite von $step = 60$ ($\Delta\phi = 6^\circ$) und 9 Strahlen im Bündel erhält man $9 \cdot 60 = 540$ Gleichungen. Dabei stellt sich die Frage, warum $\Delta\phi$ bzw. $step$ nicht so wählt wird, daß die Zahl der Gleichungen der Anzahl der Unbekannten entspricht. Diese Forderung ist technisch kaum umzusetzen und würde die Messung nur unnötig verlängern. Außerdem ergeben sich durch mehr Gleichungen verschiedene Vorteile bei der Bildrekonstruktion (vgl. 2.5.4).

Man benötigt somit ein Verfahren um überbestimmte Gleichungssysteme lösen zu können. Ein weiteres Problem ist die Dimensionierung des gewünschten Bildes, denn schon die Verdopplung der Auflösung bedeutet eine Vervierfachung der Anzahl an Unbekannten und damit auch eine deutliche Erhöhung des Rechenaufwandes (vgl. 2.5.4).

2.5.1 Lösen von überbestimmten Gleichungssystemen

Die „Lösung“ eines solchen Systems wird nach der Methode der sogenannten *Kleinsten Quadrate* bestimmt. Diese soll nun an einem Beispiel erläutert werden. Jede Gleichung eines linearen Gleichungssystems kann man sich als Hyperebene vorstellen. Bei zweidimensionalen Problemen ist dies eine Gerade, bei dreidimensionalen Problemen eine Ebene und so weiter. Jeder Punkt, der auf der Hyperebene liegt, stellt damit eine Lösung der Gleichung dar. Der Schnittpunkt zweier Geraden, die Teil eines Gleichungssystems mit zwei Unbekannten sind, löst also beide Gleichungen. Demnach besitzt ein Gleichungssystem dann eine Lösung, wenn sich alle Hyperebenen in einem Punkt schneiden. Bei überbestimmten Gleichungssystemen ist das nicht mehr zu garantieren. Betrachtet man zum Beispiel

$$\begin{aligned}y &= 0.2 \\x - y &= 0 \\5x + y &= 1\end{aligned}$$

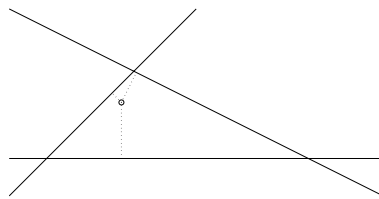


Abbildung 9: Graphische Darstellung des Beispiels

(vgl. Abbildung 9 auf Seite 13) so erkennt man, daß keine Lösung existiert. Man kann aber für jeden Punkt $(x_1; x_2)$ messen, wie „gut“ er die Gleichungen erfüllt. Dabei ist $0.2 - x_2$ der Fehler in der ersten Gleichung, $0 - x_1 - x_2$ in der zweiten und $1 - 0.5 \cdot x_1 - x_2$ der Fehler in der dritten Gleichung. Um zu beurteilen, wie gut $(x_1; x_2)$ alle drei Gleichungen erfüllt, verwenden wir als Maß $r = (0.2 - x_2)^2 + (0 - x_1 - x_2)^2 + (1 - 0.5 \cdot x_1 - x_2)^2$. Offensichtlich gäbe es nur dann einen Punkt $(x_1; x_2)$ mit $r = 0$, wenn das Gleichungssystem eine Lösung besitzen würde. Da dies in unserem Beispiel nicht zutrifft, wollen wir $(x_1; x_2)$ jetzt so wählen, daß r minimal ist. Es ergibt sich

$$x_1 = 0.6 \quad , \quad x_2 = 0.5$$

mit $r = (-0.3)^2 + (-0.1)^2 + (0.2)^2 = 0.14$. Für jeden anderen Punkt $(\tilde{x}_1; \tilde{x}_2)$ ist $\tilde{r} > 0.14$.

Diese „optimale Lösung“ bestimmt man durch das Lösen der Gaußschen Normalgleichungen. Schreibt man unser überbestimmtes Anfangssystem in der allgemeinen Form

$$A\mathbf{x} = \mathbf{b},$$

dann gibt es im allgemeinen kein \mathbf{x} mit $\mathbf{b} - A\mathbf{x} = 0$. Also bestimmen wir \mathbf{x} so, daß der *Residualvektor* $\mathbf{r} = \mathbf{b} - A\mathbf{x}$ eine möglichst kleine Länge besitzt. Gesucht ist ein Vektor \mathbf{x}^* für den $A\mathbf{x}^*$ nahe an der rechten Seite von \mathbf{b} liegt.

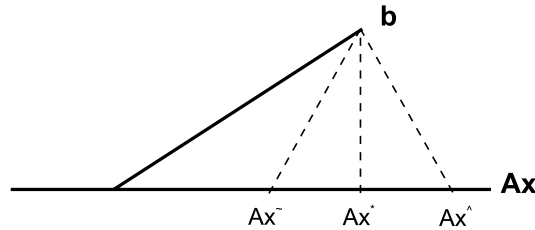


Abbildung 10: Schema für die beste Näherungslösung

Die Abbildung 10 zeigt, daß die Lösung durch folgende Beziehung charakterisiert ist

$$(\mathbf{b} - A\mathbf{x}^*) \perp A\mathbf{x} \quad (\text{für alle } \mathbf{x}).$$

Dabei bedeutet $\mathbf{v} \perp \mathbf{w}$, daß die Vektoren \mathbf{v} und \mathbf{w} senkrecht aufeinander stehen. Diese Bedingung ist erfüllt, wenn das Skalarprodukt gleich Null ist, also für

$$A\mathbf{x} \bullet (\mathbf{b} - A\mathbf{x}^*) = 0$$

Das Skalarprodukt zweier Vektoren \mathbf{x} und \mathbf{y} ist als Summe über die Produkte der Einzelkomponenten $x_i \cdot y_i$ definiert: $\mathbf{x} \bullet \mathbf{y} = \sum_{i=1}^n x_i \cdot y_i$. Damit kann es auch durch das Produkt der transponierten Matrix \mathbf{x}^T und \mathbf{y} ausgedrückt werden: $\mathbf{x}^T \mathbf{y} = \mathbf{x} \bullet \mathbf{y}$. Auf unsere ursprüngliche Forderung angewandt, ergibt sich

$$(A\mathbf{x})^T (\mathbf{b} - A\mathbf{x}^*) = 0$$

Nach dem Umstellen und Ausmultiplizieren erhält man $\mathbf{x}^T A^T (\mathbf{b} - A\mathbf{x}^*) = 0$ oder $\mathbf{x}^T (A^T \mathbf{b} - A^T A\mathbf{x}^*) = 0$. Diese Gleichung soll für alle \mathbf{x} gelten, was äquivalent ist zu

$$A^T \mathbf{b} - A^T A\mathbf{x}^* = 0$$

Man erhält die *Gaußsche Normalgleichung*

$$A^T A\mathbf{x}^* = A^T \mathbf{b} \tag{7}$$

Die Anzahl der Gleichungen dieses Systems entspricht der Anzahl der Unbekannten, da $A^T A$ eine quadratische Matrix ist. Damit kann man jetzt versuchen die

Lösung zu bestimmen. Dieses \mathbf{x}^* erhält man durch $\mathbf{x}^* = (A^T A)^{-1} A^T \mathbf{b}$. Außerdem ist zu erkennen, daß die Beziehung auch dann gilt, wenn das ursprüngliche System $A\mathbf{x} = \mathbf{b}$ lösbar ist. In diesem Fall ergibt sich die exakte Lösung.

Durch die Gaußsche Normalgleichung haben wir eine Möglichkeit, das überbestimmte Gleichungssystem zu lösen. Es zeigt sich jedoch, daß diese Methode nicht immer den effizientesten Ansatz darstellt.

2.5.2 Iterationsverfahren

Wie beschrieben, besitzt die Gaußsche Normalgleichung auch Nachteile. Betrachten wir dazu folgendes Beispiel, das dem *inversen Problem* entnommen wurde. Grundlage bildet ein „Schweizer Kreuz“-Phantom mit $n = 10$. Es werden insgesamt 90 Messungen durchgeführt, das heißt eine Messung aller 4° . Das Strahlenbündel enthält 9 Strahlen. Man erhält 810 Gleichungen bei 400 Unbekannten. Die Abbildung 11 auf Seite 15 zeigt die Besetzung der 810×400 Matrix A und der 400×400 Matrix $A^T A$.

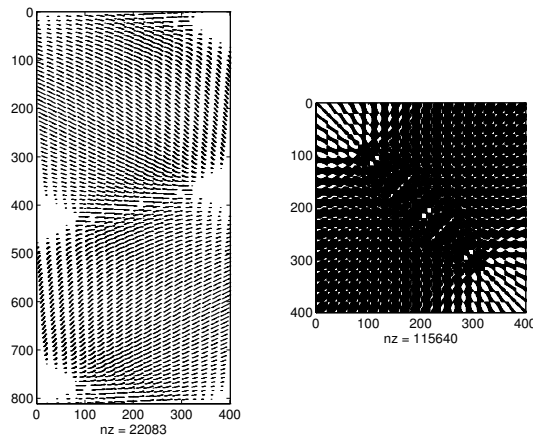


Abbildung 11: Besetzung des Gleichungssystems

Wie zu erkennen ist, besitzt die Ausgangsmatrix A nur 22083 von Null verschiedene Einträge, während $A^T A$ 115640, also mehr als fünf mal so viel solche Einträge beinhaltet. Das führt zu erheblichen Speicherproblemen, sobald größere Matrizen gewählt werden. Außerdem hat die Lösung von $A^T A \mathbf{x}^* = A^T \mathbf{b}$ durch Gaußelimination eine Komplexität von $\dim(A)^3$, was im Beispiel etwa 64 Millionen Rechenoperationen erfordert.

Man ist bemüht Verfahren zu finden, die ohne Verwendung der Normalgleichung direkt mit A operieren und dabei die senkrechte Beziehung ausnutzen. Interessant sind Algorithmen bei denen die Koeffizientenmatrix A nur über Matrix-Vektor-Operationen eingeht. Dies führt zu Iterationsverfahren, bei denen ausgehend von einem Startwert x_0 sukzessive bessere Näherungen x_1, x_2, x_3, \dots berechnet wer-

den, wobei die wesentliche Operation beim Übergang von x_n zu x_{n+1} eine Multiplikation mit A ist. Ein solches Verfahren ist zum Beispiel das **Kaczmarz-Verfahren**.

2.5.3 Simulation

Die Simulation unterscheidet sich nicht wesentlich von der des *direkten Problems*. Zunächst wird ein Testobjekt, also eine Schwächungsmatrix, geladen und die Meßwerte aufgenommen, um den realen Meßvorgang nachzuahmen. Nach dem Erfassen der Werte, wird die vorgegebene Schwächungsmatrix aus dem Speicher entfernt und anschließend das Gleichungssystem gelöst. Dabei kann der Algorithmus zur Gaußschen Normalgleichung angewandt werden.

Im Folgenden soll ein Beispiel für die Rekonstruktion durch die beschriebenen Algorithmen angeführt werden. Das gestellte Problem beinhaltet 400 Unbekannte (20×20 Phantom) und 810 Gleichungen (90 Schritte bei 9 Strahlen). Abbildung 12 zeigt das Ausgangsphantom und Abbildung 13 das rekonstruierte Phantom. Erkennbare Unterschiede sind dabei auf Rechenungenauigkeiten, wie zum Beispiel Rundungen, zurückzuführen.

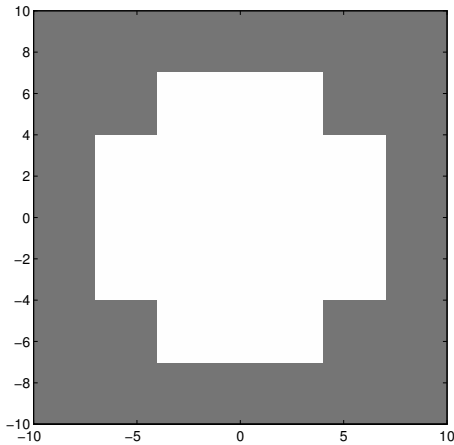


Abbildung 12: Gegebenes Phantom

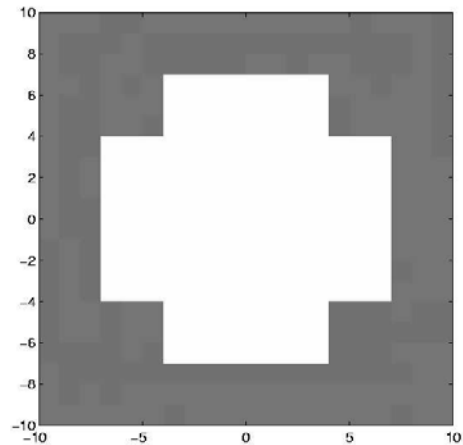


Abbildung 13: Rekonstruiertes Bild

2.5.4 Probleme

Schon bei unserer „kleinen“ Simulation sind wir auf verschiedene Probleme gestoßen, die auch in der Praxis Bedeutung haben.

Vor allem die folgenden wären dabei zu erwähnen:

- die mit der Auswertung verbundene Rechenzeit - hauptsächlich für die Lösung des Gleichungssystems. Zum Lösen dieses Problems kann man zum Beispiel auf die schon angesprochenen Iterationsverfahren zurückgreifen.

- die Auswirkungen von (kleinen) Meßfehlern bei inversen Problemen. Zur Untersuchung der möglichen Effekte können kleinere Meßfehler simuliert und die gewonnenen Ergebnisse ausgewertet werden. Daraus lassen sich dann Möglichkeiten zur Reduzierung des Effektes erarbeiten.

Die Abbildungen 14 und 15 zeigen in dem Zusammenhang die rekonstruierten Bilder bei einem simulierten Fehler von etwa 3 Prozent. In beiden Fällen wurde das Ausgangsphantom von Abbildung 12 mit einer Größe von 20×20 verwendet. Zur Erzeugung von Bild 14 setzten wir 60 Schritte ein, was einer Anzahl von 540 Gleichungen entspricht, und bei Bild 15 wurden 90 Schritte mit 810 Gleichungen benutzt.

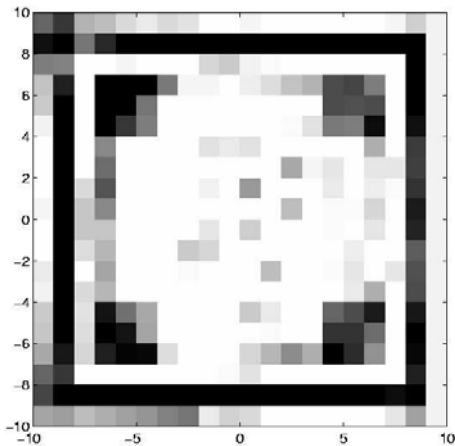


Abbildung 14: Bild bei 540 Gleichungen

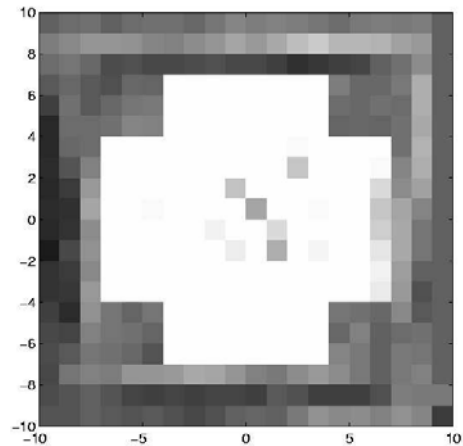


Abbildung 15: Bild bei 810 Gleichungen

Wie deutlich zu erkennen ist, hat sich die Qualität von Bild 14 zu Bild 15 gesteigert. Die größere Anzahl an Gleichungen hat die Rekonstruktion also verbessert. Die Abbildungen 16 und 17 auf Seite 18 verdeutlichen diesen Zusammenhang noch einmal. Sie stellen die Abweichung des berechneten Phantoms zum Ausgangsphantom dar.

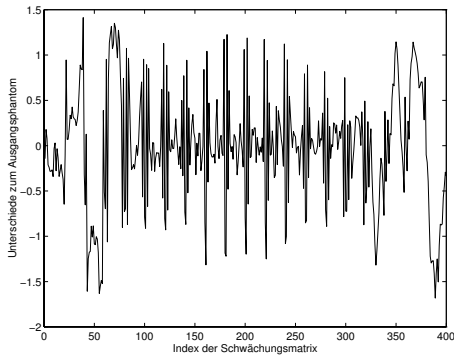


Abbildung 16: Unterschiede bei 540 Gleichungen

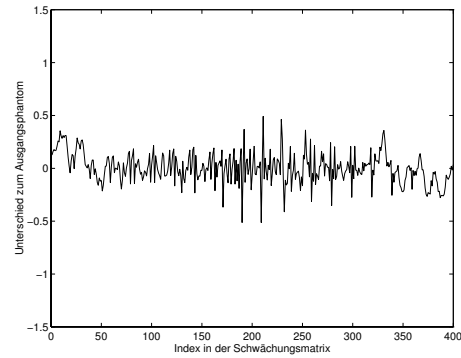


Abbildung 17: Unterschiede bei 810 Gleichungen

2.6 Bedienung der graphischen Oberfläche

In diesem Kapitel wird anhand von Beispielen dargestellt, wie der Nutzer die graphische Oberfläche für das direkte bzw. inverse Problem benutzen kann. Mit Hilfe von „Screenshots“ werden dem Benutzer die einzelnen Schaltflächen und Eingabebereiche erklärt. Die Oberfläche wurde mit Hilfe des `Matlab` Paketes erzeugt, und alle Berechnungen werden über `Matlab` ausgeführt.

2.6.1 Interface - direktes Problem

Das Programm simuliert das direkte Problem. Es wird ein Körper vorgegeben (z.B. Schweizer Kreuz) und die dazugehörigen Schwächungswerte werden berechnet. Vereinfacht bedeutet dies, daß zu einer bestimmten Anfangsintensität die Schwächung durch den Körper berechnet wird. Der Vorgang kann für beliebig viele Strahlen unter beliebigen Winkeln wiederholt werden. Das Programm erzeugt nach einem Durchlauf einen Vektor in dem die Intensitäten abgespeichert sind.

An dieser Stelle sei darauf hingewiesen, daß die theoretisch mögliche Anzahl beliebig vieler Strahlen praktisch nicht umsetzbar ist. Sie hängt hauptsächlich von der Geschwindigkeit des eingesetzten Rechnersystems ab. Eine zu hohe Strahlenanzahl kann zu langen Wartezeiten führen. Dies sollte bei der Anwendung beachtet werden, da auch die entstehende Matrix einen nicht unerheblichen Bedarf an Arbeitsspeicher benötigt (vgl. Abschnitt 2.5.4 auf Seite 16).

Die Oberfläche ist in zwei Bereiche unterteilt. Im rechten Teil befindet sich die Steuerleiste zum Eingeben der Parameter. Im linken Teil des Fensters befindet sich die Ausgabe. Beim Start des Programms sind bereits Parameter vorgegeben. Falls diese beibehalten werden sollen, dann kann die Simulation sofort über den `Start`-Button gestartet werden.

Zum Ändern der **Parameter** wird die Maus in das Feld bewegt. Durch einen Mausklick wird das Feld aktiviert - der gewünschte Wert kann eingetragen werden.

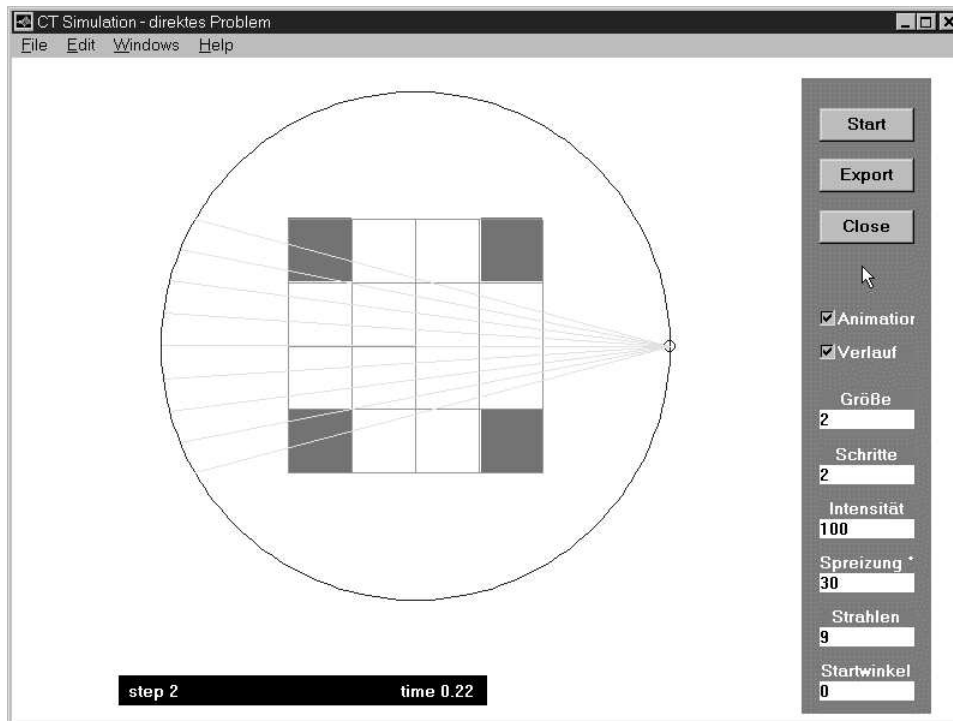


Abbildung 18: Benutzeroberfläche des direkten Problems

Bedeutung der Parameter

Größe Mit diesem Parameter wird die Größe des Objektes angegeben. Dabei ist zu beachten, daß der eingegebene Wert quadratisch in das Ergebnis einfließt. Der Wert 4 erzeugt einen Körper von 8×8 Pixeln.

Schritte Der Parameter „Schritte“ gibt an, in wieviel Schritten das Strahlenbündel das Objekt umkreist. Der Wert 4 bewirkt somit eine Messung im Abstand von 90° .

Intensität Durch die Angabe der „Intensität“ kann die Größenordnung der Schwächungswerte geändert werden. Das Verhältnis Eintritts-Austrittsintensität bleibt trotzdem erhalten.

Spreizung Die Quelle sendet gewöhnlich mehrere Strahlen aus. Durch die Wahl des Parameters kann der Überdeckungswinkel des Strahlenbündels bestimmt werden. Dabei bezieht sich die Angabe in Grad auf den Winkel zwischen den beiden äußeren Strahlen.

Strahlen Neben dem Zentralstrahl können weitere Strahlen erzeugt werden, um Meßergebnisse aufzunehmen. Der Wert 9 erzeugt zusätzlich zum Zentral-

strahl 8 weitere Strahlen. Bei der Angabe der Strahlen ist auf einen ungeraden Wert zu achten.

Startwinkel Durch die Angabe dieses Parameters kann bestimmt werden, wo sich die Quelle zum Zeitpunkt des Starts befindet. Die Angabe von 90° positioniert die Quelle genau senkrecht über dem Objekt.

Neben der Möglichkeit zur Veränderung der Parameter wurden noch zwei Schalter eingerichtet. Durch Klicken mit der Maus auf das jeweilige Kästchen können diese aktiviert bzw. deaktiviert werden. Der erste Schalter ist der **Animation**-Schalter. Wenn dieser aktiviert wird, kann der Benutzer die einzelnen Rechenschritte der Simulation beobachten. Da dies jedoch zeitaufwendig ist, sollte - falls nur die Schwächungsintensitäten gesucht sind - auf diese Option verzichtet werden. Darunter befindet sich ein Schalter, der mit **Verlauf** gekennzeichnet ist. Durch die Aktivierung kann nach durchlaufener Simulation eine Schwächungskurve dargestellt werden. Die berechneten Daten können in einer Matrix fest abgespeichert werden. Dazu ist nach beendeter Simulation der **Export**-Button zu drücken. Die Daten werden in der Datei `>direct.mat<` gespeichert.

Statuszeile Im unteren Teil der Ausgabefläche befinden sich noch zwei Statuswerte, die dem Benutzer einen Überblick über den aktuellen Stand der Simulation liefern. Die Ausgabe des **step**-Wertes gibt die Anzahl der Schritte an, die schon abgearbeitet wurden. Mit Hilfe des **time**-Wertes kann die benötigte Rechenzeit des Programms kontrolliert werden.

2.6.2 Interface - inverses Problem

Für die Berechnung des inversen Problems wurde die Oberfläche nur in einzelnen Teilpunkten geändert. So kann der Benutzer zum Beispiel sein Objekt vorgeben, daß direkte Problem lösen und anschließend aus den sich ergebenden Werten das inverse Problem lösen.

Grundlegend wurden folgende Punkte gegenüber der Oberfläche vom direkten Problem geändert.

- Die Parameter werden deaktiviert, da diese für das inverse Problem nicht benötigt werden, zumindest beim Import von Meßdaten.
- Um dem Anwender die Möglichkeit des Ladens einer vorgegebenen Datei zu ermöglichen, wird der **Export**-Button in einen **Load**-Button umgewandelt. Weiterhin existiert ein **Solve**-Button, der den **Start**-Button ersetzt.

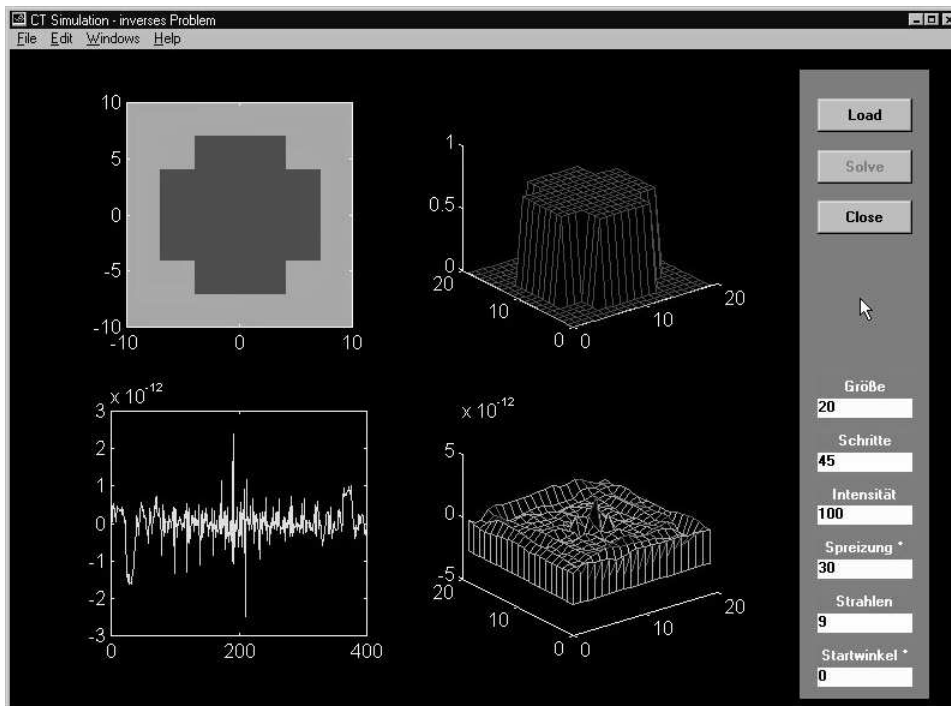


Abbildung 19: Benutzeroberfläche des inversen Problems

3 Anwendung

Über hundert Jahre ist es nun her, seit Wilhelm Conrad Röntgen 1895 als Professor für Physik an der Universität Würzburg die X-Strahlen entdeckt hat. Jedoch erst 1968 gelang es Godfrey N. Hounsfield und Allan M. Cormack den ersten Computertomographen zu entwickeln. Dafür erhielten sie 1979 den Nobelpreis für Physiologie und Medizin (siehe dazu auch [9]).



Abbildung 20: Allan M. Cormack
1924-1998 (aus [8])



Abbildung 21: Godfrey N. Hounsfield
1919 - (aus [8])

Das Grundprinzip der Computertomographen hat sich bis heute nicht verändert.

Eine Röntgenquelle bewegt sich schrittweise um den Patienten, während auf der gegenüberliegenden Seite die geschwächten Röntgenstrahlen mit einem Detektorfeld aufgefangen werden. Dieses erlaubt die Erstellung eines Meßprofils, das heißt, eine Auswertung verschieden geschwächter Strahlen.

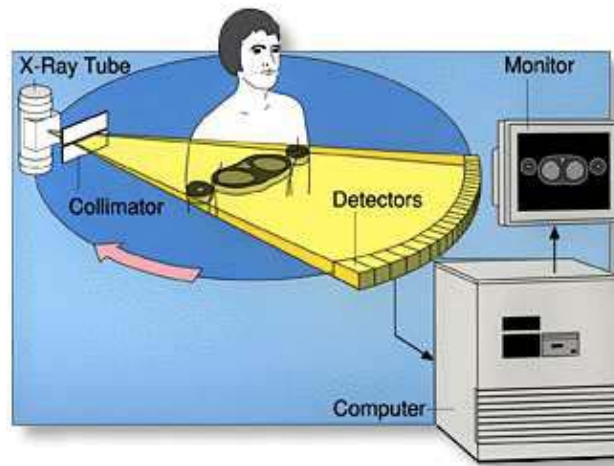


Abbildung 22: Funktionsprinzip eines Computertomographen (aus [4])

In nur wenigen Jahren wurden die Computertomographen schnell weiterentwickelt. Während die erste Generation noch eine Meßzeit von 5 Minuten benötigte, verringerte sich die Meßzeit über die Jahre auf lediglich 1-2 Sekunden. Auch die Anzahl der installierten Detektoren veränderte sich. Wurde anfangs lediglich 1 Detektor eingesetzt, so benötigt man heute für präzise Messungen über 1000 Detektoren. Neben den für die Messung eingesetzten Geräten erfordert die moderne Computertomographie weitere technische Komponenten. Für die Erzeugung der Röhrenspannung ist ein Hochspannungsgenerator nötig, der eine stabile Spannung liefern muß. Außerdem benötigt man zur Rekonstruktion der Bilder einen Hochgeschwindigkeitsrechner, der die anfallenden Gleichungssysteme schnell und effizient lösen kann. Auch an die Software, die solch ein System betreibt, werden hohe Anforderungen gestellt. So wird zum Beispiel ein Algorithmus benötigt, der die Erfassung der Meßdaten übernimmt und die Bildrekonstruktion bzw. Bildverarbeitung realisiert. Die Größe der einzelnen Bildelemente (Pixel) liegt derzeit bei $0,4 \times 0,4$ mm. Die Schichtdicke der Messung beträgt zwischen 1 und 12 mm. Die Darstellung der einzelnen Schicht erfolgt mit einer Auflösung von 256×256 bzw. 512×512 Pixeln. Die Genauigkeit und der Fakt, daß die Lage von Organen und der Unterschied von gesundem und krankem Gewebe bestimmt werden kann, revolutionierte die Radiologie. Es war erstmals möglich, akut kranke Personen zu untersuchen, ohne dabei weitere Belastungen zu erzeugen, die das Leiden verschlimmern könnten. Die Dichte- bzw. Absorptionsmessung erfolgt Hounsfield zu Ehren in der Hounsfield - Einheit (HU). Dabei wird davon ausgegangen, daß der Absorptionswert von Geweben und die physikalische Dichte linear abhängig sind.



Abbildung 23: Typisches Schädel-CT Bild (aus [7])

Die Dichtewerte werden nach folgender Gleichung berechnet:

$$\text{HU} = \frac{\mu - \mu_{H_2O}}{\mu_{H_2O}} \cdot 1000$$

Mit Hilfe der Hounsfield-Skala werden die Schwächungskoeffizienten verschiedener Körpergewebe berechnet. Für Luft ergibt sich ein Wert von -1000 HU, Blut besitzt einen Wert von $+12$ HU, und Knochen einen Wert von $+200$ bis $+1000$ HU.

Die Computertomographie hat derzeit einen Stand erreicht, der für „normale“ Untersuchungen ausreichen würde. Die Scan- bzw. Berechnungszeiten liegen lediglich im Sekundenbereich. Jedoch ist eine Untersuchung des Brustkorbes oder des Herzens schwer möglich, da diese durch Atmung und Herzschlag beeinflusst werden und somit das Bild „verwackeln“. Derzeit arbeitet man an Ultra-Fast-CTs (UFCT), die dennoch eine Aufnahme solcher Bereiche möglich machen. Die Scanzeiten beim UFCT sind dabei soweit verkürzt worden, daß die Bewegung der Herzmuskeln oder des Brustkorbes das Bild nicht mehr beeinträchtigen.

Technische Angabe	Mittelklasse CT	Hochleistungs CT
Meßfeld	45 cm	50 cm
Strahlenfächer	42 Grad	50 Grad
Detektorenmaterial	Xenon	Xenon
Detektorenanzahl	850	1300
Meßwerte (512 x 512)	110 000	1 000 000
Meßprofile (512 x 512)	130	800
Schichtdicke	2, 4, 6, 8, 10	1, 3, 5, 8, 10
Meßzeit 240 Grad	2,5 s	1,2 s
Meßzeit 360 Grad	4,5 s	1,9 s
Bildrechenzeit	14 s	4 s
CT-Werte-Skala	-1000 bis 3000 HU	-2000 bis 4000 HU
Bildpunktgröße	0,1 - 1 mm	0,05 - 1,5 mm
Röhrendrehung	9000 U/min	9000 U/min
Funktionsgarantie	60 000 Aufnahmen	60 000 Aufnahmen

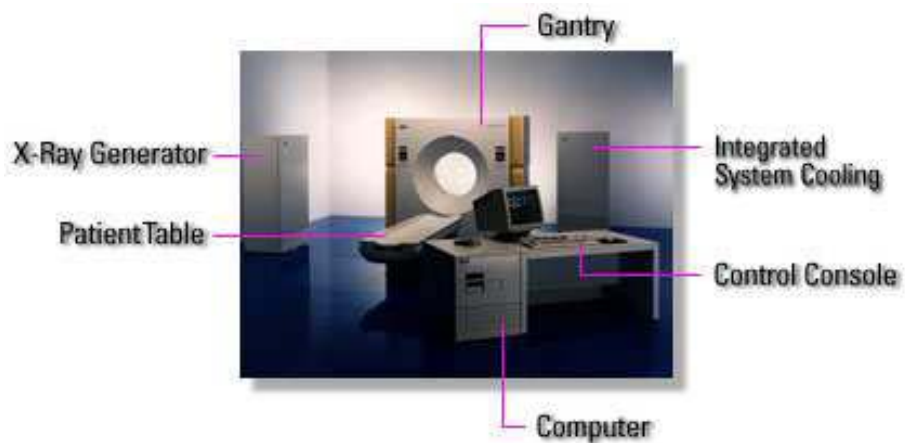


Abbildung 24: Das von Siemens entwickelte Gerät SOMOTOM Plus 4(aus [4])

4 Schlußbetrachtungen

Am Beispiel der Computertomographie kann man erkennen, wie wichtig es ist, verschiedene Wissenschaftszweige miteinander zu verbinden. Die Realisierung der CT-Geräte wurde überhaupt erst durch das Zusammenspiel von Physik, Mathematik und Technik möglich.

Mit dieser Arbeit wollen wir einen Einblick in die Computertomographie und die Problematik, ein praktisches Problem mathematisch zu erschließen, geben. Sie soll außerdem einen Überblick über die heutige Technik liefern und helfen, die dahinter stehende Theorie greifbar zu machen. Es bietet sich auch ein breites Spektrum an Fortsetzungsmöglichkeiten an. So könnte man zum Beispiel das mathematische Modell so verbessern, daß die Berechnung schneller erfolgt. Zusätzlich ist eine 3-dimensionale Rekonstruktion möglich. Zur Veranschaulichung und Überprüfung der aufgestellten Ansätze wäre der Aufbau einer Versuchsanordnung, die beispielsweise mit sichtbarem Licht funktioniert, denkbar.

Die heute in der Praxis eingesetzten Methoden zur Rekonstruktion sind sehr kompliziert und erfordern den Einsatz von hochspezialisierten Geräten. Die Weiterentwicklung der Tomographen ist seit Hounsfield ein anhaltender Prozess und selbst heute ist ein Ende dieser Entwicklung nicht abzusehen. Derzeit wird nach immer schnelleren und exakteren Verfahren gesucht, die es den Ärzten erlauben Patienten noch besser zu versorgen. Hier sind wieder die Mathematiker gefragt, neue Möglichkeiten aufzuzeigen, welche schließlich durch die Technik eine Anwendung finden.

Abschließend möchten wir uns bei der TU Bergakademie Freiberg und besonders unserem Betreuer Professor Michael Eiermann für die Unterstützung bei diesem Projekt bedanken.

Literatur

- [1] Theodor Laubenberger, *Technik der medizinischen Radiologie*, Deutscher Ärzte-Verlag, 5. Auflage, Köln 1990
- [2] R. Felix, C. C. Jaffe, H. U. Lemke, M. L. Rhodes: *CAR'91 Computer Assisted Radiology*, Springer-Verlag, Berlin 1991.
- [3] Lothar von Wolfersdorf, *Inverse und schlecht gestellte Probleme*, Akademie-Verlag Heft 5 Band 124, Berlin 1994
- [4] <http://www.siemens.de/med/d/gg/ct/basics/howworks.html>
- [5] <http://www.siemens.de/med/d/gg/ct/basics/usedfor.html>
- [6] <http://www.siemens.de/med/d/gg/ct/basics/ct-basic.html>
- [7] <http://www.siemens.de/med/d/gg/ct/news/new.html>
- [8] <http://nobel.sdsc.edu/laureates/medicine-1979.html>
- [9] <http://nobel.sdsc.edu/laureates/medicine-1979-press.html>

A Anhang

Dieses Kapitel zeigt den von uns erstellten Quellcode zur Berechnung des direkten Problems. Zusätzlich finden sich hier auch noch Zusatzprogramme, die die Erzeugung von Phantomen (Körper) erleichtern.

A.1 Berechnung der Schnittpunkte - »spunkt.m«

```
% Berechnung der Strahlenschnittpunkte mit dem Phantomgitter
%
% Parameter:  n    - Groesse des Phantomgitters
%             s_a  - Anstiege der Strahlen im Buendel
%             Q    - Quellpunkt

% Berechnung der y-Schnittpunkte von x = -n bis n
Y_S = diag(s_a)*(ones(length(alpha), 1)*[-n:n]-Q(1))+Q(2);

% Berechnung der x-Schnittpunkte von y = -n bis n
X_S = (ones(length(alpha), 1)*[-n:n]-Q(2))./(diag(s_a)*ones(length(alpha), 2*n+1))+Q(1);

% Ordnen der Schnittpunkte (x-aufsteigend)
P_X = [ X_S' ; diag([-n:n])*ones(size(X_S'))];

[S_X, i] = sort(P_X);
S_Y = (S_X-Q(1))*diag(s_a)+Q(2);

% Indizes der durchlaufenen Pixel
% Phantomkoordinaten berechnen
B_X = ceil(average(S_X)+n);
B_Y = ceil(average(S_Y)+n);

% Auffinden aussenliegender Pixel
i = find(B_X < 1 | B_X > 2*n);
j = find(B_Y < 1 | B_Y > 2*n);

% Index fuer Schwaechungsmatrix
B = [B_X+B_Y*2*n]-2*n;

% Aussortieren der aussenliegenden Pixel
B(i)=ones(length(i),1);
B(j)=ones(length(j),1);
```

```

% Laengenberechnung (Pythagoras)
len = sqrt(diff(S_X).^2+diff(S_Y).^2);

% Aussortieren der aussenliegenden Pixel
len(i)=zeros(length(i),1);
len(j)=zeros(length(j),1);

```

A.2 Erstellen einer eigenen Schwächungsmatrix - *>>make_object.m<<*

```

% Eingabeparameter abfragen
MSIZE = input('Groesse (n):');
f = input('Schwaechungsfaktor:t1');

% leere Matrix erzeugen (mit Nullen gef"ullt)
M = zeros(MSIZE,MSIZE);

% neue Figur erzeugen
figure;
hold off

% Gitternetz zeichnen und Achsen angleichen
[tx, ty] = meshgrid(0:MSIZE, 0:MSIZE);
plot(tx, ty, '-w'), hold on, plot( ty, tx, '-w')
axis([0 MSIZE 0 MSIZE])

% Mausposition abspeichern
[x, y, buttons] = ginput(1);

% Beim Klick wird das Feld unter der Maus markiert
while(buttons ~= 2)
    if buttons == 1
        fill( [floor(x) ceil(x) ceil(x) floor(x) floor(x)], ...
              [ceil(y) ceil(y) floor(y) floor(y) ceil(y)], 'w')
        M(ceil(x), ceil(y)) = f;
    elseif buttons == 3
        fill( [floor(x) ceil(x) ceil(x) floor(x) floor(x)], ...
              [ceil(y) ceil(y) floor(y) floor(y) ceil(y)], 'k')
        M(ceil(x), ceil(y)) = 0;
    end
    [x, y, buttons] = ginput(1);
end
close;

```



```

% neu erstellte Matrix wird in Datei gespeichert
MFNAME = input('Dateiname: ','s');
if MFNAME ~isempty(M)
    M=M';
    save(MFNAME, 'M');
end

```

A.3 Berechnung der Schwächung - »ct_dcalc.m«

```

function CT_dcalc

% Parameter festlegen
alpha=[-4:4]*pi/48;
n=10;

% Index fuer jedes Feld der Matrix erzeugen
SC = reshape(make_scross(n),2*n,2*n);
SM = reshape(SC',4*n*n,1);
phi=0;
r=2*n;
Q=r*[cos(phi);sin(phi)];
Iin = 100;
step = 45;

% Berechnen der Schnittpunkte
I = [];
for k=1:step
    phi=k*2*pi/step;
    s_a=tan(phi+alpha);
    Q=r*[cos(phi);sin(phi)];

% Zeichnen des Phantoms; besteht aus regelmaessigem Gitter
% mit Maschenweite 1 von -n bis +n
[x,y]=meshgrid([-n:1:n],[-n:1:n]);

% graphische Ausgabe des Phantoms
colormap(gray);
figure(FIG), hold off,
image(0.5-n,0.5-n,(SC'+0.5)*60);
set(gca, 'YDir', 'normal');
hold on,
plot(x,y,'-g'), hold on, plot(y,x,'-g')

```

```

% Zeichnen der Quelle und des Kreises, auf dem sich die Quelle bewegt
kreis_x=r*cos(2*pi*[0:200]/200);
kreis_y=r*sin(2*pi*[0:200]/200);
plot(kreis_x,kreis_y,'-r')
plot(Q(1),Q(2),'or')

% Zeichne Zentralstrahl
plot([Q(1) -Q(1)], [Q(2) -Q(2)], '-y')
P=Q*ones(1,length(alpha))-[cos(alpha+phi);sin(alpha+phi)] ...
*diag(cos(alpha))*2*r;
for j=1:length(alpha),
    plot([Q(1) P(1,j)], [Q(2) P(2,j)], '-c')
end
axis([-r r -r r])
axis('square')
axis('off')

% Funktionsaufruf
spunkt

% Austrittsintensitaeten
ST = reshape(SM(B), 41, 9);
Iout = exp(sum(-ST.*len))*Iin;
end

```

A.4 Erzeugung des Schweizer-Kreuz - *>make_scross.m*

```

function R = make_scross(n)

% Schwaechungskoeffizient
f = 0.7;

% Erzeugung einer quadratischen Matrix der Groesse n
M = zeros( n, n);

% Auffuellen der Matrix mit den Schwaechungskoeffizient
M(1:ceil(n/3),1:ceil(2*n/3))=ones(ceil(n/3), ceil(2*n/3))*f;
M(1:ceil(2*n/3),1:ceil(n/3))=ones(ceil(2*n/3), ceil(n/3))*f;

% Durch Drehung und Verbindung der einzelnen Matrizen wird Schweizer Kreuz erzeugt
M = [ rot90(rot90(M)) rot90(M);rot90(rot90(rot90(M))) M];

% Index fuer Berechnung umsetzen
R = reshape(M,1,4*n*n);

```