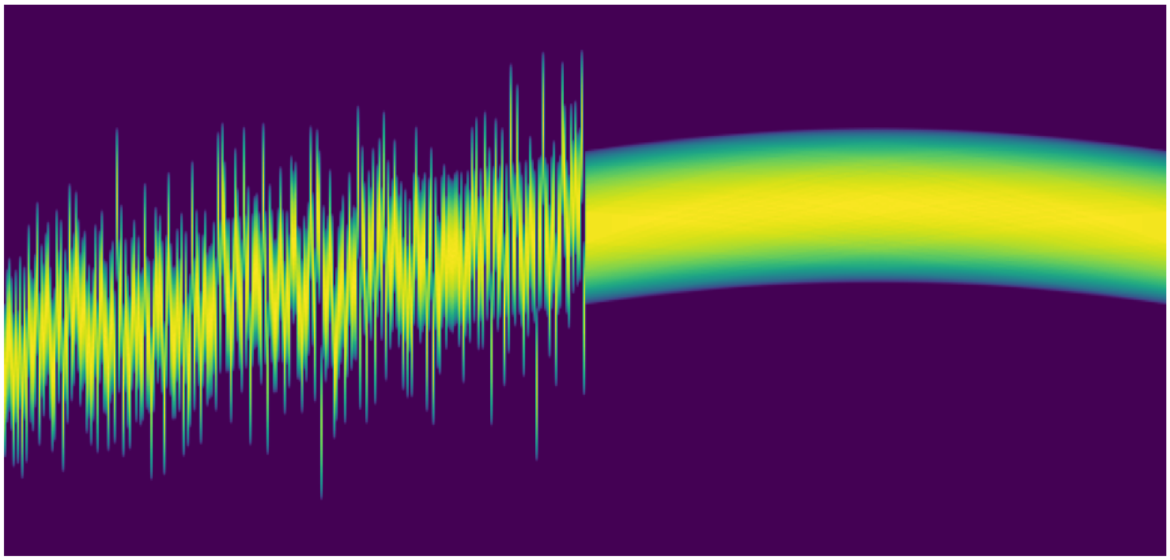
GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN IN PUBLICA COMMODA SEIT 1737

Sinogram Processing for Nano Computed Tomography with Bayesian Reconstruction



Bachelor Thesis in Mathematics and Mathematical Data Science at the University of Göttingen

by

Luca Raphael Lukačević

Primary Supervisor

Jun.-Prof. Dr. Anne Wald

Secondary Supervisor

Prof. Dr. Axel Munk

Submission

July 2023

Contents

1 Introduction

In this work we will adress a certain kinds of processing problem that turns up in Nano Computed Tomography.

Nano Computed Tomography (or Nano CT) is a special kinds of CT problem. The mathematics behind the tomography mostly stays the same, with one big additional concern that needs to be addressed. Objects on the nano scale will have a lot of intrinsic kinetic energy that leads to them moving a lot during the scanning process, up to 10 percent the size of their diameter. Movement at that scale can no longer be ignored in the reconstruction, as this random shifts induced during the scanning process will cause significant artefacts in the reconstructed images. In this work we will address the modelling process of this translation noise and how to process the CT data to mitigate the translational noise.

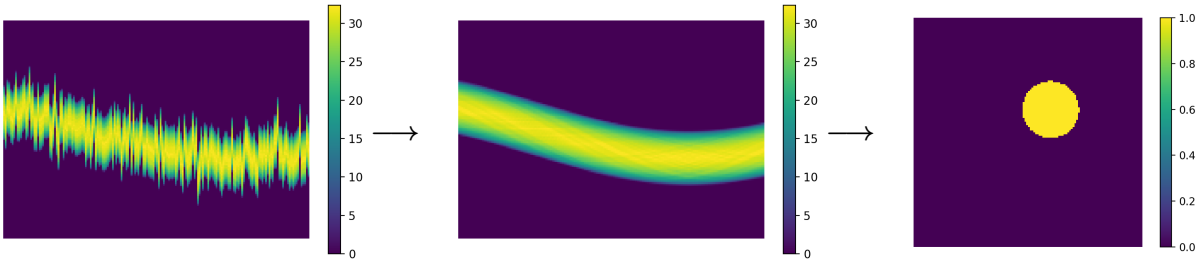


Figure 1: This is the overall goal for this thesis. Going from the left picture of a so called **sinogram** that contains translational noise to a processed sinogram with few to no noise to then a reconstructed object, in this instance a circle.

As Nano CT is first and foremost a CT problem, we will start this thesis out by carefully laying out the physical foundations of CT and put CT into the greater context of mathematical inverse problems. Inverse problems are a class of mathematical problems that try to calculate from a set of data and a generation rule for the data the factors that must have produced the data. As computed tomography is an inverse problem and there is a lot of theory on solving inverse problems, we will lay the foundations of them and of CT in Chapter 2 and give a basic approach to solving them: the Tikhonov regularization.

The Tikhonov regularization relies on a regularization parameter λ and we will choose the optimal parameter λ using Bayesian statistics. Chapter 3 will be spent on introducing relevant tools that are needed to formulate inverse problems using Bayesian statistics as so called Bayesian inverse problems and showing that the Tikhonov regularization can be seen as the solution to a Bayesian inverse problem with certain assumptions. By using this more general Bayesian approach we are then able to find an objective criterion for choosing parameter λ.

Chapter 4 will cover two important numerical algorithms that we will need to solve our processing and reconstruction problem: the CG method and the Levenberg-Marquardt algorithm.

In Chapter 5 we will combine all of the tools from Chapter 2 to Chapter 4. We will first describe our data, which we will generate artificially in Python. We then make use of the center to process our sinograms and finally reconstruct the objects using the Tikhonov regularization from Chapter 2 and the CG method from Chapter 4, selecting λ via Chapter 3.

Our method for the sinogram processing hinges on Proposition 3.3 of [ERMSS20] and on our method of discretizing the problem. This results in a parametric Bayesian inverse problem, a non-parametric approach can be found here [MNP19]. The parametric problem is already solved in [HF16], we only add more details to the derivation. The rest of the material is standard and used from textbooks, where we reference them at the respective moments.

2 Inverse Problems

Since we are dealing with a concrete inverse problem in Nano CT, we will first consider general inverse problems and how to handle them. We look at the mathematical modelling of CT more closely and finally consider the Tikhonov regularization as a solution method, which will come in handy later on.

2.1 Introducing Inverse Problems

For the introductory part we will be guided by the article [Kel76] of Joseph Keller. Two problems are called inverse to each other if the formulation of one depends in part or completely on the solution of the other. Of the two problems, typically one will be better understood than the other. This problem is called the **forward problem**. The forward problem is usually mathematically more well understood and as a consequence, a solution of the forward problem usually readily exists. The **inverse problem** then consists of finding the cause of the observed data from the forward problem. The inverse problem is usually less well understood than the forward problem. Mathematically, we concern ourselves with operator equations of the type

$$Tx = y, \tag{1}$$

where $T : X \to Y$ is an operator between linear spaces X and Y. The forward problem then consists of evaluating T at x, whereas for the inverse problem we are trying to invert the equation $Tx = y$.

Following french mathematician Jacques Hadamard we call the inverse problem $Tx = y$ well-posed if

- for all $y \in Y$ there exists an $x \in X$ with $Tx = y$;
- T is injective, i.e. $z \neq x$ implies $Tz \neq y$;
- for all $y \in Y$ this solution depends continuously on the data, i.e., $(x_n)_{n\in\mathbb{N}}$ with $Tx_n \to y$ implies $x_n \to x$.

If one of these conditions does not hold we call the equation ill-posed. One should note that for physical problems and noise free data we can expect the first condition to usually hold. The operator T will reflect some underlying physical relationship between effects and causes, hence we are really just requiring every effect to have a cause with respect to T. The three conditions are further not necessarily independent from each other. If we take X and Y as Banach spaces and $T \in \mathcal{L}(X, Y)$ and assume that the first two conditions hold, then the third condition will automatically hold by the bounded inverse theorem.

2.2 Computed Tomography as an Inverse Problem

So let us now consider the concrete problem of computed tomography. Computed tomography or CT is a technique that is well known for its use in medical imaging. The goal is to estimate the density function of some medium by sending X-rays through it and observing the attenuation (or gradual loss of flux intensity/magnitude). X-rays have a higher frequency than visible light and are able to penetrate most media that we are interested in. As part of the modeling process we then ignore refraction effects. Our base assumption for CT is that the rays travel on a straight line. In CT, there then is a known source emitting X-rays with a specific intensity. On the opposite side of the object, there is a detector that measures the intensity of the light for a second time. Our derivation will loosely follow [Nat86] for this part.

Physical Model We will first consider an X-ray passing through a medium. By then choosing coordinates accordingly, we can always achieve to describe the ray one dimensional. Now if we denote by $I : \mathbb{R} \to \mathbb{R}$

and $I(x)$ the light intensity of our beam at point x and consider the change in intensity for close points x_1 and x_2, then

$$I(x_2) - I(x_1) =: \Delta I \propto -I(x_1)\Delta x = -I(x_1)(x_2 - x_1).$$

The proportionality constant is denoted by μ and hence we get $\Delta I = -I(x_1)\mu(x_1)\Delta x$. Note that μ depends on the position and was often termed the "optical density" of a body. By rearranging terms we get

$$I(x_2) = I(x_1) - \mu(x_1)I(x_1)\Delta x = I(x_1)(1 - \mu(x_1)\Delta x).$$

The interpretation of this is that the intensity at point x_2 is the intensity at point x_1, times a factor between 0 and 1 that depends on the length of the difference and the density of the original point. For a length close to 0 or a density close to zero, we have a factor close to 1 and we will have the same light intensity. This makes sense, as if the light ray only moved a bit or we have moved through media with low density, the intensity should stay the same. However, for larger Δx or μ, the factor goes to zero and hence the intensity goes to zero. This is of course only an interpretation for close points, as we assume μ to be constant.

However, through this ansatz and letting $\Delta \to 0$, we then derive the differential equation

$$-\mathrm{d}I = \mu I \mathrm{d}x,$$

or

$$\frac{\mathrm{d}I}{\mathrm{d}x}(x) = I'(x) = -\mu(x)I(x).$$

Assuming there lies only some medium between x_1 and x_2 we can divide by $I(x)$ and integrate both sides to get

$$\ln(\frac{I(x_1)}{I(x_2)}) = \int_{x_1}^{x_2} \mu(x)\mathrm{d}x.$$

Since the medium lies only between x_1 and x_2, we have supp $\mu \subset [x_1, x_2]$ and we can extend the integration over the whole line to get:

$$\ln(\frac{I(x_1)}{I(x_2)}) = \int_{x_1}^{x_2} \mu(x)\mathrm{d}x = \int_{-\infty}^{\infty} \mu(x)\mathrm{d}x. \tag{2}$$

Radon Transform The important thing to note about (2) is that we can physically measure the left side of this equation. The mathematical objective is now to infer the density μ by only considering the integral over μ. This is of course not possible, as there can be many different real functions, that have the same integral value. However, assuming μ to be a density for all lines through the medium, we would then have $\mu : \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$ as a density function of the whole plane. If we were to then know the line integral value of μ for all possible lines through the medium, we could then reconstruct μ, as was already shown by Austrian mathematician Johann Radon in 1917 ([Rad17]).

Now, an arbitrary line L in 2d-space (or $\mathbb{R}^2$) can be parameterized by its unit normal vector $\boldsymbol{\theta} \in \mathbb{S}^1$ and its distance s to the origin. We get

$$L = L_{\boldsymbol{\theta},s} = \{s\boldsymbol{\theta} + t\boldsymbol{\theta}^{\perp} \mid t \in \mathbb{R}\},$$

where $\boldsymbol{\theta}^{\perp}$ is $\boldsymbol{\theta}$ rotated 90° anticlockwise.

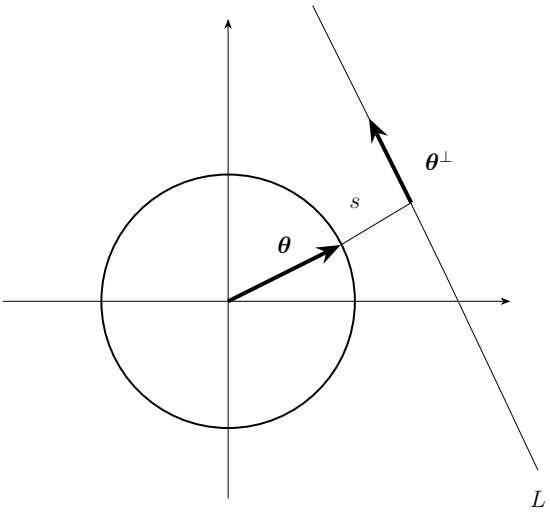


Figure 2: Visualization of the parameterized line L

By choosing $\theta \in [0, 2\pi]$ we can give a detailed description of $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^\perp$:

$$\boldsymbol{\theta} = \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix}$$

and

$$\boldsymbol{\theta}^\perp = \begin{pmatrix} -\sin(\theta) \\ \cos(\theta) \end{pmatrix}.$$

This now enables us to define the Radon transform.

Definition 2.1 Suppose $\mu : \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$ is integrable. We then define the Radon transform $\mathcal{R}$ of μ as

$$\mathcal{R}\mu(L) = \mathcal{R}\mu(\boldsymbol{\theta}, s) := \int_{-\infty}^{\infty} \mu(s\boldsymbol{\theta} + t\boldsymbol{\theta}^\perp)\mathrm{d}t.$$

Remark One can show that the Radon transform is a well defined linear operator

$$\mathcal{R} : L^1(\mathbb{R}^2) \to L^1([0, 2\pi] \times \mathbb{R}).$$

For more details we again refer to [Nat86], Chapter I and II.

Computed Tomography as an Inverse Problem Now with (2) in mind we want to formulate CT as an inverse problem of the type $Tx = y$. For an object that has a given density function μ, we can think about measuring the attenuation of shooting light beams from all possible angles through the object. Resulting would be a function

$$g = \mathcal{R}\mu,$$

where we are able to measure g by equation (2). Hence we formulate the problem as a problem of type (1).

Discretization of the Problem In practice we are unable to measure g for all possible lines L through the object. Instead, we are content with making a finite choice of rays to shoot through the object. This choice is determined by the resolution that we want to achieve and other more theoretical sampling considerations, but also practical ones like the acceptable time duration of a scan. Based on these considerations, the scanners are designed with a certain scanner geometry. The most popular choices are the fan beam and parallel beam geometry. We will concentrate only on the parallel beam geometry

though. Here we choose equally spaced angles $\theta_1, ..., \theta_t$ and distances $s_1, ..., s_r$.

If we assume that supp $\mu \subset [-1,1]^2 =: D$ we can cover D with many small squares (or pixels) $p_1, ..., p_m$ and approximate μ at each pixel by a constant to approximate μ in the end by a piecewise constant function. So we take f as the vectorized image with components $f_j \approx \mu|_{p_j}$, where $f = (f_j)_{j=1}^m$.

If we assume made measurements $g = (g_i)_{i=1}^k$, meaning

$$\int_{-\infty}^{\infty} \mu(s_{L_i}\boldsymbol{\theta}_{L_i} + t\boldsymbol{\theta}_{L_i}^{\perp})\mathrm{d}t =: \int_{L_i} \mu = g_i, \tag{3}$$

we can then find an approximation of the integrals.

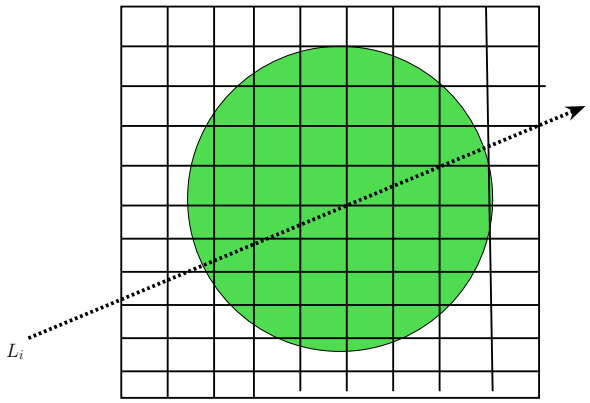


Figure 3: L_i intersects with the object

Let $a_{i,j} := \text{length}(L_i \cap p_j)$ be the length of the line L_i inside p_j. Let $a_i := (a_{i,1}, \ldots, a_{i,m})^T \in \mathbb{R}^m$, $i = 1, \ldots, k$. The discretized version of equation (3) is now of the form

$$f^T a_i = a_i^T f = g_i, \quad i = 1, \ldots, k.$$

Taking all equations $a_i^T f = g_i$ at once, we obtain $Af = g$, where $A \in \mathbb{R}^{k \times m}$ is determined by the rows a_i^T, and $g = (g_1, \ldots, g_k)^T$.

2.3 Solutions to Inverse Problems and the Tikhonov Regularization

Guided by [Wer92] Chapter 1, let us now consider how we can solve a linear equation of the form

$$Af = g$$

for $A \in \mathbb{R}^{k \times m}$, $f \in \mathbb{R}^m$ and $g \in \mathbb{R}^k$. We cannot rely on A to be injective nor surjective, hence we will instead minimize the least-squares error:

$$\begin{aligned} \min_f \quad & \|Af - g\|_2^2 \\ \text{s.t.} \quad & f \in \mathbb{R}^m. \end{aligned} \tag{4}$$

However, if the least squares problem (4) is unstable (i.e. if $A = QR$ is a QR decomposition of A, R would have eigenvalues close to zero), we settle on minimizing

$$\begin{aligned} \min_f \quad & \|Af - g\|_2^2 + \lambda \|f\|_2^2 \\ \text{s.t.} \quad & f \in \mathbb{R}^m, \end{aligned} \tag{5}$$

for $\lambda > 0$ a positive constant. (5) is also known as Tikhonov regularization. Problems (4) and (5)

can be minimized respectively by solving an equivalent normal equation. For this, we need the following auxiliary lemma.

Lemma 2.2 *For a matrix $A \in \mathbb{R}^{k\times m}$ it holds that*

1. *A^TA is positive semidefinite.*
2. *A^TA is positive definite iff* $\ker A = \{0\}$.
3. $\ker A^TA = \ker A$.
4. $\operatorname{Im} A^T = (\ker A)^\perp$.
5. $\operatorname{Im} A^TA = \operatorname{Im} A^T$

PROOF 1. For all $x \in \mathbb{R}^m$ we have $x^T(A^TA)x = (xA)^TAx = \|Ax\|_2^2 \geq 0$.

2. This follows immediately from the fact that A^TA is positive semidefinite.

3. $Ax = 0$ implies $A^TAx = A^T0 = 0$ and hence $\ker A^TA \subset \ker A$. If $A^TAx = 0$, then $x^TA^TAx = 0$ and hence $\|Ax\|_2^2 = 0$, which implies $Ax = 0$. This proves $\ker A^TA \supset \ker A$ and hence equality.

4. Let $y = A^Tx \in \operatorname{Im} A^T$. Then, for all $z \in \ker A$, we have the relation $z^Ty = z^TA^Tx = (Az)^Tx = 0$, which implies $y \in (\ker A)^\perp$. Therefore, we have $\operatorname{Im} A^T \subseteq (\ker A)^\perp$. Now, equality follows from the following dimension argument:

$$\dim((\ker A)^\perp) = m - \dim(\ker A) = \dim(\operatorname{Im} A) = \dim(\operatorname{Im} A^T),$$

where the last equality comes from the fact that row rank equals column rank for any matrix.

5. Using the previous parts of the lemma we get $\operatorname{Im} A^TA = (\ker A^TA)^\perp = (\ker A)^\perp = \operatorname{Im} A^T$. □

Theorem 2.3 *A vector $f \in \mathbb{R}^m$ will minimize the least squares problem (4) if and only if it is the solution to*

$$A^TAf = A^Tg. \tag{6}$$

A solution to (6) always exists.

PROOF Let us first consider the proof for the existence of a solution. This follows at once from the previous lemma as $A^Tg \in \operatorname{Im} A^T$ but then by 4. we have that $A^Tg \in \operatorname{Im} A^TA$, hence a solution to the equation exists. For the first part of the theorem we will define $\psi(x) := \|Ax - g\|_2^2$. If f is now a solution to (6), we get for all $x \in \mathbb{R}^m$ that

$$\begin{aligned}
\psi(x) - \psi(f) &= \|Ax - g\|_2^2 - \|Af - g\|_2^2 \\
&= x^TA^TAx - f^TA^TAf - 2x^TA^Tg + 2f^TA^Tg \\
&= x^TA^TAx - f^TA^TAf - 2x^TA^TAf + 2f^TA^TAf \\
&= (x-f)^TA^TA(x-f) \\
&= \|A(x-f)\|_2^2 \\
&\geq 0.
\end{aligned}$$

Thus we can see that f minimizes ψ, and the minimum is attained for all x with $x - f \in \ker A = \ker A^TA$. Now we further know that since f solves (6) the solution space will have the form $f + \ker A^TA$ and hence $x = f + (x - f)$ also solves (6). This concludes the proof. □

Making use of the normal equation (6), we can now easily prove a similar equation for (5).

Corollary 2.4 *The vector $f \in \mathbb{R}^m$ will minimize the least squares problem (4) if and only if it is the solution to*

$$(A^T A + \lambda I_m) f = A^T g, \tag{7}$$

where I_m is the $m \times m$ identity matrix. A solution to (7) always exists.

PROOF Proving this corollary can also be achieved similarly to Theorem 2.3, but we can instead use the following idea. Minimizing the Tikhonov regularization functional is equivalent to minimizing

$$\left\| \begin{pmatrix} A \\ \sqrt{\lambda} I_m \end{pmatrix} f - \begin{pmatrix} g \\ 0 \end{pmatrix} \right\|^2_{\mathbb{R}^k \times \mathbb{R}^m} = \left\| \begin{pmatrix} Af - g \\ \sqrt{\lambda} f \end{pmatrix} \right\|^2_{\mathbb{R}^k \times \mathbb{R}^m},$$

but now the left side is in the form of the usual least squares functional (4). Making use of the previous theorem then yields the normal equation

$$\begin{pmatrix} A^T & \sqrt{\lambda} I_m \end{pmatrix} \begin{pmatrix} A \\ \sqrt{\lambda} I_m \end{pmatrix} f = \begin{pmatrix} A^T & \sqrt{\lambda} I_m \end{pmatrix} \begin{pmatrix} g \\ 0 \end{pmatrix}.$$

Now matrix multiplication shows that this is equivalent to

$$(A^T A + \lambda I_m) f = A^T g. \qquad \square$$

We will see in later chapters how to solve (7) efficiently using conjugate gradients, but we will already note some of the regularity properties of the matrix $(A^T A + \lambda I_m)$.

Proposition 2.5 *For all $\lambda > 0$, the matrix $(A^T A + \lambda I_m)$ is positive definite and invertible.*

PROOF The sum of a positive definite matrix and a positive semidefinite matrix is again positive definite. Now λI_m is positive definite for all $\lambda > 0$, and $A^T A$ is positive semidefinite by Lemma 2.2. Hence $(A^T A + \lambda I_m)$ is positive definite. Since positive definite matrices only have positive eigenvalues, $(A^T A + \lambda I_m)$ is also invertible. $\square$

3 Bayesian Inversion

In this section we will introduce Bayesian inverse problems, showing among other things how the Tikhonov regularization can be derived that way. The advantage with this approach is that it will also allow us to choose an appropriate regularization parameter $\lambda > 0$ and deal with noisy measurements. Specifically, we will now focus on solving the equation

$$y = Ax + \nu, \tag{8}$$

where $y \in \mathbb{R}^n$ is an observation vector, $A \in \mathbb{R}^{n\times m}$ is the model, $x \in \mathbb{R}^m$ is the vector of unknowns, and $\nu \in \mathbb{R}^n$ is noise. In our concrete problem of computed tomography the noise term n will comprise additive Gaussian noise. We also assume that all the variables are realizations of random vectors, hence $Y = y$, $X = x$, and $N = \nu$ for random vectors N, X, and Y. We can already see that Y is fully determined by X and N (since A is just our usual Radon matrix), hence we want to use the language of Bayesian statistics to find a way to express this dependence in order to solve the equation. For this, we will first introduce the needed Bayesian framework and needed tools, and then go on to formulate our problem precisely in the developed framework, ending off with deriving the Tikhonov regularization problem with the right choice for λ.

3.1 Conditional Expectation and Distribution

To introduce the Bayesian framework that is needed to reformulate our inverse problem, we first need to develop the theory of conditional distributions. Conditional distributions allow us to specify the dependence structure between random vectors and then infer the distribution and other information about the random vectors.

We begin with a prelude that establishes some foundational concepts.

Definition 3.1 Suppose that $(\Omega, \mathcal{A}, \mathbb{P})$ is a measure space (meaning that Ω is a set, $\mathcal{A}$ is a σ-algebra and $\mathbb{P}$ is a probability measure) and $(B, \|\cdot\|_B)$ is a Banach space. We say that a function $h : \Omega \to B$ is in $L^p(\Omega, B)$, if the function $\omega \mapsto \|h(\omega)\|_B$ is in $L^p(\Omega, \mathbb{R})$. We then define

$$\|h\|_p := \left(\int_\Omega \|h(\omega)\|_B^p \mathrm{d}\mathbb{P}(\omega)\right)^{1/p}.$$

Remark In the case that $B = \mathbb{R}^m$, $h \in L^p(\Omega, B)$ is equivalent to all component functions themselves being in $L^p(\Omega, \mathbb{R})$. Since all norms are equivalent on $\mathbb{R}^m$, they will all produce equivalent norms on $L^p(\Omega, \mathbb{R}^m)$. We will then choose the p-norm, such that

$$\|h\|_p = \left(\int_\Omega \sum_{i=1}^m |h_i(\omega)|^p \mathrm{d}\mathbb{P}(\omega)\right)^{1/p} = \left(\sum_{i=1}^m \int_\Omega |h_i(\omega)|^p \mathrm{d}\mathbb{P}(\omega)\right)^{1/p}.$$

To now motivate the definition of the conditional expectation, let us consider an experiment. First, we want to uniformly draw a random number P from the unit interval, $P \sim \mathcal{U}(0, 1)$ or P is uniformly distributed and based on the outcome $P = p$, we then flip a coin with probability of success p. Let X be the outcome of the coin. Then we would intuitively like to require that

$$\mathbb{P}(X = 1|P = p) = p.$$

However, this probability is ill-defined, as $\mathbb{P}(P = p) = 0$ for all $p \in (0, 1)$. Hence a generalization of the elemental conditional probability is needed.

For the rest of this section we will be guided by [Dur19].

For two measures μ and ν we introduce the notation $\nu << \mu$ if $\mu(A) = 0 \implies \nu(A) = 0$ for all A and say that ν is absolutely continuous with respect to μ. The **Radon-Nikodym Theorem** then famously states that $\nu << \mu$ iff ν has a density with respect to μ, or there exists a measurable function f such that

$$\nu(A) = \int_A f \, \mathrm{d}\mu \quad \text{for all } A.$$

The function f is often denoted by $d\nu/d\mu$ and called the **Radon-Nikodym derivative**. The measure ν is commonly denoted by $f \cdot \mu$ or $f \cdot d\mu = \frac{d\nu}{d\mu} \cdot d\mu$.

We will now start off by defining conditional expectations.

Definition 3.2 Suppose $(\Omega, \mathcal{A}, \mathbb{P})$ is a probability space and $X : (\Omega, \mathcal{A}, \mathbb{P}) \to \mathbb{R}^m$ a random vector. For a sub-σ-algebra $\mathcal{F} \subset \mathcal{A}$ we then call a random vector $Z : \Omega \to \mathbb{R}^m$ the conditional expectation of X with respect to $\mathcal{F}$, if

1. $Z \in L^1(\Omega, \mathbb{R}^m)$,
2. Z is $\mathcal{F}$-measurable,
3. for all $F \in \mathcal{F}$ it holds that
$$\int_F X \mathrm{d}\mathbb{P} = \int_F Z \mathrm{d}\mathbb{P}. \tag{9}$$

Remark We then denote Z by $\mathbb{E}(X|\mathcal{F})$.

Theorem 3.3 *Suppose $X \in L^1(\Omega, \mathbb{R}^m)$. Then $\mathbb{E}(X|\mathcal{F})$ exists and is a.s. unique.*

Proof We first show existence component-wise. First suppose that $X_i \geq 0$. We then define $\mathbb{Q}_i(F) := \int_F X_i \mathrm{d}\mathbb{P}$ as a new measure on $\mathcal{F}$. Since $\mathbb{Q}_i << \mathbb{P}|_{\mathcal{F}}$, we have that there exists the Radon-Nikodym derivative $Z_i : \Omega \to \mathbb{R}$, that is $\mathcal{F}$-measurable. By definition, we then have for all $F \in \mathcal{F}$ that

$$\int_F X_i \mathrm{d}\mathbb{P} = \mathbb{Q}_i(F) = \int_F Z_i \mathrm{d}\mathbb{P}.$$

Hence the Z_i fulfill condition (9) in Definition 3.2. For general X_i we now use the standard decomposition $X_i = X_i^+ - X_i^-$ (where we remind the reader that $X_i^+ := \max\{X, 0\}$ and $X_i^- := \max\{-X, 0\}$ as pointwise maxima), apply what we have previously shown and put everything together for all components at once. The integrability follows by $|X_i| = X_i^+ + X_i^-$, which are both integrable by assumption. That shows existence of $\mathbb{E}(X|\mathcal{F})$.

For uniqueness we will without loss of generality assume that uniqueness fails in the first component with $\hat{Z}$ and $\tilde{Z}$ both fulfilling the conditions in Definition 3.2. We then consider $F_1 = \{\hat{Z}_1 > \tilde{Z}_1\}$ and $F_2 = \{\hat{Z}_1 < \tilde{Z}\}$. Since $\hat{Z}_1$ and $\tilde{Z}_1$ are not a.s. equal to each other it must hold that $\mathbb{P}(F_1) > 0$ or $\mathbb{P}(F_2) > 0$. Without loss of generality assume $\mathbb{P}(F_1) > 0$. We then have

$$\int_{F_1} \underbrace{\hat{Z}_1 - \tilde{Z}_1}_{>0} \mathrm{d}\mathbb{P} > 0,$$

which is equivalent to

$$\int_{F_1} \hat{Z}_1 \mathrm{d}\mathbb{P} > \int_{F_1} \tilde{Z}_1 \mathrm{d}\mathbb{P}.$$

This is a contradiction. Hence we have shown a.s. uniqueness. □

We will now collect some basic properties of conditional expectations that will later be useful when it comes to calculations.

Theorem 3.4 *Suppose $X, (X_i)_{i\in\mathbb{N}}$, and Y are random vectors in $L^1(\Omega, \mathbb{R}^m)$ and in $L^1(\Omega, \mathbb{R}^n)$ respectively, $\alpha, \beta \in \mathbb{R}$ and $\mathcal{F}, \mathcal{H} \subset \mathcal{A}$ sub-σ-algebras. Then the following things hold a.s. or in L^1.*

(a) If $m = n$, then $\mathbb{E}(\alpha X + \beta Y|\mathcal{F}) = \alpha\mathbb{E}(X|\mathcal{F}) + \beta\mathbb{E}(Y|\mathcal{F})$.

(b) X and $\mathcal{F}$ independent implies $\mathbb{E}(X|\mathcal{F}) = \mathbb{E}(X)$.

(c) If X is $\mathcal{F}$-measurable, then $\mathbb{E}(X|\mathcal{F}) = X$.

(d) $\mathbb{E}(\mathbb{E}(X|\mathcal{F})) = \mathbb{E}(X)$ (the law of iterated expectation).

(e) The tower property: $\mathcal{H} \subset \mathcal{F}$ implies $\mathbb{E}(\mathbb{E}(X|\mathcal{F})|\mathcal{H}) = \mathbb{E}(X|\mathcal{H})$.

(f) Independence: if $m = 1$ and X is independent from $\sigma(Y, \mathcal{F})$, then $\mathbb{E}(XY|\mathcal{F}) = \mathbb{E}(X)\mathbb{E}(Y|\mathcal{F})$.

(g) The pull-out property: if $m = 1$ and X is $\mathcal{F}$-measurable, then $\mathbb{E}(XY|\mathcal{F}) = X\mathbb{E}(Y|\mathcal{F})$.

The following properties only hold for $m = n = 1$.

(h) Monotonicity: $X \leq Y$ implies $\mathbb{E}(X|\mathcal{F}) \leq \mathbb{E}(Y|\mathcal{F})$.

(i) Monotone convergence: if $0 \leq X_i \uparrow X$ a.s., then $\mathbb{E}(X_i|\mathcal{F}) \uparrow \mathbb{E}(X|\mathcal{F})$ almost surely.

(j) Conditional Fatou: if $0 \leq X_i$ a.s. and $\liminf_{i\to\infty} X_i = X$ almost surely, then $\mathbb{E}(X|\mathcal{F}) \leq \liminf_{i\to\infty} \mathbb{E}(X_i|\mathcal{F})$ almost surely.

(k) Dominated convergence: if $X_i \to X$ a.s. and $|X_i| \leq Y$ with $Y \in L^1$, then $\mathbb{E}(X_i|\mathcal{F}) \to \mathbb{E}(X|\mathcal{F})$ almost surely.

PROOF (a) This follows easily by using the definition.

(b) Integrability and measurability are clear, remains to show the third property. For $F \in \mathcal{F}$ we have that

$$\mathbb{E}(\mathbb{1}_F X) = \mathbb{E}(\mathbb{1}_F)\mathbb{E}(X) = \mathbb{E}(\mathbb{1}_F \mathbb{E}(X)).$$

(c) This again follows straight from the definition.

(d) This follows from the third property of the conditional expectation with $F = \Omega$.

(e) Integrability is by definition. The $\mathcal{H}$-measurability of $\mathbb{E}(X|\mathcal{H})$ is also clear. For $H \in \mathcal{H}$ we have

$$\int_H \mathbb{E}(X|\mathcal{F})\mathrm{d}\mathbb{P} = \int_H X\mathrm{d}\mathbb{P} = \int_H \mathbb{E}(X|\mathcal{H})\mathrm{d}\mathbb{P},$$

where we used $H \in \mathcal{F}$. That shows the statement.

(f) Integrability and measurability are easy to verify. For $F \in \mathcal{F}$ we get

$$\begin{aligned}\int_F \mathbb{E}(X)\mathbb{E}(Y|\mathcal{F})\mathrm{d}\mathbb{P} &= \mathbb{E}(X)\int_F \mathbb{E}(Y|\mathcal{F})\mathrm{d}\mathbb{P} \\ &= \mathbb{E}(X)\int_F Y\mathrm{d}\mathbb{P} \\ &= \mathbb{E}(X)\int \mathbb{1}_F Y\mathrm{d}\mathbb{P} \\ &= \int_F XY\mathrm{d}\mathbb{P},\end{aligned}$$

where we used the independence in the last step.

(g) We only show the third property as the other two are straightforward. First take $X, Y \geq 0$. We will use the function sequence $X_k := \frac{1}{2^k}\lfloor 2^k X \rfloor \uparrow X$ a.s. and hence also $X_k \mathbb{E}(Y|\mathcal{F}) \uparrow X\mathbb{E}(Y|\mathcal{F})$ almost surely. Now repeatedly applying the normal Monotone Convergence Theorem (MCT) gives

$$\begin{aligned}
\int_F X\mathbb{E}(Y|\mathcal{F})\mathrm{d}\mathbb{P} &= \lim_{k\to\infty}\int_F X_k\mathbb{E}(Y|\mathcal{F})\mathrm{d}\mathbb{P} &= \lim_{k\to\infty}\int_F \sum_{j=0}^{\infty}\frac{j}{2^k}\mathbb{1}_{\{X_k=\frac{j}{2^k}\}}\mathbb{E}(Y|\mathcal{F})\mathrm{d}\mathbb{P} \\
&&= \lim_{k\to\infty}\sum_{j=0}^{\infty}\int_F \frac{j}{2^k}\mathbb{1}_{\{X_k=\frac{j}{2^k}\}}Y\mathrm{d}\mathbb{P} \\
&&= \lim_{k\to\infty}\int_F X_k Y\mathrm{d}\mathbb{P} \\
&&= \int_F XY\mathrm{d}\mathbb{P}
\end{aligned}$$

for all $F \in \mathcal{F}$. For arbitrary X and Y we now use the decompositions $X = X^+ - X^-$ and $Y = Y^+ - Y^-$.

(h) $F := \{\mathbb{E}(X|\mathcal{F}) > \mathbb{E}(Y|\mathcal{F})\} \in \mathcal{F}$ by the $\mathcal{F}$-measurability of the conditional expectation. We thus get

$$0 \geq \int_F \mathbb{E}(Y|\mathcal{F}) - \mathbb{E}(X|\mathcal{F})\mathrm{d}\mathbb{P} = \int_F \mathbb{E}(Y-X|\mathcal{F})\mathrm{d}\mathbb{P} = \int_F (Y-X)\mathrm{d}\mathbb{P} \geq 0$$

by assumption. Hence $\mathbb{1}_F(\mathbb{E}(Y|\mathcal{F}) - \mathbb{E}(X|\mathcal{F})) = 0$ a.s. and thus $\mathbb{1}_F = 0$ a.s. or equivalently $\mathbb{E}(X|\mathcal{F}) \leq \mathbb{E}(Y|\mathcal{F})$ almost surely.

(i) Using the monotonicity yields that $0 \leq \mathbb{E}(X_i|\mathcal{F}) \leq \mathbb{E}(X_{i+1}|\mathcal{F}) \leq \ldots \leq \mathbb{E}(X|\mathcal{F})$ so there is an almost sure limit $V := \lim_{i\to\infty}\mathbb{E}(X_i|\mathcal{F})$ and $V \leq \mathbb{E}(X|\mathcal{F})$. Define $B = \{V < \mathbb{E}(X|\mathcal{F})\} \in \mathcal{F}$, then

$$\begin{aligned}
\int_B \mathbb{E}(X|\mathcal{F})\mathrm{d}\mathbb{P} &= \int_B X\mathrm{d}\mathbb{P} \\
&= \lim_{i\to\infty}\int_B X_i\mathrm{d}\mathbb{P} \\
&= \lim_{i\to\infty}\int_\Omega \mathbb{E}(\mathbb{1}_B X_i|\mathcal{F})\mathrm{d}\mathbb{P} \\
&= \lim_{i\to\infty}\int_B \mathbb{E}(X_i|\mathcal{F})\mathrm{d}\mathbb{P} \\
&= \int_B V\mathrm{d}\mathbb{P}
\end{aligned}$$

by the normal MCT. But then $\mathbb{P}(B)$ must be 0. Hence $V = \mathbb{E}(X|\mathcal{F})$.

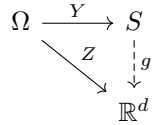
The conditional versions of Fatou's Lemma and the DCT can be now proven by use of the MCT for conditional expectations analogously to the usual case. We will forgo these proofs. □

We are particularly interested in the case, where we condition on another random vector (like in our motivating example).

Definition 3.5 For $\mathcal{F} = \sigma(Y)$, we call $\mathbb{E}(X|\mathcal{F})$ the conditional expectation of X with respect to Y and denote it by $\mathbb{E}(X|Y)$.

In that special case the conditional expectation contains more meaning as we will show by making use of the following lemma.

Lemma 3.6 *Suppose $Y : (\Omega, \mathcal{A}) \to (S, \mathcal{S})$ is a measurable mapping and $Z : \Omega \to \mathbb{R}^d$ is $\sigma(Y)$-measurable. Then there exists an $\mathcal{S}$-measurable mapping $g : S \to \mathbb{R}^d$ such that $Z = g \circ Y$.*



PROOF For the proof we will without loss of generality assume $d = 1$. Otherwise we would prove equality component-wise, which suffices for the statement. The proof will be done in a couple of steps.

1. With the decomposition $Z = Z^+ - Z^-$, assume $Z \geq 0$.

2. If $Z \geq 0$ is $\sigma(Y)$-measurable, then there exists a sequence $(Z_i)_{i\in\mathbb{N}}$ of simple functions with $Z_i \uparrow Z$. Assuming the statement already holds for simple functions we then find a sequence of g_i such that $Z_i = g_i \circ Y$. Then we have

$$Z = \lim_{i\to\infty} Z_i = \lim_{i\to\infty} (g_i \circ Y) = \limsup_{i\to\infty} (g_i \circ Y) = \underbrace{(\limsup_{i\to\infty} g_i)}_{=:g} \circ Y = g \circ Y.$$

3. It remains to show the statement for simple functions of the form $Z = \sum_{j=1}^d a_j \mathbb{1}_{A_j}$. Since Z is $\sigma(Y)$-measurable, all the A_j must lie in $\sigma(Y)$ and hence we have $A_j = Y^{-1}(B_j)$ for some B_j. But then

$$Z = \sum_{j=1}^d a_j \mathbb{1}_{A_j} = \sum_{j=1}^d a_j \mathbb{1}_{Y^{-1}(B_j)} = \underbrace{\sum_{j=1}^d a_j \mathbb{1}_{B_j}}_{=:g} \circ Y = g \circ Y.$$

That concludes the proof. □

The following statement now shows that in the special case of $\mathcal{F} = \sigma(Y)$, we can think of $\mathbb{E}(X|Y)$ as a function of Y, that locally approximates X through its defining properties.

Theorem 3.7 *(and Definition of the conditional expectation of X given $Y = y$) Let $X \in L^p(\Omega, \mathbb{R}^m)$ and $Y \in L^p(\Omega, \mathbb{R}^n)$ be random vectors. Then*

$$\mathbb{E}(X|Y) = g \circ Y \tag{10}$$

for a measurable function $g : \mathbb{R}^n \to \mathbb{R}^m$. The function g is $\mathbb{P}^Y$ a.s. unique and we write $g(y) = \mathbb{E}(X|Y = y)$.

PROOF By assumption $\mathbb{E}(X|Y)$ is Y-measurable and hence we get $\mathbb{E}(X|Y) = g\circ Y$ by the previous lemma. If we have $\hat{g}$ and $\tilde{g}$ that fulfill (10), then $\mathbb{P}^Y(\hat{g} \neq \tilde{g}) = \mathbb{P}(\hat{g} \circ Y \neq \tilde{g} \circ Y) = \mathbb{P}(\mathbb{E}(X|Y) \neq \mathbb{E}(X|Y)) = 0$. That proves the uniqueness. □

We will see that $\mathbb{E}(X|Y = y)$ can be obtained by integration against a conditional measure which we will now define.

Definition 3.8 Suppose $(\Omega, \mathcal{A}, \mathbb{P})$ is a probability space and $X : (\Omega, \mathcal{A}, \mathbb{P}) \to \mathbb{R}^m$ and $Y : (\Omega, \mathcal{A}, \mathbb{P}) \to \mathbb{R}^n$ are random vectors. A mapping from $\mathbb{R}^n$ and the Borel sets $\mathcal{B}(\mathbb{R}^m)$,

$$\mu : \mathbb{R}^n \times \mathcal{B}(\mathbb{R}^m) \to [0, 1]$$

such that

1. $\mu(y, \cdot)$ is a probability measure for all $y \in \mathbb{R}^n$,

2. for all $A \in \mathcal{B}(\mathbb{R}^m)$ we have $\mathbb{P}^Y$ a.s. equivalence of the mappings $y \mapsto \mathbb{E}(\mathbb{1}_{\{X\in A\}}|Y = y)$ and $y \mapsto \mu(y, A)$,

is called a regular conditional distribution of X with respect to $Y = y$.

Remark We will denote $\mathbb{E}(\mathbb{1}_{\{X\in A\}}|Y = y)$ by $\mathbb{P}(X \in A|Y = y) = \mathbb{P}^{X|Y=y}(A)$ and further $\mu(y, A) = \mathbb{P}^{X|Y=y}(A)$ or $\mu(y, \cdot) = \mathbb{P}^{X|Y=y}$.

Regular conditional distributions exist under mild conditions that hold for our purposes.

Definition 3.9 A measure space $(S, \mathcal{S})$ is said to be nice, if there exists a bijection $\psi : S \to \mathbb{R}$ such that ψ and ψ^{-1} are measurable.

Theorem 3.10 *If S is a Borel subset of a complete separable metric space M and $\mathcal{S}$ is the collection of Borel subsets of S, then $(S, \mathcal{S})$ is nice.*

PROOF For the proof we refer to Chapter 2 of [Dur19], Theorem 2.1.22. □

Theorem 3.11 *Suppose X and Y take values in nice spaces. Then a regular conditional distribution of X with respect to $Y = y$ exists.*

PROOF For the proof we again refer to [Dur19], Theorem 4.1.17 and 4.1.18. □

The following lemma will be needed to show several versions of the Disintegration Theorem.

Lemma 3.12 *The map $g : \mathbb{R}^n \to \mathbb{R}^m$, $g(y) = \mathbb{E}(X|Y = y)$ is the $\mathbb{P}^Y$-almost surely unique measurable map that satisfies*

$$\int_B g(y) d\mathbb{P}^Y(y) = \mathbb{E}(X \mathbb{1}_{\{Y \in B\}})$$

for every $B \in \mathcal{B}(\mathbb{R}^n)$.

PROOF We will use the transformation formula to get the following equivalences:

$$\begin{aligned}
&\int_B g(y) \mathrm{d}\mathbb{P}^Y(y) = \mathbb{E}(X \mathbb{1}_{\{Y \in B\}}) \text{ for all } B \in \mathcal{B}(\mathbb{R}^n) \\
\iff &\int \mathbb{1}_B \circ Y \cdot g \circ Y \mathrm{d}\mathbb{P} = \mathbb{E}(X \mathbb{1}_{\{Y \in B\}}) \text{ for all } B \in \mathcal{B}(\mathbb{R}^n) \\
\iff &\int \mathbb{1}_D \cdot g \circ Y \mathrm{d}\mathbb{P} = \mathbb{E}(X \mathbb{1}_D) \text{ for all } D \in \sigma(Y) \\
\iff &g \circ Y = \mathbb{E}(X|Y).
\end{aligned}$$

□

Now we can prove the Disintegration Theorem. It will give us a powerful tool to calculate expectations without knowing the complete distribution $\mathbb{P}^{(X,Y)}$, but instead only the distribution of Y and the conditional distribution of X wrt. to $Y = y$.

Theorem 3.13 *(Disintegration I) Assume that X and Y are random vectors such that for all $y \in \mathbb{R}^n$ the conditional distribution $\mathbb{P}^{X|Y=y}$ exists. Let $\Phi : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$ be a measurable map such that $\mathbb{E}(|\Phi(X, Y)|) < \infty$. Then, the map*

$$y \longmapsto \int_{\mathbb{R}^m} \Phi(x, y) d\mathbb{P}^{X|Y=y}(x)$$

is $\mathbb{P}^Y$-integrable and

$$\int_{\mathbb{R}^m \times \mathbb{R}^n} \Phi(x, y) d\mathbb{P}^{(X,Y)}(x, y) = \int_{\mathbb{R}^n} \left(\int_{\mathbb{R}^m} \Phi(x, y) d\mathbb{P}^{X|Y=y}(x) \right) d\mathbb{P}^Y(y). \tag{11}$$

PROOF The proof consists of a couple of steps. We will first show this to hold when Φ is an indicator function and then lift the argument to bigger function classes.

First we will assume Φ is of the simple form $\Phi(x, y) = \mathbb{1}_{A \times B}(x, y) = \mathbb{1}_A(x)\mathbb{1}_B(y)$ for $A \in \mathcal{B}(\mathbb{R}^m)$ and

$B \in \mathcal{B}(\mathbb{R}^n)$. Then we get

$$\begin{aligned}
\int_{\mathbb{R}^n} \left(\int_{\mathbb{R}^m} \Phi(x,y) \mathrm{d}\mathbb{P}^{X|Y=y}(x) \right) \mathrm{d}\mathbb{P}^Y(y) &= \int_{\mathbb{R}^n} \left(\int_{\mathbb{R}^m} \mathbb{1}_A(x)\mathbb{1}_B(y) \mathrm{d}\mathbb{P}^{X|Y=y}(x) \right) \mathrm{d}\mathbb{P}^Y(y) \\
&= \int_B \underbrace{\left(\int_{\mathbb{R}^m} \mathbb{1}_A(x) \mathrm{d}\mathbb{P}^{X|Y=y}(x) \right)}_{=\mathbb{P}(X\in A|Y=y)=\mathbb{E}(\mathbb{1}_{\{X\in A\}}|Y=y)} \mathrm{d}\mathbb{P}^Y(y) \\
&= \int_B \mathbb{E}(\mathbb{1}_{\{X\in A\}}|Y=y) \mathrm{d}\mathbb{P}^Y(y) \\
&= \mathbb{E}(\mathbb{1}_{\{X\in A\}}\mathbb{1}_{\{Y\in B\}}) \\
&= \int_{\mathbb{R}^m\times\mathbb{R}^n} \Phi(x,y) \mathrm{d}\mathbb{P}^{(X,Y)}(x,y),
\end{aligned}$$

where we have used Lemma 3.12 in the second to last equality for $\mathbb{1}_{\{X\in A\}}$ instead of X. By using a π-λ argument we can now extend the statement to all measurable indicators $\Phi(x,y) = \mathbb{1}_D(x,y)$, where D lies in the product σ-algebra $\mathcal{B}(\mathbb{R}^m) \otimes \mathcal{B}(\mathbb{R}^n)$. More concretely, the sets $\{A \times B \mid A \in \mathcal{B}(\mathbb{R}^m) \text{ and } B \in \mathcal{B}(\mathbb{R}^n)\}$ are an intersection stable generator (or π-system) of $\mathcal{B}(\mathbb{R}^m) \otimes \mathcal{B}(\mathbb{R}^n)$ and are contained in

$$\mathcal{D} := \{D \in \mathcal{B}(\mathbb{R}^m) \otimes \mathcal{B}(\mathbb{R}^n) \mid \text{equation 11 holds for } \Phi = \mathbb{1}_D\}.$$

By Dynkin's π-λ Theorem we then have that $\mathcal{D} = \mathcal{B}(\mathbb{R}^m) \otimes \mathcal{B}(\mathbb{R}^n)$ and hence equation 11 holds for all measurable indicators. For simple, measurable and positive, and purely measurable functions the statement is now easy to verify using our standard inductive argument. That concludes the proof. □

One can now show that the conditional expectation of X with respect to $Y = y$ is $\mathbb{P}^Y$ a.s. the expectation of X against the measure $\mathbb{P}^{X|Y=y}$.

Corollary 3.14 *In the above setting we $\mathbb{P}^Y$ a.s. have*

$$\mathbb{E}(X|Y=y) = \int_{\mathbb{R}^m} x d\mathbb{P}^{X|Y=y}(x).$$

PROOF We will use the characterization from Lemma 3.12 to prove this. Taking $B \in \mathcal{B}(\mathbb{R}^n)$ arbitrary, define $\Phi_B(x,y) := x \cdot \mathbb{1}_B(y)$. Application of the Disintegration I Theorem then yields

$$\begin{aligned}
\mathbb{E}(X\mathbb{1}_{\{Y\in B\}}) &= \mathbb{E}(\Phi_B(X,Y)) \\
&= \int_{\mathbb{R}^n} \left(\int_{\mathbb{R}^m} \Phi_B(x,y) \mathrm{d}\mathbb{P}^{X|Y=y}(x) \right) \mathrm{d}\mathbb{P}^Y(y) \\
&= \int_B \underbrace{\left(\int_{\mathbb{R}^m} x \mathrm{d}\mathbb{P}^{X|Y=y}(x) \right)}_{=g(y)} \mathrm{d}\mathbb{P}^Y(y).
\end{aligned}$$

Hence the statement follows from Lemma 3.12 as we can see that the function $g(y) := \int_{\mathbb{R}^m} x \mathrm{d}\mathbb{P}^{X|Y=y}(x)$ fulfills the characterizing relation. □

Now we give a useful second version of the Disintegration Theorem, that will be used in future computations.

Theorem 3.15 ***(Disintegration II)*** *Assume that X and Y are random vectors such that for all $y \in \mathbb{R}^n$ the conditional distribution $\mathbb{P}^{X|Y=y}$ exists. Let $\Phi : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$ be a measurable map such that $\mathbb{E}(|\Phi(X,Y)|) < \infty$. Then*

$$\mathbb{E}(\Phi(X,Y)|Y=y) = \mathbb{E}(\Phi(X,y)|Y=y)$$

for $\mathbb{P}^Y$-almost every $y \in Y$.

PROOF We again show the characterizing relation from Lemma 3.12. So take $B \in \mathcal{B}(\mathbb{R}^n)$ arbitrary and define $\hat{\Phi}_B(x,y) := \Phi(x,y)\mathbb{1}_B(y)$. Using Theorem 3.13 we then get

$$\begin{aligned}\mathbb{E}(\Phi(X,Y)\mathbb{1}_{\{Y\in B\}}) &= \mathbb{E}(\hat{\Phi}_B(X,Y)) \\ &= \int_{\mathbb{R}^n}\left(\int_{\mathbb{R}^m}\hat{\Phi}_B(x,y)\mathrm{d}\mathbb{P}^{X|Y=y}(x)\right)\mathrm{d}\mathbb{P}^Y(y) \\ &= \int_B\left(\int_{\mathbb{R}^m}\Phi(x,y)\mathrm{d}\mathbb{P}^{X|Y=y}(x)\right)\mathrm{d}\mathbb{P}^Y(y) \\ &= \int_B \mathbb{E}(\Phi(X,y)|Y=y)\mathrm{d}\mathbb{P}^Y(y).\end{aligned}$$

Hence $\mathbb{E}(\Phi(X,y)|Y=y)$ fulfils the characterizing relation of $\mathbb{E}(\Phi(X,Y)|Y=y)$. That concludes the proof. □

In our applications we will work in the setting of X and Y having a common density. Then we are actually able to determine a regular conditional distribution more concretely.

Lemma 3.16 *Suppose X and Y are random vectors with a common density, hence $\mathbb{P}^{(X,Y)} = f\cdot(\lambda^m\otimes\lambda^n)$. Then the density of Y is given by*

$$f_Y(y) := \int_{\mathbb{R}^m} f(x,y)d\lambda^m(x), \quad y\in\mathbb{R}^n.$$

PROOF This proof is a simple application of Tonelli's Theorem:

$$\mathbb{P}^Y(B) = \mathbb{P}((X,Y)\in\mathbb{R}^m\times B) = \int_{\mathbb{R}^m\times B} f(x,y)\mathrm{d}\lambda^{m+n}(x,y) = \int_B \underbrace{\left(\int_{\mathbb{R}^m} f(x,y)\mathrm{d}\lambda^m(x)\right)}_{=f_Y(y)}\mathrm{d}\lambda^n(y) \text{ for all } B.$$

□

Theorem 3.17 *Staying in the previous setting, if there is $y\in\mathbb{R}^n$ such that $f_Y(y)>0$, then the function defined by*

$$f_{X|Y=y}(x) := \frac{f(x,y)}{f_Y(y)}, \quad x\in\mathbb{R}^m,$$

is a probability density (with respect to λ^m). For $y\in\mathbb{R}^n$ and $A\in\mathcal{B}(\mathbb{R}^m)$ the mapping

$$(y,A)\longmapsto\begin{cases}\int_A f_{X|Y=y}(x)d\lambda^m(x), & f_Y(y)>0,\\ \delta_0(A), & f_Y(y)=0,\end{cases}$$

is $\mathbb{P}^Y$ a.s. the conditional distribution of X wrt. $Y=y$, which we will denote by $\mathbb{P}^{X|Y=y}$ (as usual). By δ_0 we denote the Dirac measure at 0:

$$\delta_0(A) := \begin{cases}1 & \text{if } 0\in A,\\ 0 & \text{else.}\end{cases}$$

PROOF We first prove the first assertion.

$$\int_{\mathbb{R}^m} f_{X|Y=y}(x)\mathrm{d}\lambda^m(x) = \int_{\mathbb{R}^m}\frac{f(x,y)}{f_Y(y)}(x)\mathrm{d}\lambda^m(x) = \frac{1}{f_Y(y)}\int_{\mathbb{R}^m} f(x,y)\mathrm{d}\lambda^m(x) = \frac{f_Y(y)}{f_Y(y)} = 1.$$

Now we prove the second assertion. For $y\in\mathbb{R}^n$ arbitrary we easily see $\mathbb{P}^{X|Y=y}$ to be a probability measure

(in both cases).

It remains to show the second property in Definition 3.8. Take $A \in \mathcal{B}(\mathbb{R}^m)$ arbitrary. We will again use Lemma 3.12 to prove the statement. The measurability follows from the fact that $\{f_Y > 0\}$ and $\{f_Y = 0\}$ are measurable sets (since f_Y is a density and hence itself measurable), that $\delta_0(A)$ is constant for fixed A, and Fubini's Theorem. For fixed A we rewrite

$$\mathbb{P}^{X|Y=y}(A) = \mathbb{1}_{\{f_Y>0\}}(y)\frac{1}{f_Y(y)}\int_A f(x,y)\mathrm{d}\lambda^m(x) + \mathbb{1}_{\{f_Y=0\}}(y)\underbrace{\delta_0(A)}_{\text{constant}}, \tag{12}$$

which is easily seen to be measurable (with Fubini's Theorem for the integral). Now take $B \in \mathcal{B}(\mathbb{R}^n)$. We know $\mathbb{P}^Y(f_Y = 0) = 0$, as

$$\mathbb{P}^Y(f_Y = 0) = \int_{\{f_Y=0\}} f_Y(y)\,\mathrm{d}\lambda^n(y) = \int_{\{f_Y=0\}} 0\;\mathrm{d}\lambda^n(y) = 0.$$

Using (12) we get

$$\begin{aligned}
\int_B \mathbb{P}^{X|Y=y}(A)\mathrm{d}\mathbb{P}^Y(y) &= \int_B \left(\mathbb{1}_{\{f_Y>0\}}(y)\int_A f_{X|Y=y}(x)\mathrm{d}\lambda^m(x) + \mathbb{1}_{\{f_Y=0\}}(y)\delta_0(A)\right)\mathrm{d}\mathbb{P}^Y(y)\\
&= \int_{\{Y\in B\}}\left(\int_A f_{X|Y=y}(x)\mathrm{d}\lambda^m(x)\right) f_Y(y)\mathrm{d}\lambda^n(y)\\
&= \int_{\{Y\in B\}}\int_A f(x,y)\mathrm{d}\lambda^m(x)\mathrm{d}\lambda^n(y)\\
&= \mathbb{E}(\mathbb{1}_{\{X\in A\}}\mathbb{1}_{\{Y\in B\}}).
\end{aligned}$$

Hence we see that our just defined mapping $\mathbb{P}^{X|Y=y}(A)$ equals $\mathbb{E}(\mathbb{1}_{\{X\in A\}}|Y = y)$ by Lemma 3.12 where we apply the lemma to $\mathbb{1}_{\{X\in A\}}$ instead of X. □

For conditional densities there also holds a kind of Bayes Theorem, that we will make use of frequently.

Theorem 3.18 *In our setting with X and Y having a common density, we have*

$$f_{X|Y=y}(x)f_Y(y) = f_{Y|X=x}(y)f_X(x)$$

for all $x \in \mathbb{R}^m$ and $y \in \mathbb{R}^n$.

Proof Let us first assume that $f_Y(y) > 0, f_X(x) > 0$. Then we get

$$f_{X|Y=y}(x) = \frac{f(x,y)}{f_Y(y)} = \frac{f(x,y)f_X(x)}{f_Y(y)f_X(x)} = \frac{f_{Y|X=x}(y)f_X(x)}{f_Y(y)}.$$

Multiplying by $f_Y(y)$ does the trick.

Now assume that $f_Y(y) = 0, f_X(x) > 0$. Then obviously $f_{X|Y=y}(x)f_Y(y) = 0$. But also $f_{Y|X=x}(y) = 0$, hence $f_{Y|X=x}(y)f_X(x) = 0$.

If $f_Y(y) = f_X(x) = 0$ the statement follows trivially. All other cases follow by symmetry. □

Remark We use Theorem 3.18 in the form

$$f_{X|Y=y}(x) = \frac{f_{Y|X=x}(y)f_X(x)}{\int_{\mathbb{R}^m} f_{Y|X=v}(y)f_X(v)\mathrm{d}v},$$

where $\int_{\mathbb{R}^m} f_{Y|X=v}(y) f_X(v) \mathrm{d}v = f_Y(y)$ as can be easily verified by the calculation

$$\int_{\mathbb{R}^m} f_{Y|X=v}(y) f_X(v) \mathrm{d}v \; = \int_{\mathbb{R}^m} \frac{f(v,y)}{f_X(v)} f_X(v) \mathrm{d}v \; = f_Y(y).$$

3.2 Bayesian Inverse Problems

After having laid the foundation, we will now cast CT as a Bayesian inverse problem. We are considering random vectors X, Y, and N, where X and N are independent and Y is given by the relation

$$Y = AX + N. \tag{13}$$

What we are interested in is finding out about the true distribution of X, given a made measurement $Y = y$. In other words, we are interested in the conditional distribution $\mathbb{P}^{X|Y=y}$. Knowledge of $\mathbb{P}^{X|Y=y}$ will enable us to extract a point estimate of x by defining the so called MAP estimate, which will then serve as our solution for the inverse problem.

Definition 3.19 If X and Y are random vectors such that $\mathbb{P}^{X|Y=y}$ has a density, then we define the maximum a posteriori (MAP) estimator of X with respect to $Y = y$ as

$$\hat{X}_{\mathrm{MAP}}(y) := \arg\max_{\mathrm{x}} f_{X|Y=y}(x).$$

We will assume $X \sim \mathbb{P}^X$ and $N \sim \mathbb{P}^N$ follow us known distributions that are absolutely continuous with respect to the Lebesgue measure, i.e. $\mathbb{P}^X << \lambda^m$ and $\mathbb{P}^N << \lambda^n$. Then we know that these two measures both have densities f_X and f_N respectively. $\mathbb{P}^X$ is usually called the prior distribution and reflects our prior belief about the unknown vector that is to be estimated. $\mathbb{P}^N$ is the noise distribution. For $x \in \mathbb{R}^m$ we then get

$$\begin{aligned} \mathbb{P}^{Y|X=x} &= \mathbb{P}^{AX+N|X=x} \\ &= \mathbb{P}^{Ax+N|X=x} \\ &= \mathbb{P}^{N+Ax} \\ &= \mathbb{P}^N(\cdot - Ax), \end{aligned}$$

where the second equality comes from Theorem 3.15 and the second to last equality comes from the independence of X and N, as well as the fact that Ax is constant. Hence we see that the density of $\mathbb{P}^{Y|X=x}$ takes on the following simple form:

$$f_{Y|X=x}(y) = f_N(y - Ax).$$

Using now Bayes theorem for densities, we derive

$$f_{X|Y=y}(x) = \frac{f_{Y|X=x}(y) f_X(x)}{\int_{\mathbb{R}^m} f_{Y|X=v}(y) f_X(v) \mathrm{d}v} = \frac{f_N(y - Ax) f_X(x)}{\int_{\mathbb{R}^m} f_N(y - Av) f_X(v) \mathrm{d}v}. \tag{14}$$

Now we notice that when computing the MAP of X with respect to $Y = y$ that we can ignore the denominator (since it does not depend on x) and hence we get

$$\hat{X}_{\mathrm{MAP}}(y) = \arg\max_{\mathrm{x}} f_N(y - Ax) f_X(x).$$

Now if we assume that $X \sim \mathcal{N}(0, \frac{1}{\lambda} I_m)$ and $N \sim \mathcal{N}(0, I_n)$ as prior and noise distribution, then we get

$$f_N(y - Ax) f_X(x) \propto \exp(-\frac{1}{2}||y - Ax||_2^2 - \frac{\lambda}{2}||x||_2^2),$$

taking the logarithm (a monotone function) and dividing by $-\frac{1}{2}$ yields

$$\hat{X}_{\text{MAP}}(y) = \arg\min_{\text{x}} ||Ax - y||_2^2 + \lambda||x||_2^2.$$

Hence we recover the Tikhonov regularization (5). Using this Bayesian approach lets us now also infer about the optimal regularization parameter λ. For this we will assume that λ is a hyper parameter realization, hence $\Lambda = \lambda$ for Λ a random vector with its own prior distribution $\Lambda \sim \mathbb{P}^{\Lambda}$. We can also assume that Λ is independent from N. Using the disintegration corollary for $\Lambda = \lambda$ and noting that generally

$$\mathbb{P}^{(Y|X=x)|\Lambda=\lambda} = \mathbb{P}^{Y|X=x,\Lambda=\lambda},$$

all the derivations prior hold, although they should technically all be seen conditioned on $\Lambda = \lambda$.

3.3 Choosing the Regularization Parameter

Guided by the result in [HF16], minimization problem (5), we will now derive a criterion for choosing the proper regularization parameter λ. For other methods we refer to [SS18]. Assuming a uniform prior on Λ, hence $\Lambda \sim \mathcal{U}(0, b)$, we can again use Bayes theorem for densities to derive a MAP estimate for Λ:

$$\hat{\Lambda}_{\text{MAP}} = \arg\max_{\lambda \in (0,b)} f_{\Lambda|Y=y}(\lambda) = \arg\max_{\lambda \in (0,b)} f_{Y|\Lambda=\lambda}(y)\, f_{\Lambda}(\lambda) = \arg\max_{\lambda \in (0,b)} f_{Y|\Lambda=\lambda}(y).$$

To evaluate $f_{Y|\Lambda=\lambda}(y)$, we need a Bayesian integration:

$$f_{Y|\Lambda=\lambda}(y) = \int_{\mathbb{R}^m} f_{Y|X=x,\Lambda=\lambda}(y) f_{X|\Lambda=\lambda}(x) \mathrm{d}x.$$

We have $\mathbb{P}^{Y|X=x,\Lambda=\lambda} \sim \mathcal{N}(Ax, I_n)$ and $\mathbb{P}^{X|\Lambda=\lambda} \sim \mathcal{N}(0, \frac{1}{\lambda} I_m)$, hence

$$f_{Y|X=x,\Lambda=\lambda}(y) = \frac{1}{(2\pi)^{\frac{n}{2}}} \exp\left(-\frac{1}{2}||Ax - y||_2^2\right) \quad \text{and} \quad f_{X|\Lambda=\lambda}(x) = (\frac{\lambda}{2\pi})^{\frac{m}{2}} \exp\left(-\frac{\lambda}{2}||x||_2^2\right).$$

Thus the integral we need to evaluate becomes

$$f_{Y|\Lambda=\lambda}(y) = \frac{\lambda^{\frac{m}{2}}}{(2\pi)^{\frac{m+n}{2}}} \int_{\mathbb{R}^m} \exp\left(-\frac{1}{2}||Ax - y||_2^2 - \frac{\lambda}{2}||x||_2^2\right) \mathrm{d}x. \tag{15}$$

To solve (15) we make the following calculation, assuming that $\hat{x}_\lambda$ solves the Tikhonov normal equation, hence assuming that $\hat{x}_\lambda = (A^T A + \lambda I_m)^{-1} A^T y$ holds:

$$\begin{aligned}
&(y - Ax)^T (y - Ax) + \lambda x^T x \\
&= y^T y - x^T A^T y - y^T A x + x^T A^T A x + \lambda x^T x \\
&= y^T y - x^T (A^T A + \lambda I_m)(A^T A + \lambda I_m)^{-1} A^T y - y^T A (A^T A + \lambda I_m)^{-1} (A^T A + \lambda I_m) x + x^T (A^T A + \lambda I_m) x \\
&= y^T y - x^T (A^T A + \lambda I_m) \hat{x}_\lambda - \hat{x}_\lambda^T (A^T A + \lambda I_m) x + x^T (A^T A + \lambda I_m) x \\
&= y^T y - \hat{x}_\lambda^T (A^T A + \lambda I_m) \hat{x}_\lambda + (x - \hat{x}_\lambda)^T (A^T A + \lambda I_m)(x - \hat{x}_\lambda).
\end{aligned}$$

Plugging in $x = \hat{x}_\lambda$ into the prior calculation we further see $||y - A\hat{x}_\lambda||_2^2 + \lambda||\hat{x}_\lambda||_2^2 = y^T y - \hat{x}_\lambda^T (A^T A + \lambda I_m) \hat{x}_\lambda$.

Overall, we get

$$(y-Ax)^T(y-Ax)+\lambda x^T x = ||y-A\hat{x}_\lambda||_2^2+\lambda||\hat{x}_\lambda||_2^2+(x-\hat{x}_\lambda)^T(A^TA+\lambda I_m)(x-\hat{x}_\lambda).$$

Now we can resubstitute into the integral (15) and pull out the first and second summand, as they do not depend on x:

$$\frac{\lambda^{\frac{m}{2}}}{(2\pi)^{\frac{m+n}{2}}}\int_{\mathbb{R}^m}\exp\left(-\frac{1}{2}||Ax-y||_2^2-\frac{\lambda}{2}||x||_2^2\right)\mathrm{d}x$$

$$=\frac{\lambda^{\frac{m}{2}}\exp\left(-\frac{1}{2}||A\hat{x}_\lambda-y||_2^2-\frac{\lambda}{2}||\hat{x}_\lambda||_2^2\right)}{(2\pi)^{\frac{m+n}{2}}}\int_{\mathbb{R}^m}\exp\left(-\frac{1}{2}(x-\hat{x}_\lambda)^T(A^TA+\lambda I_m)(x-\hat{x}_\lambda)\right)\mathrm{d}x.$$

If we imagine a random vector $Z\sim\mathcal{N}(\hat{x}_\lambda,(A^TA+\lambda I_m)^{-1})$, then we know Z has density

$$f_Z(z)=\frac{\sqrt{\det(A^TA+\lambda I_m)}}{(2\pi)^{\frac{m}{2}}}\exp\left(-\frac{1}{2}(z-\hat{x}_\lambda)^T(A^TA+\lambda I_m)(z-\hat{x}_\lambda)\right).$$

Since we know that probability densities integrate to 1 we get

$$1=\int_{\mathbb{R}^m}f_Z(z)\mathrm{d}z=\int_{\mathbb{R}^m}\frac{\sqrt{\det(A^TA+\lambda I_m)}}{(2\pi)^{\frac{m}{2}}}\exp\left(-\frac{1}{2}(z-\hat{x}_\lambda)^T(A^TA+\lambda I_m)(z-\hat{x}_\lambda)\right)\mathrm{d}z.$$

This in turn implies that

$$\int_{\mathbb{R}^m}\exp\left(-\frac{1}{2}(x-\hat{x}_\lambda)^T(A^TA+\lambda I_m)(x-\hat{x}_\lambda)\right)\mathrm{d}x=\frac{(2\pi)^{\frac{m}{2}}}{\sqrt{\det(A^TA+\lambda I_m)}}.$$

Together with the other factors we get

$$f_{Y|\Lambda=\lambda}(y)=\frac{\exp\left(-\frac{1}{2}||A\hat{x}_\lambda-y||_2^2-\frac{\lambda}{2}||\hat{x}_\lambda||_2^2\right)}{(2\pi)^{\frac{n}{2}}\sqrt{\det(A^TA+\lambda I_m)}}.$$

Hence

$$\begin{aligned}\hat{\Lambda}_{\mathrm{MAP}}&=\underset{\lambda\in(0,b)}{\arg\max}\ f_{Y|\Lambda=\lambda}(y)=\underset{\lambda\in(0,b)}{\arg\min}\ [-2\ln f_{Y|\Lambda=\lambda}(y)]\\&=\underset{\lambda\in(0,b)}{\arg\min}\ ||A\hat{x}_\lambda-y||_2^2+\lambda||\hat{x}_\lambda||_2^2+n\ln 2\pi+\ln\det\left(\frac{A^TA}{\lambda}+I_m\right).\end{aligned}\tag{16}$$

While this function is computationally intensive to optimize, it gives an objective criteria for choosing the right regularization parameter. We finally then get

$$\hat{X}_{\mathrm{MAP}}(y)=\hat{x}_{\lambda^*},$$

where $\lambda^*=\hat{\Lambda}_{\mathrm{MAP}}(y)$.

4 Numerical Algorithms

We will now in this chapter briefly derive the numerical algorithms needed in solving the reconstruction and in denoising the sinograms.

4.1 Conjugated Gradient Method (CG Method)

In Chapter 2 we derived (7) to solve for the reconstruction, namely

$$(A^TA + \lambda I_m)x = A^Ty,$$

which we can interpret as solving $Mx = b$ for $M = (A^TA + \lambda I_m)$ and $b = A^Ty$. Due to Proposition 2.5, the matrix M is invertible and the solution to (7) is hence known to exist and be unique. We previously denoted it by $\hat{x}_\lambda$. Our goal now is to derive an algorithm to find $\hat{x}_\lambda$. Since M is self-adjoint and positive definite, we want to use the CG method, for which we will derive an algorithm. For a more detailed algorithmic analysis and derivation we refer to [DH19], Chapter 8.3. The solution $\hat{x}_\lambda := M^{-1}b$ of the equation is the uniquely determined minimum of the quadratic energy functional

$$\Phi(x) := \frac{1}{2}x^TMx - x^Tb, \quad x \in \mathbb{R}^m. \tag{17}$$

As a quick digression, we introduce the inner product with respect to a positive definite and self-adjoint matrix.

Definition 4.1 Suppose $M \in \mathbb{R}^{m\times m}$ is positive definite and self-adjoint. Then the inner product $\langle\cdot,\cdot\rangle_M$ with respect to M is defined as

$$\langle u, v\rangle_M := u^TMv.$$

The corresponding norm is denoted as $\|\cdot\|_M$. We call two vectors u and v M-orthogonal if they are orthogonal with respect to $\langle\cdot,\cdot\rangle_M$.

We will not prove that $\langle\cdot,\cdot\rangle_M$ is an inner product, as this is elementary. Using now Definition 4.1 and $M\hat{x}_\lambda = b$ or $\hat{x}_\lambda = M^{-1}b$, we can see

$$\begin{aligned}
\Phi(x) - \Phi(\hat{x}_\lambda) &= \frac{1}{2}x^TMx - x^Tb - \frac{1}{2}\hat{x}_\lambda^TM\hat{x}_\lambda + \hat{x}_\lambda^Tb \\
&= \underbrace{\frac{1}{2}x^TMx - x^TM\hat{x}_\lambda + \frac{1}{2}\hat{x}_\lambda^TM\hat{x}_\lambda}_{=\frac{1}{2}(x-\hat{x}_\lambda)^TM(x-\hat{x}_\lambda)} + x^TM\hat{x}_\lambda - \hat{x}_\lambda^TM\hat{x}_\lambda - x^Tb + \hat{x}_\lambda^Tb \\
&= \frac{1}{2}\underbrace{(x-\hat{x}_\lambda)^TM(x-\hat{x}_\lambda)}_{=\|x-\hat{x}_\lambda\|_M^2} + \underbrace{x^T\underbrace{M\hat{x}_\lambda}_{=b} - x^Tb}_{=0} \underbrace{- \hat{x}_\lambda^T\underbrace{M\hat{x}_\lambda}_{=b} + \hat{x}_\lambda^Tb}_{=0} \\
&= \frac{1}{2}\|x-\hat{x}_\lambda\|_M^2.
\end{aligned}$$

Geometrically, this means that the graph of the function Φ with respect to the energy norm is a circular paraboloid with its center at $\hat{x}_\lambda$. This observation forms the basis of our derivation of the conjugate gradient method. Starting from an approximate solution x_k, in the k-th iteration step, we first determine a search direction $d_k \in \mathbb{R}^m \setminus \{0\}$ and choose the next iterate using the ansatz

$$x_{k+1} = x_k + \alpha_k d_k.$$

Here, we choose the step size $\alpha_k \in \mathbb{R}$ as the minimum of the function

$$f(\alpha) := \Phi(x_k + \alpha d_k) = \Phi(x_k) + \alpha d_k^T M x_k - d_k^T b + \frac{\alpha^2}{2} d_k^T M d_k.$$

Since $d_k^T M d_k > 0$, we obtain the minimum of f by solving the equation $f'(\alpha_k) = 0$ for α_k. Let us denote

$$r_k := b - M x_k$$

as the k-th residual, hence

$$\alpha_k := \frac{r_k^T d_k}{d_k^T M d_k}.$$

The choice of search directions d_k is still open. One possible option would be $d_k = -\nabla\Phi(x_k)$, the direction of steepest descent of Φ. Since

$$\Phi(x + p) = \Phi(x) + p^T (Mx - b) + \underbrace{\frac{1}{2} p^T M p}_{=\mathcal{O}(\|p\|_2^2)},$$

we have $\nabla\Phi(x) = Mx - b$, thus $-\nabla\Phi(x_k) = r_k$.

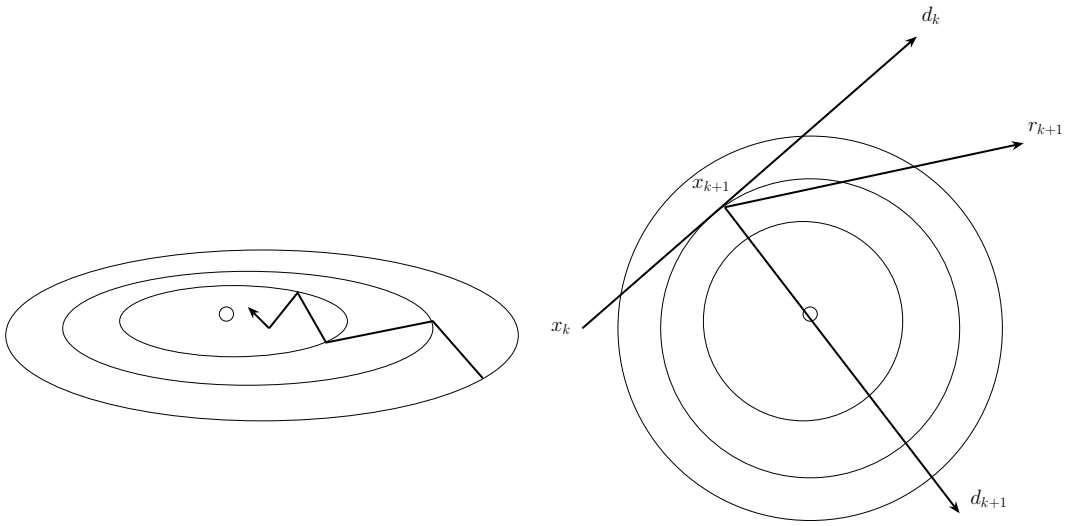


Figure 4: Search directions with negative gradients (left) and with conjugate gradients (right). The ellipses (circles) are the level sets of Φ in the euclidean geometry (left) and the $\langle\cdot,\cdot\rangle_M$-geometry (right).

However, this may not be the best choice for the search direction, as illustrated by the left image in figure 4 in a two-dimensional example. The right image shows the contour lines of Φ in the plane spanned by d_k and r_{k+1}. With respect to the $\|\cdot\|_M$ geometry, the contour lines are circles. Due to the minimality property of α_k, x_{k+1} is the point where the line $\{x_k + \alpha d_k : \alpha \in \mathbb{R}\}$ tangentially touches a contour line. By choosing d_{k+1} orthogonal to d_k with respect to the $\langle\cdot,\cdot\rangle_M$ inner product, x_{k+2} lies at the center of the circle. This leads us to the approach

$$d_{k+1} = r_{k+1} + \beta_k d_k,$$

where we aim to determine β_k such that

$$\langle d_{k+1}, d_k\rangle_M = 0.$$

Thus, we obtain the equation

$$\beta_k = -\frac{r_{k+1}^T M d_k}{d_k^T M d_k}.$$

Since the search directions d_k are chosen to be M-conjugate to each other, the method we have just derived is called the conjugate gradient method or CG method for short. We will now summarize our derivation in an algorithm.

Conjugate Gradient method (provisional)

Initialization : $M \in \mathbb{R}^{m\times m}$ self-adjoint and positive definite, $b \in \mathbb{R}^m$ right-hand side, $x_0 \in \mathbb{R}^m$ initial vector

$k = 0$; $r_0 := b - Mx_0$; $d_0 := r_0$;

while $r_k \neq 0$ **do**

- $\alpha_k := \frac{r_k^T d_k}{d_k^T M d_k}$;
- $x_{k+1} := x_k + \alpha_k d_k$;
- $r_{k+1} := b - Mx_{k+1}$;
- $\beta_k := -\frac{r_{k+1}^T M d_k}{d_k^T M d_k}$;
- $d_{k+1} := r_{k+1} + \beta_k d_k$;
- $k := k + 1$;

end

Termination : If the termination condition $r_k = 0$ is fulfilled, then x_k is the exact solution of the equation $Mx = b$.

One can show that this version of the CG method will terminate in m steps at the latest. See again Chapter 8.3 of [DH19]. However, the algorithm is not the one used in practice, as rounding errors can disrupt the M-orthogonality of the vectors and because m can be much larger than the amount of steps we want the algorithm to run. We are more interested in a termination condition based on a tolerance level ϵ. Hence we will consider the norm of the residual $r_k = M(\hat{x}_\lambda - x_k) = b - Mx_k$. One should be aware of the fact that $\|r_k\|_2$ can have a different order of magnitude than $\|\hat{x}_\lambda - x_k\|_2$, and reasoning about $\|\hat{x}_\lambda - x_k\|_2$ would require a bound on $\|M^{-1}\|_2$, as $\|\hat{x}_\lambda - x_k\|_2 = \|M^{-1}M(\hat{x}_\lambda - x_k)\|_2 = \|M^{-1}r_k\|_2 \leq \|M^{-1}\|_2 \cdot \|r_k\|_2$. For implementation we will not use formulas in the above algorithm for α_k and β_k given, but instead

$$\alpha_k := \frac{\|r_k\|_2^2}{d_k^T M d_k} \quad \text{and} \quad \beta_k := \frac{\|r_{k+1}\|_2^2}{\|r_k\|_2^2},$$

which can be attained in the following way.

$$\text{For } \alpha_k: \; r_k^T d_k = r_k^T r_k + \beta_k r_k^T d_{k-1} = \|r_k\|_2^2,$$
$$\text{for } \beta_k: \; r_{k+1}^T M d_k = \frac{1}{\alpha_k}(r_{k+1}^T r_k - r_{k+1}^T r_{k+1}) = -\frac{1}{\alpha_k}\|r_{k+1}\|_2^2 = -\frac{\|r_{k+1}\|_2^2}{\|r_k\|_2^2} r_k^T M d_k.$$

With these we also already calculate $\|r_{k+1}\|_2^2$ and Md_k before needing them. Instead of using $r_{k+1} = b - Mx_{k+1}$, we further use

$$r_{k+1} = r_k - \alpha_k M d_k.$$

This holds since

$$r_{k+1} = b - Mx_{k+1} = b - M(x_k + \alpha_k d_k) = r_k - \alpha_k M d_k.$$

Thus we save another matrix multiplication. Overall we then get the following algorithm.

Conjugate Gradient method (final)

Initialization : $M \in \mathbb{R}^{m\times m}$ self-adjoint and positive definite, $b \in \mathbb{R}^m$ right-hand side, $x_0 \in \mathbb{R}^m$ initial vector, ϵ tolerance level

$k = 0$, $r_0 := b - Mx_0$, $d_0 := r_0$;

while $\|r_k\|_2 > \epsilon$ **do**

$\alpha_k := \frac{\|r_k\|_2^2}{d_k^T M d_k}$; `// Saving` Md_k

$x_{k+1} := x_k + \alpha_k d_k$;

$r_{k+1} := r_k - \alpha_k M d_k$;

$\beta_k := \frac{\|r_{k+1}\|_2^2}{\|r_k\|_2^2}$; `// Saving` $\|r_{k+1}\|_2^2$

$d_{k+1} := r_{k+1} + \beta_k d_k$;

$k := k + 1$;

end

Termination : If the termination condition $\|r_k\|_2 \leq \epsilon$ is fulfilled, then x_k is an approximation of $M^{-1}b$.

Without proof we will cite from [DH19] the error bound

$$\|\hat{x}_\lambda - x_k\|_M \leq 2\left(\frac{\sqrt{\kappa(M)}-1}{\sqrt{\kappa(M)}+1}\right)^k \|\hat{x}_\lambda - x_0\|_M,$$

where $\kappa(M)$ is the condition number of M:

$$\kappa(M) := \|M\|_2\|M^{-1}\|_2.$$

This bound shows the convergence of our algorithm.

4.2 Levenberg–Marquardt Algorithm

We will now cover the Levenberg–Marquardt method. This method gets used when having data (x_i, y_i), $i = 1, ..., n$ and wanting to find the parameters $\beta = (\beta_1, ..., \beta_m)^T$ such that the function

$$f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$$
$$(x, \beta) \mapsto f(x, \beta)$$

has minimal error with respect to the data. Here f represents a family of curves that we are trying to optimally fit. Hence we are interested in finding

$$\hat{\beta} = \underset{\beta\in\mathbb{R}^m}{\arg\min}\ \frac{1}{2}\sum_{i=1}^{n}(f(x_i,\beta) - y_i)^2 = \underset{\beta\in\mathbb{R}^m}{\arg\min}\ F(\beta),$$

where $F(\beta) = \frac{1}{2}\sum_{i=1}^{n}(f(x_i,\beta) - y_i)^2$. The Levenberg–Marquardt algorithm is a trust-region iterative method to minimize F. Our derivation of it will follow [NW06]. For a more detailed analysis of it we again refer to [NW06], Chapter 4 and Chapter 10.

Similar to line search methods, trust-region methods utilize a quadratic model of the objective function to generate steps; however, line search methods and trust-region methods employ this model in different ways. Line search methods leverage the model to determine a search direction and subsequently focus on finding an appropriate step length along that direction. In contrast, trust-region methods establish a region around the current iterate within which they trust the model's adequacy as a representation of

the objective function. They then select the step as the approximate minimizer of the model within this region, effectively determining both the direction and length of the step simultaneously. Whenever the size of the trust region is modified, the direction of the step also changes.

The size of the trust region plays a crucial role in the effectiveness of each step. If the region is excessively small, the algorithm may miss the opportunity to take a substantial step that would significantly advance toward the objective function's minimizer. Conversely, if the region is too large, the minimizer of the model may be distant from the minimizer of the objective function within that region, necessitating a reduction in region size and subsequent reevaluation.

In practice, the size of the trust region is adjusted based on the algorithm's performance in prior iterations. When the model consistently yields reliable results, generating good steps and accurately predicting the objective function's behavior, the trust region can be expanded to allow for longer and more ambitious steps. A failed step indicates that the model is an inadequate representation of the objective function within the current trust region. In such cases, the region size is reduced, and the process is repeated.

For f a twice differentiable function, so is F. At each iterate β_k we will now assume the model function m_k to be quadratic. More accurately, m_k is based on the Taylor series expansion of F around β_k, which is

$$F(\beta_k + p) = F_k + g_k^T p + \frac{1}{2} p^T \nabla^2 F(\beta_k + tp) p,$$

where $F_k = F(\beta_k)$ and $g_k = \nabla F(\beta_k)$, and t is some scalar in the unit interval. By using an approximation B_k to the Hessian in the second-order term, m_k is defined as follows:

$$m_k(p) = F_k + g_k^T p + \frac{1}{2} p^T B_k p,$$

where B_k is some symmetric matrix. The difference between $m_k(p)$ and $F(\beta_k + p)$ is $\mathcal{O}(\|p\|^2)$, which is small when p is small.

To obtain each step, we seek a solution of the subproblem

$$\begin{aligned} \min_{p \in \mathbb{R}^m} \quad & m_k(p) = F_k + g_k^T p + \frac{1}{2} p^T B_k p \\ \text{s.t.} \quad & \|p\|_2 \leq \Delta_k, \end{aligned}$$

where $\Delta_k > 0$ is the trust-region radius. Thus, the trust-region approach requires solving a sequence of subproblems where both the objective function and constraint are quadratic, expressed as $p^T p \leq \Delta_k^2$.

An important part in trust-region methods plays the radius Δ_k. This choice will for us be based on the agreement of F to the model function m_k. We define the ratio

$$\rho_k := \frac{F(\beta_k) - F(\beta_k + p_k)}{m_k(0) - m_k(p_k)}. \tag{18}$$

The numerator is called actual reduction and the denominator is called the predicted reduction. Note that the predicted reduction is always non-negative since $p = 0$ is in the trust region. If ρ_k is negative, then we thus know that the actual prediction is negative and we will reject β_{k+1}. If ρ_k is close to 1 the model function and F agree well with each other. In that instance we can increase the radius further in the next step. If ρ_k is positive but close to 0 the improvement is only small and we will shrink the radius in the next iteration. Here is an algorithm to summarize our considerations.

Trust Region for Step $k+1$

Initialization : $\hat{\Delta} > 0, \rho_k$ the approximation ratio, $\eta \in [0, \frac{1}{4})$, β_k current position, p_k step direction, Δ_k current radius

if $\rho_k < \frac{1}{4}$ **then**
  $\Delta_{k+1} = \frac{1}{4}\Delta_k$;
end
else
  if $\rho_k > \frac{3}{4}$ *and* $\|p_k\| = \Delta_k$ **then**
    $\Delta_{k+1} = \min(2\Delta_k, \hat{\Delta})$;
  end
  else
    $\Delta_{k+1} = \Delta_k$;
  end
end
if $\rho_k > \eta$ **then**
  $\beta_{k+1} = \beta_k + p_k$;
end
else
  $\beta_{k+1} = \beta_k$;
end

Termination : We return the new radius Δ_{k+1} and the new position β_{k+1}.

Let us now discuss how to choose the appropriate step direction p_k. We define the residual function $r(\beta) := (f(x,\beta) - y)$ and further $r_k := (f(x,\beta_k) - y)$ to be the k-th residual. Note that $g_k = \nabla F(\beta_k) = J_k^T r_k$ if we define $J_k = Dr(\beta_k) = \left(\frac{\partial r_j}{\partial \beta_i}(\beta_k)\right)_{\substack{j=1,\dots,n \\ i=1,\dots,m}}$. We will use the Hessian approximation

$$B_k = J_k^T J_k = Dr(\beta_k)^T Dr(\beta_k).$$

With that choice of the Hessian we have the following minimization problem.

Lemma 4.2 *Equivalent are the minimization problems*

$$\begin{aligned} \min_{p \in \mathbb{R}^m} \quad & m_k(p) \\ s.t. \quad & \|p\|_2 \leq \Delta_k, \end{aligned}$$

and

$$\begin{aligned} \min_{p \in \mathbb{R}^m} \quad & \frac{1}{2}\|J_k p + r_k\|_2^2 \\ s.t. \quad & \|p\|_2 \leq \Delta_k. \end{aligned} \tag{19}$$

Proof We calculate

$$\begin{aligned} \frac{1}{2}\|J_k p + r_k\|_2^2 &= \frac{1}{2}\|J_k p + (f(x,\beta_k) - y)\|_2^2 \\ &= \frac{1}{2}\left(J_k p + (f(x,\beta_k) - y)\right)^T \left(J_k p + (f(x,\beta_k) - y)\right) \\ &= \underbrace{\frac{1}{2}(f(x,\beta_k) - y)^T (f(x,\beta_k) - y)}_{=F_k} - \underbrace{(f(x,\beta_k) - y)^T J_k}_{=-g_k^T} p + \frac{1}{2} p^T \underbrace{J_k^T J_k}_{=B_k} p \\ &= m_k(p). \end{aligned}$$

□

Problem (19) now can be solved with the help of the following theorem.

Theorem 4.3 *The vector $\hat{p} \in \mathbb{R}^m$ solves problem (19) if and only if $\|\hat{p}\|_2 \leq \Delta_k$ and there exists a scalar $\lambda \geq 0$ such that*

$$(J_k^T J_k + \lambda I_m)\hat{p} = -g_k = -J_k^T r_k \quad \textit{and} \tag{20}$$

$$\lambda(\Delta_k - \|\hat{p}\|_2) = 0. \tag{21}$$

PROOF The proof of this theorem can be found in [NW06], Lemma 10.2. □

We need a way to actually find the λ from Theorem 4.3. An algorithm outlined in [NW06] will be presented now. We define a vector

$$p_k(\lambda) = -(J_k^T J_k + \lambda I_m)^{-1} g_k$$

that will fulfill equation (20) from Theorem 4.3. We look at $\lambda \mapsto \|p_k(\lambda)\|_2$ and try to find λ such that $\|p_k(\lambda)\|_2 = \Delta_k$ as a one dimensional root finding problem. For finding the root, we quote the following algorithm from [NW06], Algorithm 4.3, dropping the indices of B_k and Δ_k for more readability.

Relaxation Size Parameter λ

Initialization : Given λ_0, Δ, B.

for $\ell = 0, 1, 2, \ldots$ **do**

Cholesky factor $(B + \lambda_\ell I_m) = L^T L$;

Solve $L^T L p_\ell = -g_k$, $L^T q_\ell = p_\ell$;

Set $\lambda_{\ell+1} = \lambda_\ell + \left(\frac{\|p_\ell\|_2}{\|q_\ell\|_2}\right)^2 \left(\frac{\|p_\ell\|_2 - \Delta}{\Delta}\right)$;

end

Termination : We return the new $\lambda = \lambda_\ell$ and $p = p_\ell$.

Putting it all together we get the following algorithm.

Levenberg–Marquardt Algorithm

Initialization : Given initial guesses $\beta_0, \lambda_0, \Delta_0$, constants $\hat{\Delta} > 0, \eta \in [0, \frac{1}{4})$.

$J_0 = Dr(\beta_0)$ and $r_0 = (f(x, \beta_0) - y)$;

for $k = 0, 1, 2, \ldots$ **do**

Calculate $g_k = J_k^T r_k$ and $B_k = J_k^T J_k$;

Solve $B_k p = -g_k$;

if $\|p\|_2 \leq \Delta_k$ **then**

Set $p_k = p$;

end

else

Calculate p_k and λ_k from Algorithm 4 with λ_0, Δ_0 and B_k as initial values;

end

Calculate ρ_k by equation (18);

Update the new β_{k+1} and Δ_{k+1} through Algorithm 3 with $\hat{\Delta}, \rho_k, \eta, \beta_k, p_k$ and Δ_k as initial values;

Calculate $J_{k+1} = Dr(\beta_k)$ and $r_{k+1} = (f(x, \beta_{k+1}) - y)$;

end

Termination : We return the approximate solution $\hat{\beta} = \beta_k$.

Let us denote by $\mathcal{L}$ the level set

$$\mathcal{L} = \{\beta \in \mathbb{R}^m \,:\, F(\beta) \leq F(\beta_0)\},$$

for the initial guess β_0. We state the following convergence theorem.

Theorem 4.4 *Let $\eta \in (0, \frac{1}{4})$ and suppose that the level set $\mathcal{L}$ is bounded. Further suppose that all the component functions of the residual function r are Lipschitz continuously differentiable in a neighborhood U of $\mathcal{L}$. Assume that for each k the search direction p_k satisfies the inequality*

$$m_k(0) - m_k(p_k) \geq c_1 \|J_k^T r_k\|_2 \min\left(\Delta_k, \frac{\|J_k^T r_k\|_2}{\|J_k^T J_k\|_2}\right),$$

for some constant $c_1 > 0$, and in addition $\|p_k\|_2 \leq \gamma \Delta_k$ for some constant $\gamma \geq 1$. We then have that

$$\lim_{k\to\infty} \nabla F(\beta_k) = \lim_{k\to\infty} J_k^T r_k = 0.$$

Proof For the proof we refer to [NW06], Theorem 10.3. □

5 Sinogram Processing and Numerical Results

This chapter will now deal with the Nano CT problem as a whole. While the fundamental principles of tomography mostly remain unchanged, there is a crucial additional consideration that requires attention. At the nano scale, objects possess significant intrinsic kinetic energy, leading to substantial movement during the scanning process—up to 10 percent of their diameter. Addressing this movement is essential during reconstruction, as the random shifts induced during scanning can result in noticeable artifacts in the final images.

Our investigation revolves around the modeling of this translation noise and the exploration of effective techniques for processing CT data, thereby minimizing the impact of translational noise. The data generation process will be discussed, followed by sinogram processing to reduce the prevailing noise. Notably, we will employ the findings from [ERMSS20] for sinogram processing. Finally, numerical results will be presented to showcase our discoveries.

5.1 Data Generation and Sinogram Processing

Nano computed tomography deals with the additional challenge of having to deal with the object moving during the scanning process. Mathematically, we will assume that the density of our object $\mu : \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$ will undergo a rigid motion consisting of a time-dependent translation

$$T : \mathbb{R} \to \mathbb{R}^2.$$

With this, the Radon transform will become dependent on the time τ:

$$\mathcal{R}\mu(\tau, L) := \int_L \mu\left(\begin{pmatrix} x \\ y \end{pmatrix} + T(\tau)\right) \mathrm{d}s(x, y) \quad \forall \tau \in \mathbb{R}.$$

Here we now consider the Radon transform to be defined on Cartesian and not polar coordinates to better match the notation from [ERMSS20]. L is the line that we are integrating over and by $\mathrm{d}s(x, y)$ we are symbolizing the line integral. Instead of introducing Cartesian coordinates, we could instead also require T to be a translation into $[0, 2\pi] \times \mathbb{R}$. Based on the context we will use the approach that simplifies matters more.

In Chapter 2 we parametrized all lines $L \subset \mathbb{R}^2$ via the vectors $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^\perp$ that were given by the formulas

$$\boldsymbol{\theta} = \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix}$$

and

$$\boldsymbol{\theta}^\perp = \begin{pmatrix} -\sin(\theta) \\ \cos(\theta) \end{pmatrix}.$$

We considered the lines $L = L_{\boldsymbol{\theta},s} = \{s\boldsymbol{\theta} + r\boldsymbol{\theta}^\perp : r \in \mathbb{R}\}$. Through the bijection $\theta \mapsto \boldsymbol{\theta}$ on $[0, 2\pi)$ we also have a 1-1 correspondence

$$\theta \longleftrightarrow \boldsymbol{\theta},$$

which will allow us to freely change between referencing θ or $\boldsymbol{\theta}$.

If we want to parametrize lines L with Cartesian coordinates, we have

$$L_{(x,y)} = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} + r \begin{pmatrix} -\sin(\alpha) \\ \cos(\alpha) \end{pmatrix} : r \in \mathbb{R} \right\},$$

where

$$\alpha = \alpha(x,y) = \begin{cases} \arccos\left(\frac{x}{\sqrt{x^2+y^2}}\right) & \text{if } y \geq 0, \\ -\arccos\left(\frac{x}{\sqrt{x^2+y^2}}\right) & \text{if } y < 0. \end{cases}$$

We effectively set

$$s := \sqrt{x^2+y^2}$$

and

$$\boldsymbol{\theta} := \begin{pmatrix} \cos(\alpha(x,y)) \\ \sin(\alpha(x,y)) \end{pmatrix}$$

to then set $L_{(x,y)} := L_{\boldsymbol{\theta},s}$ with the previous definitions.

Our goal is it now to recover the line integrals

$$\int_L \mu\left(\begin{pmatrix} x \\ y \end{pmatrix}\right) \mathrm{d}s(x,y),$$

while only making use of the time dependent versions

$$\mathcal{R}\mu(\tau, L) = \int_L \mu\left(\begin{pmatrix} x \\ y \end{pmatrix} + T(\tau)\right) \mathrm{d}s(x,y).$$

In order to process the sinograms we will follow the ansatz from [ERMSS20] in two dimensions, that consists of realigning the sinogram based on the center of the projections. Before presenting concrete results, let us consider some intuition that will help illustrate the ideas. We will now assume to send rays in the negative y or $-e_2$ direction for a single time frame. Then any translation of our object along the y-axis will actually not change our attenuation profile as we can see in the following figure.

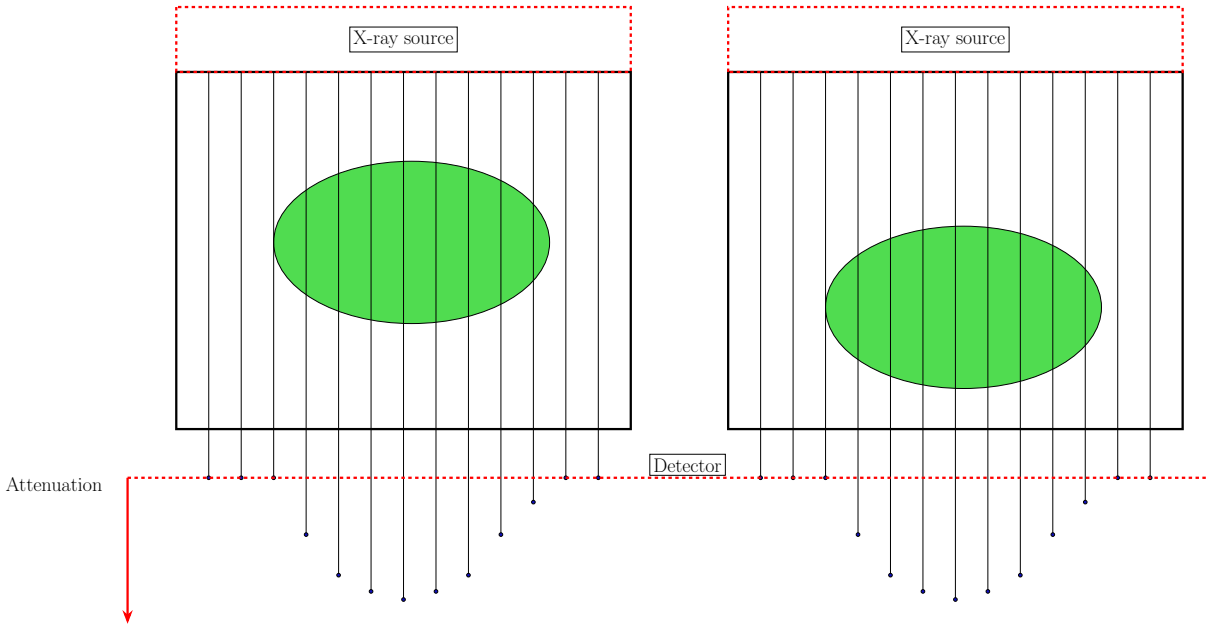


Figure 5: Attenuation of an elliptical object centered (left) and shifted down (right). The attenuation stays the same for any shift in direction of the y-axis.

If the object gets shifted along the x-axis then the attenuation profile changes, but not the shape. Instead we only have a translation of the profile that corresponds to the shift of the object, illustrated by the following figure.

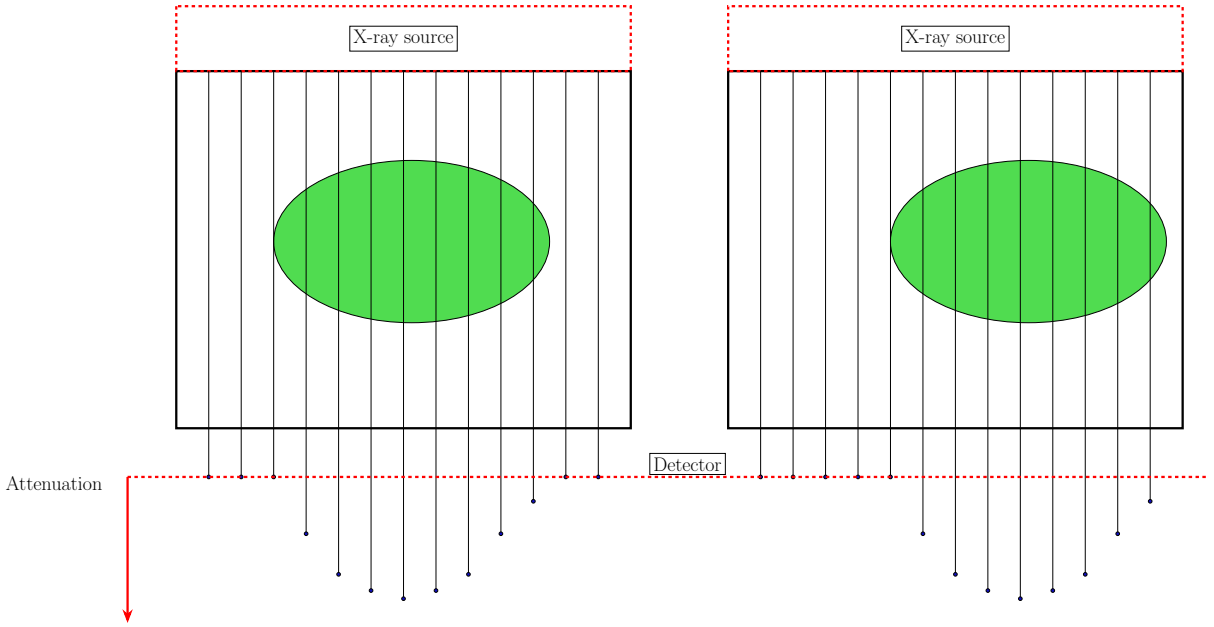


Figure 6: Attenuation of an elliptical object centered (left) and shifted right (right). The attenuation now shifts along the shift in direction of the x-axis.

When we have a translation in x and y direction the shift in y direction will still have no impact on the attenuation. So we will just have to consider the shift along the x-axis, which can be seen below.

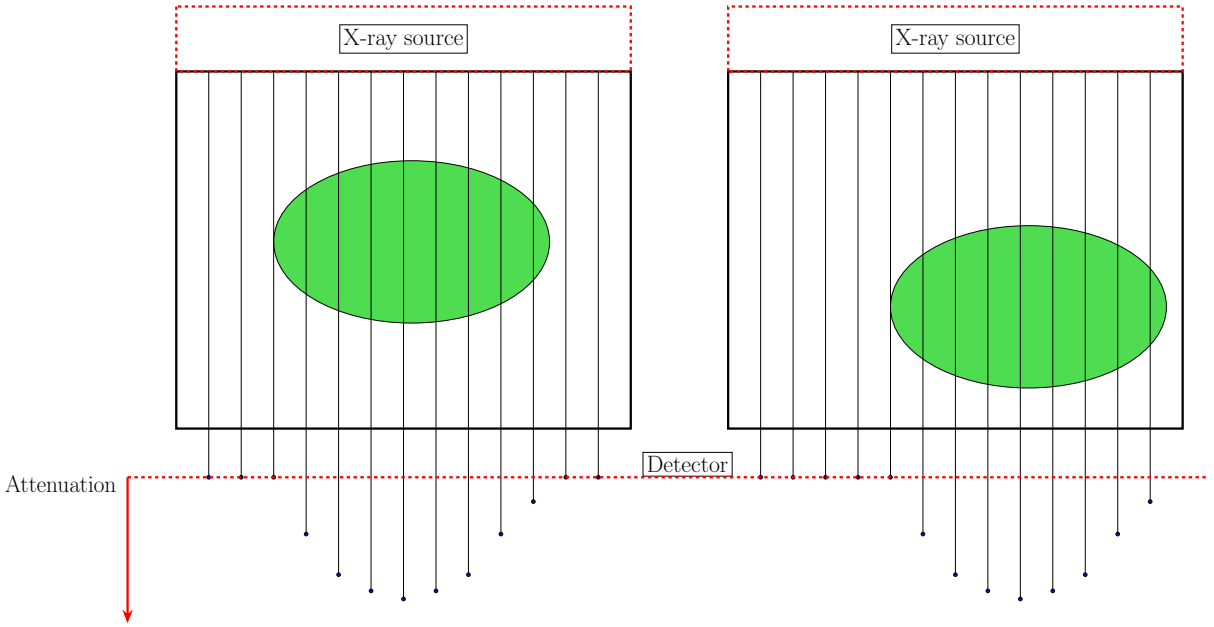


Figure 7: Attenuation of an elliptical object centered (left) and shifted to the right and down (right). The attenuation now shifts only along the shift in direction of the x-axis.

With these considerations in mind the idea is the following. For each angle of the sinogram we only need to consider the translation in the orthogonal direction of its unit vector, the translation in direction of the angle can be ignored. The translation that we want to correct will for fixed angle be the same for all rays. We thus only need to find a way to calculate this translation. For this we make use of the center. The center will shift along with the rest of the object. Not only that, the center of the Radon transformed object actually shifts with the object as well and equals (up to a global translation) the projection of the

center to the angle direction.

Mathematically, we can formulate what we learned in the following way.

Theorem 5.1 *For any $q \in \mathbb{R}$ we have*

$$\mathcal{R}\mu(\tau, L_{\boldsymbol{\theta},s}) = \mathcal{R}\mu(\tau, L_{\boldsymbol{\theta},s} + q\boldsymbol{\theta}^{\perp}) \quad \textit{for all } \tau \in \mathbb{R}.$$

PROOF The proof is fairly simple and consists of substituting $\tilde{r} = r + q$ in the integral:

$$\begin{aligned}
\mathcal{R}\mu(\tau, L_{\boldsymbol{\theta},s} + q\boldsymbol{\theta}^{\perp}) &= \int_{-\infty}^{\infty} \mu\left(s\boldsymbol{\theta} + r\boldsymbol{\theta}^{\perp} + q\boldsymbol{\theta}^{\perp} + T(\tau)\right) \mathrm{d}r \\
&= \int_{-\infty}^{\infty} \mu\left(s\boldsymbol{\theta} + (r+q)\boldsymbol{\theta}^{\perp} + T(\tau)\right) \mathrm{d}r \\
&= \int_{-\infty}^{\infty} \mu\left(s\boldsymbol{\theta} + \tilde{r}\boldsymbol{\theta}^{\perp} + T(\tau)\right) \mathrm{d}\tilde{r} \\
&= \mathcal{R}\mu(\tau, L_{\boldsymbol{\theta},s}) \quad \text{for all } \tau \in \mathbb{R}.
\end{aligned}$$

□

Let us now define the key mathematical concept for our technique, the center.

Definition 5.2 For a density μ we define its center as

$$\mathcal{C}_{\mu}^{2} := \underbrace{\frac{1}{\int_{\mathbb{R}^2} \mu(x,y)\mathrm{d}\lambda^2(x,y)} \int_{\mathbb{R}^2} \begin{pmatrix} x \\ y \end{pmatrix} \mu(x,y)\mathrm{d}\lambda^2(x,y)}_{\in\mathbb{R}^2}.$$

We further define the center of the Radon-transformed object with respect of a certain angle to be

$$\mathcal{C}^1(\tau,\theta) := \frac{1}{\int_{\mathbb{R}} \mathcal{R}\mu(\tau,\boldsymbol{\theta},s)\mathrm{d}\lambda(s)} \int_{\mathbb{R}} s \cdot \mathcal{R}\mu(\tau,\boldsymbol{\theta},s)\mathrm{d}\lambda(s) \quad \text{for all } \tau \in \mathbb{R}$$

and all $\theta \in [0, 2\pi]$.

And now we define the orthogonal projection.

Definition 5.3 For an angle θ we define the orthogonal projection

$$P_{\theta}(s\boldsymbol{\theta} + r\boldsymbol{\theta}^{\perp}) := s.$$

We hence get the following result.

Theorem 5.4 *With the above definitions we have for all $\theta \in [0, 2\pi]$ that*

$$P_{\theta}(\mathcal{C}_{\mu}^{2} - T(\tau)) = \mathcal{C}^1(\tau,\theta) \quad \textit{for all } \tau \in \mathbb{R}.$$

PROOF The proof can be found in [ERMSS20], Proposition 3.3. We will give the same proof here, only with our modified notation and the additional complication of our line integrals having a continuously changing angle θ, where as in [ERMSS20] $\theta \equiv 0$.

To calculate the center $\mathcal{C}^1(\tau,\theta)$, we first remark that with the substitution $\underbrace{v}_{\in\mathbb{R}^2} = s\boldsymbol{\theta} + r\boldsymbol{\theta}^{\perp} + T(\tau)$ we have

$$\int_{\mathbb{R}} \mathcal{R}\mu(\tau,\boldsymbol{\theta},s)\mathrm{d}\lambda(s) = \int_{\mathbb{R}}\int_{\mathbb{R}} \mu\left(s\boldsymbol{\theta} + r\boldsymbol{\theta}^{\perp} + T(\tau)\right) \mathrm{d}\lambda(r)\mathrm{d}\lambda(s) = \int_{\mathbb{R}^2} \mu(v)\mathrm{d}\lambda^2(v).$$

Hence the denominator of $\mathcal{C}^1(\tau,\theta)$ equals just the whole attenuation of the object. In a similar way, we

find

$$\begin{aligned}
\int_{\mathbb{R}} s \cdot \mathcal{R}\mu(\tau, \boldsymbol{\theta}, s)\mathrm{d}\lambda(s) &= \int_{\mathbb{R}}\int_{\mathbb{R}} s \cdot \mu\left(s\boldsymbol{\theta} + r\boldsymbol{\theta}^{\perp} + T(\tau)\right)\mathrm{d}\lambda(r)\mathrm{d}\lambda(s) \\
&= \int_{\mathbb{R}}\int_{\mathbb{R}} P_\theta(s\boldsymbol{\theta} + r\boldsymbol{\theta}^{\perp}) \cdot \mu\left(s\boldsymbol{\theta} + r\boldsymbol{\theta}^{\perp} + T(\tau)\right)\mathrm{d}\lambda(r)\mathrm{d}\lambda(s) \\
&= \int_{\mathbb{R}^2} P_\theta(v - T(\tau)) \cdot \mu(v)\mathrm{d}\lambda^2(v) \\
&= P_\theta\left(\int_{\mathbb{R}^2} v \cdot \mu(v)\mathrm{d}\lambda^2(v) - T(\tau) \cdot \int_{\mathbb{R}^2} \mu(v)\mathrm{d}\lambda^2(v)\right) \\
&= P_\theta(\mathcal{C}^2_\mu - T(\tau)) \cdot \int_{\mathbb{R}^2} \mu(v)\mathrm{d}\lambda^2(v).
\end{aligned}$$

Dividing by $\int_{\mathbb{R}^2} \mu(v)\mathrm{d}\lambda^2(v) = \int_{\mathbb{R}} \mathcal{R}\mu(\tau, \boldsymbol{\theta}, s)\mathrm{d}\lambda(s)$ gives the equation for all θ and all τ. □

The idea of the sinogram processing is to calculate $\mathcal{C}^1(\tau, \theta)$ from the data and to then have an estimate of the object translation via Theorem 5.4. We know the translation in direction of $\boldsymbol{\theta}^{\perp}$ will not change the Radon transform, hence we do not need to correct for it. We only correct for motion in the direction of $\boldsymbol{\theta}$ and Theorem 5.4 gives us the amount we need to correct for each time step.

Now for the actual data that we will use, we again turn to the discretized model. We remember from Chapter 2 that our data consists of a vector $g \in \mathbb{R}^k$ that contains all the measured line integrals. More concretely, we assumed supp $\mu \subset [-1, 1]^2$. For the parallel beam geometry we then choose equally spaced $s_1, \ldots, s_r \in [-1, 1]$ and $\theta_1, \ldots, \theta_t \in [0, \pi]$ with $k = r \cdot t$ and get

$$g = (g_{s_1\theta_1}, \ldots, g_{s_r\theta_1}, g_{s_1\theta_2}, \ldots, g_{s_r\theta_2}, \ldots, g_{s_1\theta_t}, \ldots, g_{s_r\theta_t})^T$$

the vectorized sinogram, where $g_{s_i\theta_j} = \mathcal{R}\mu(\boldsymbol{\theta}_j, s_i)$.

The **sinogram** is then the data grouped as a matrix:

$$\mathfrak{g} = \left(g_{s_i\theta_j}\right)_{\substack{i=1,\ldots,r \\ j=1,\ldots,t}} = \left(\mathcal{R}\mu(\boldsymbol{\theta}_j, s_i)\right)_{\substack{i=1,\ldots,r \\ j=1,\ldots,t}}.$$

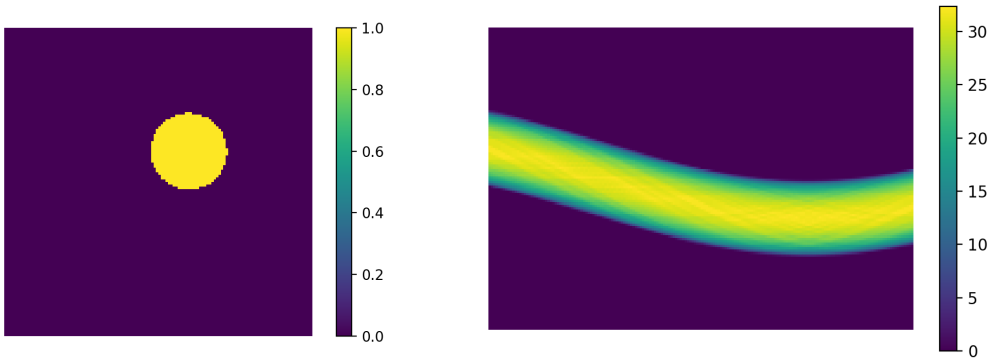


Figure 8: Here is an example of a circle and its sinogram. The sinusoidal shape of Radon transformed points and circles gives the sinogram its name.

We know from Theorem 5.1 that we only have to consider the translation in direction of $\boldsymbol{\theta}$. Because of the scanning geometry that we simulate we can assume that for a fixed angle θ_j we will measure all the column values $\mathfrak{g}_{\cdot j}$ at the same time τ_j. Because of this, we will add the translation noise to our sinograms in the following way:

$$\mathfrak{g}_{ij} \longrightarrow \mathfrak{g}_{i+\lfloor\varepsilon_j\rceil, j} := \mathfrak{g}'_{ij} \quad \text{for } \varepsilon_j \sim \mathcal{N}(0, \sigma^2) \text{ and all } i = 1, \ldots, r.$$

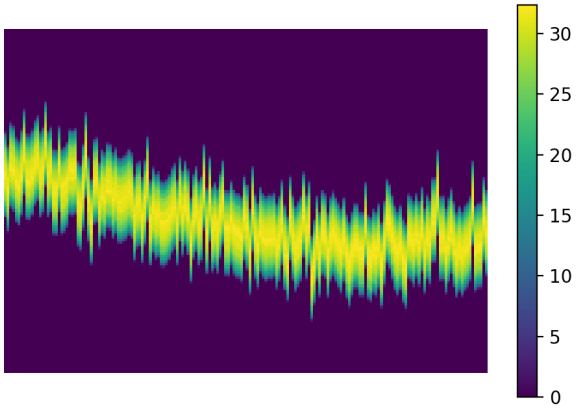


Figure 9: This sinogram of a circle contains the random shifts caused by translational noise.

We will now try to leverage Theorem 5.4 to reconstruct $\mathfrak{g}$ from the noisy data $\mathfrak{g}'$. When discretizing the center we get that

$$\mathcal{C}^1(\tau,\theta) = \frac{1}{\int_{\mathbb{R}} \mathcal{R}\mu(\tau,\boldsymbol{\theta},s)\mathrm{d}\lambda(s)} \int_{\mathbb{R}} s \cdot \mathcal{R}\mu(\tau,\boldsymbol{\theta},s)\mathrm{d}\lambda(s) \approx \frac{1}{\sum_{i=1}^{r} g_{s_i\theta}} \sum_{i=1}^{r} s_i \cdot g_{s_i\theta}.$$

For our data $\mathfrak{g}'$ we then calculate the center for all angles θ_j, resulting in a curve

$$y = (y_j)_{j=1,\dots,t} \approx \left(\mathcal{C}^1(\tau_j,\theta_j)\right)_{j=1,\dots,t}$$

and data $(x,y) = (j,y_j)_{j=1,\dots,t}$.

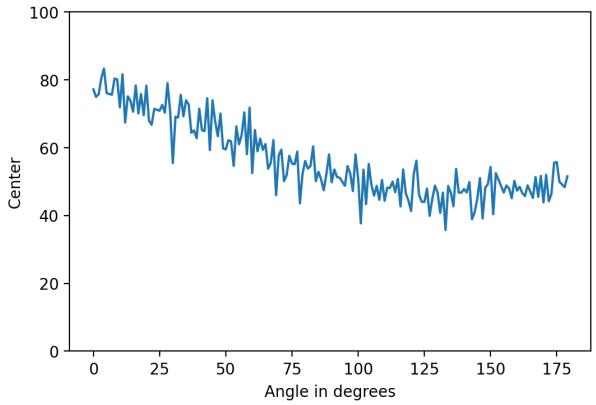


Figure 10: This is the center of the noisy sinogram.

Using the Levenberg–Marquardt Algorithm from Chapter 4 we now fit this curve to a polynomial of a specified degree N to get $(j,p_N(j))_{j=1,\dots,t}$ as the corrected center curve. In the numerical examples we will use $N = 3$ to prevent overfitting, as we will focus on simple objects for which the center will be a more stationary curve.

We will then shift back $\mathfrak{g}'_{\cdot j}$ by the difference of the two curves:

$$\mathfrak{g}^*_{\cdot j} := \mathfrak{g}'_{\cdot + \lfloor y_j - p_N(j) \rceil, j} \quad \text{for all } j = 1,\dots,t,$$

or element-wise

$$\mathfrak{g}^*_{ij} = \mathfrak{g}'_{i+\lfloor y_j - p_N(j) \rceil, j} \quad \text{for all } i = 1,\dots,r \text{ and all } j = 1,\dots,t.$$

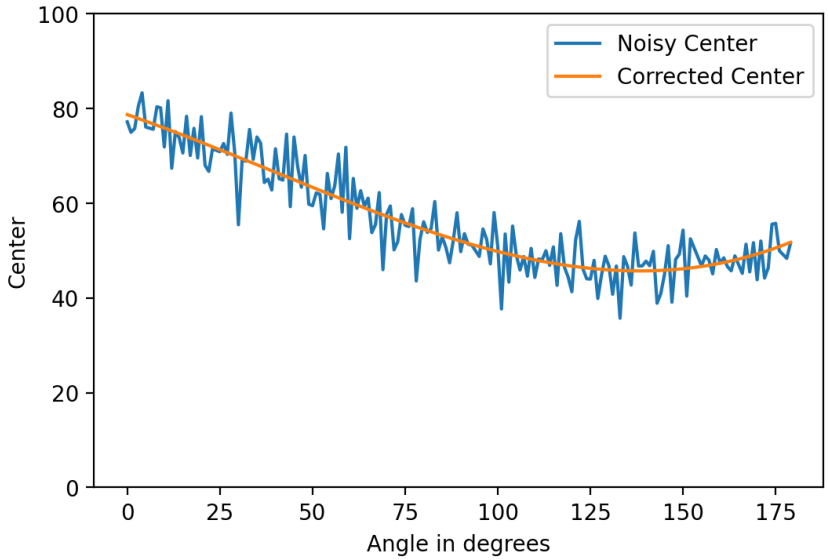


Figure 11: Corrected center of the noisy sinogram

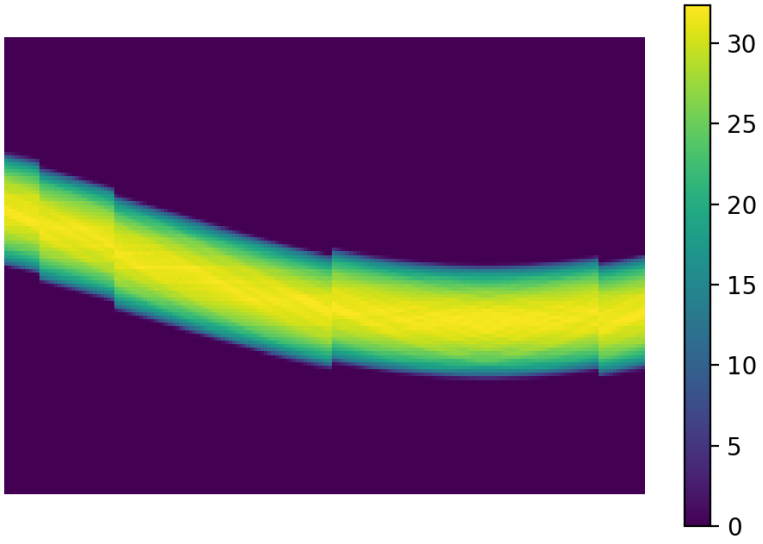


Figure 12: Processed sinogram with multiple jumps or segments. This is a less than ideal instance of the reconstruction where the jumps will cause artefacts in the reconstructed image.

5.2 Results

We will now discuss numerical results of our sinogram processing and our reconstruction. We will first discuss qualitatively and then quantitatively. For our quantitative analysis we will generate 30 examples each of different objects and then compare MSE (mean squared error) and EMD (earth mover's distance) averages of the noisy and processed sinograms.

We will elucidate the reconstruction process based solving the discretized inverse problem with regularization parameter obtained by the formula in Chapter 3. We will again compare average MSE and EMD, this time for our algorithm and for the implemented method of the Python library SciPy.

5.2.1 Results of the Sinogram Processing

Throughout our analysis we will concentrate on simple objects. Namely, we are interested in a circle, a rectangle, and two circles.

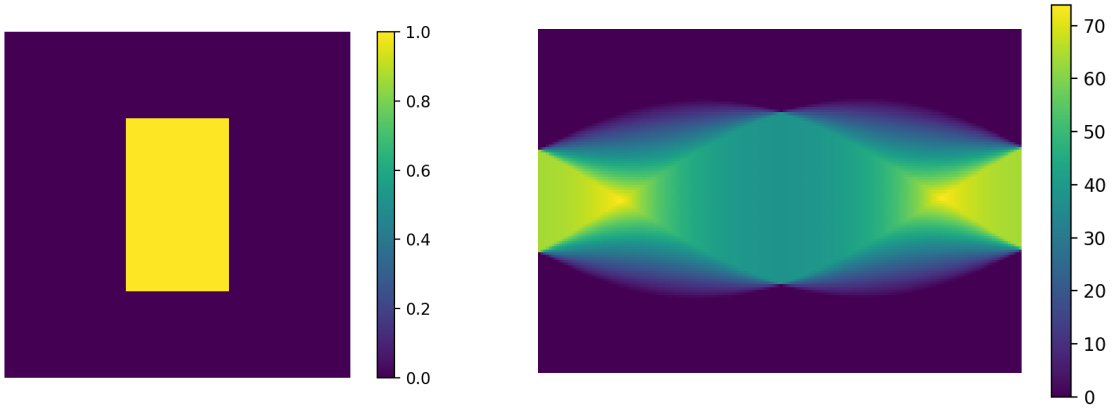


Figure 13: Here is an example of a rectangle and its sinogram.

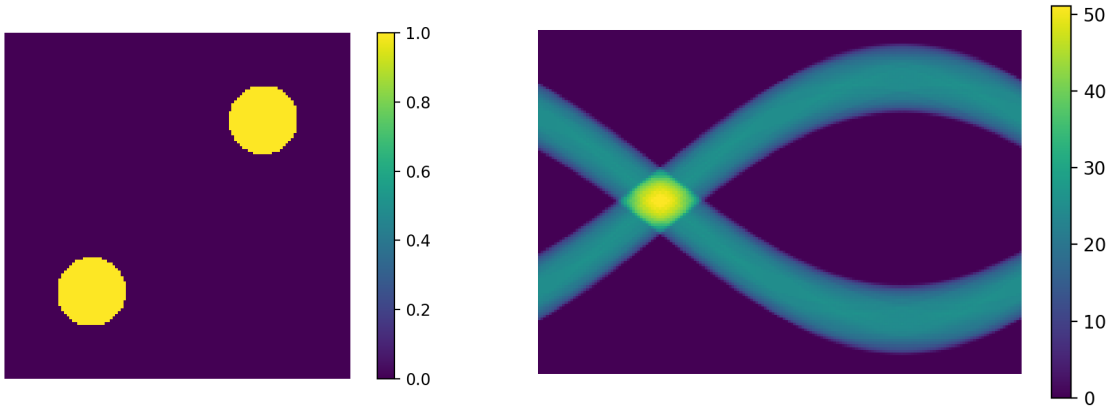


Figure 14: Here is an object consisting of two circles and its sinogram.

We will use objects that are represented by 128×128 pixel images and the sinograms will have dimensions 128×180, hence 180 angles with 128 rays each. Qualitatively, there are a few effects that can be seen during the reconstruction, which we are going to discuss now.

In some instances, we will be able to reconstruct the sinogram $\mathfrak{g}$ perfectly.

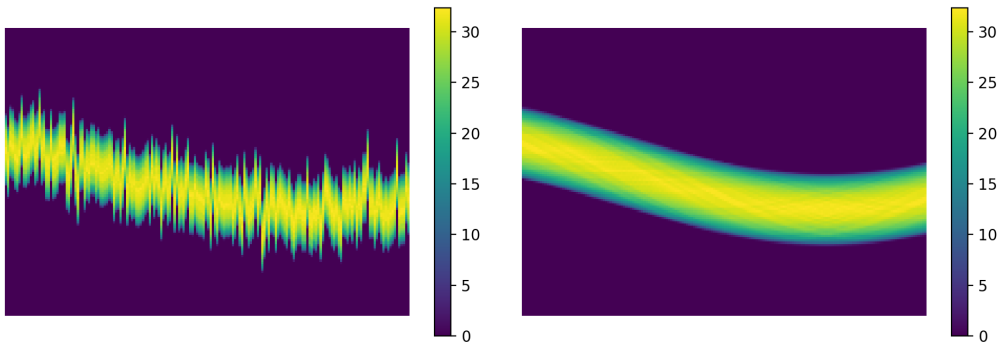


Figure 15: Noisy sinogram and perfectly processed sinogram

Another common occurrence is that almost all lines of the sinogram are reconstructed well, except for

a small portion of them. This happened most profoundly in the two circle sinogram, but almost never in the sinogram of the rectangle. The reason for this would need to be investigated further.

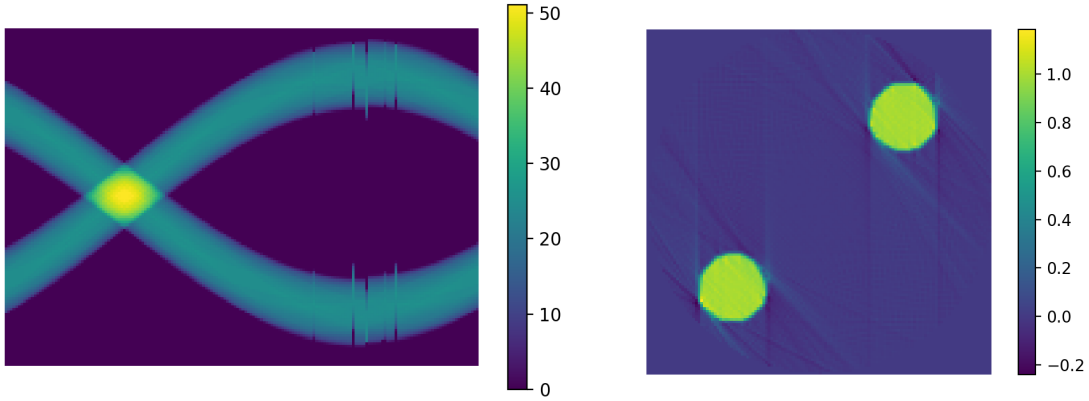


Figure 16: An almost perfectly processed sinogram where only five columns are not realigned and the resulting reconstructed object. We see that even small mistakes in the processed sinogram will cause artifacts in the reconstruction.

Another effect is that the processed sinogram recovers the original shape but with a shift upwards or downwards. This occurs frequently for the single circle and for the rectangle.

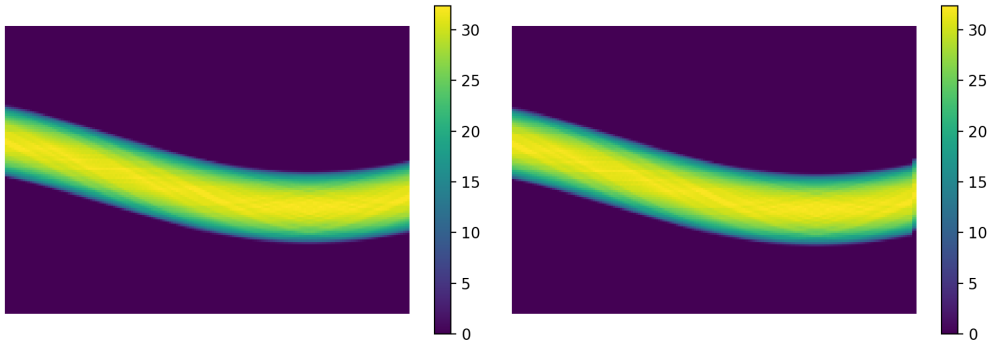


Figure 17: Normal sinogram and reconstructed sinogram that has a shift down. Based on the size of the shift, the reconstruction can still be successful. Correcting for shifts would require to either know the starting position of the object or to implement some kind of sparsity requirement into the reconstruction, as the unshifted sinogram usually have the most sparse reconstructions.

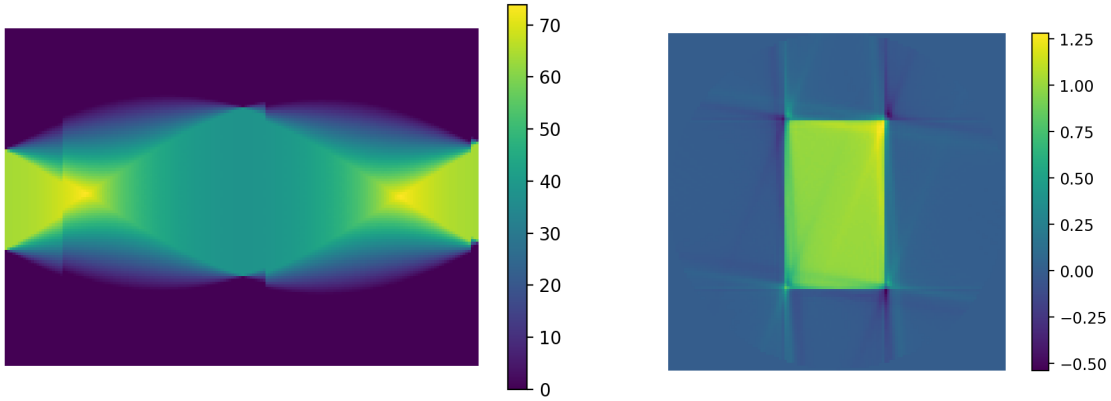


Figure 18: Processed sinogram of a rectangle with four visible segments and its reconstruction. There are noticeable artifacts in the reconstruction caused by the jumps.

The most common occurrence is that certain segments of the sinogram are reconstructed perfectly, but others are shifted either upwards or downwards by a couple of pixels. These effects are usually mitigated with increasing the number of pixels and could be due to numerical approximations in the processing procedure. The exact reason would need to be investigated further. This will impact the reconstruction significantly, depending on the size of the segment and the size of the shift.

Let us now look at the qualitative results of our sinogram processing. Our objects are defined on $[-1,1]^2$. The circle has center $(0.2, 0.2)^T$ and radius 0.25, overall we have

$$\mu_c(x,y) := \begin{cases} 1 & \text{if } \sqrt{(x-0.2)^2+(y-0.2)^2} \leq 0.25 \\ 0 & \text{else.} \end{cases}$$

The rectangle is centered at the origin with side lengths 0.6 and 1, i.e.

$$\mu_r(x,y) := \begin{cases} 1 & \text{if } |x| \leq 0.3 \text{ and } |y| \leq 0.5 \\ 0 & \text{else.} \end{cases}$$

For the two circles we define a density of

$$\mu_{2c}(x,y) := \begin{cases} 1 & \text{if } \sqrt{(x-0.5)^2+(y-0.5)^2} \leq 0.2 \text{ or } \sqrt{(x+0.5)^2+(y+0.5)^2} \leq 0.2, \\ 0 & \text{else.} \end{cases}$$

We present our results in a table. For the shift of the columns of the sinograms we decided on

$$\varepsilon_i \sim \mathcal{N}(0, 0.078),$$

such that the variance is about five pixel per column in the images.

The reconstruction was able to lower errors for all objects and all metrics, where we have taken the average error for a sample of 30 instances of the objects each. Specifically, we are looking at

$$\text{MSE}(\mathfrak{g}, \mathfrak{g}^*) \text{ and } \text{MSE}(\mathfrak{g}, \mathfrak{g}'),$$

where $\mathfrak{g}$ was the ground truth sinogram, $\mathfrak{g}^*$ was the reconstructed sinogram and $\mathfrak{g}'$ was the sinogram that contained the random shifts. Here we mean

$$\text{MSE}(\mathfrak{g}, \mathfrak{g}^*) := \sum_{i,j} (\mathfrak{g}_{ij} - \mathfrak{g}^*_{ij})^2$$

which is the standard definition for the mean squared error.

We also look at

$$\text{EMD}(\mathfrak{g}, \mathfrak{g}^*) \text{ and } \text{EMD}(\mathfrak{g}, \mathfrak{g}'),$$

where we define the earth mover's distance in the following way, see [LO07] for more details.

A signature of size k is defined as a set $S = \{s_l = (w_l, m_l)\}_{l=1}^k$, where m_l represents the position of the l-th element, and w_l is its weight. In our context, the positions m_l correspond to the pixel values (a, b), and the weights are denoted by $w_l := \mathfrak{g}_{ab}$.

When dealing with two signatures $P = \{(p_i, u_i)\}_{i=1}^k$ and $Q = \{(q_j, v_j)\}_{j=1}^k$ of size k for images $\mathfrak{g}$ and $\mathfrak{g}^*$, respectively, the EMD between them can be formulated as a solution to a transportation problem. Here, the elements in P are treated as "supplies" located at u_i, while elements in Q represent "demands" situated at v_j. The quantities p_i and q_j indicate the amount of supply and demand, respectively, and

$d_{ij} := d_{\ell^2}(u_i, v_j) = \|u_i - v_j\|_2$ represents the ground distance between u_i and v_j.

The EMD is defined as the minimum (normalized) work required to accomplish the supply-demand transportation, given by the expression:

$$\mathrm{EMD}(\mathfrak{g}, \mathfrak{g}^*) := \mathrm{EMD}(P, Q) := \min_{F=\{f_{ij}\}} \frac{\sum_{i,j} f_{ij} d_{ij}}{\sum_{i,j} f_{ij}},$$

subject to the following constraints:

$$\begin{aligned}
&\sum_{j=1}^{k} f_{ij} \leq p_i, \\
&\sum_{i=1}^{k} f_{ij} \leq q_j, \\
&\sum_{i,j} f_{ij} = \min\left\{ \sum_{i=1}^{k} p_i, \sum_{j=1}^{k} q_j \right\}, \\
&f_{ij} \geq 0.
\end{aligned}$$

In short summary, the EMD quantifies the dissimilarity between two images $\mathfrak{g}$ and $\mathfrak{g}^*$ by determining the optimal way to transport mass from one image to the other while minimizing the overall transportation cost. In its discrete form it is a linear program for which solutions exist in Python.

Now we can look at the table results, where the processing was able to lower errors for all objects and all metrics.

Shape	MSE ($\times 10^{-5}$)		EMD	
	Reconstructed	Noisy	Reconstructed	Noisy
Circle	0.369	5.839	4.629	7.956
Rectangle	0.865	13.118	10.550	12.598
Two Circles	0.814	8.366	21.102	26.570

5.2.2 Results of the Reconstruction

Let us now present numerical results of our reconstruction process. To reconstruct the objects, we will not work with the sinogram $\mathfrak{g}$ but instead group the data back into the vector $g =: y$. We then aim at solving

$$(A^T A + \lambda I_m)x = y,$$

where A was defined in Chapter 2. We solve this equation with the CG method, as the matrix $(A^T A + \lambda I_m)$ is positive definite for all $\lambda > 0$, as already discussed. The optimal λ gets picked by choosing the λ such that the function (16)

$$-2 \ln f_{Y|\Lambda=\lambda}(y) = \|A\hat{x}_\lambda - y\|_2^2 + \lambda \|\hat{x}_\lambda\|_2^2 + n \ln 2\pi + \ln \det\left(\frac{A^T A}{\lambda} + I_m\right)$$

from Chapter 3 is minimal.

Note that in order to evaluate the function (16) we need to minimize the Tikhonov functional. Hence for choosing the optimal λ we need to reconstruct the image in every step, only with a different regularization parameter. To lower computation costs we will lower the dimensions of the sinograms and images to 64×90 for the sinograms and 64×64 for the images. We only use a single sinogram, the ground truth sinogram, for the comparison.

We will compare our algorithm to the 'iradon' function from SciPy with the default parameters. The function 'iradon' uses the filtered backprojection with a ramp filter as its method. The complexity of $-2\ln f_{Y|\Lambda=\lambda}(y)$ limits us to only perform a grid search, from which we will pick the parameter.

λ	$-2\ln f_{Y\|\Lambda=\lambda}(y)$	MSE	EMD
'iradon'	–	5.087	5.881
$\lambda_1 = 0.01$	2087.400	2.140	26.267
$\lambda_2 = 0.1$	1650.111	2.219	26.638
$\lambda_3 = 0.3$	1476.999	2.148	23.987
$\lambda_4 = 0.5$	1424.070	2.220	22.210
$\lambda_5 = 1$	1405.368	2.232	29.200
$\lambda_6 = 1.5$	1446.192	2.378	25.611
$\lambda_7 = 2$	1507.979	2.401	25.616
$\lambda_8 = 3$	1664.699	2.454	22.730
$\lambda_9 = 5$	2036.861	2.696	29.890
$\lambda_{10} = 10$	3063.066	3.083	28.066

Table 1: Comparing different function values for the circle. The MSE and EMD here are of the reconstructed object to the ground truth circle image.

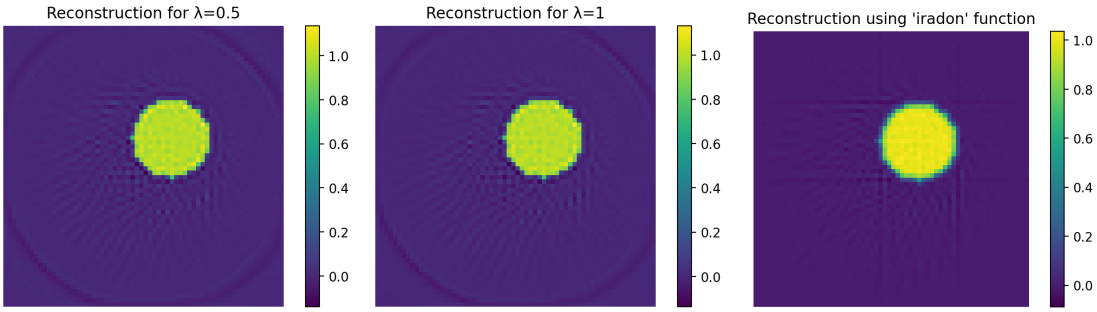


Figure 19: Reconstruction for $\lambda = 0.5$, $\lambda = 1$, and for the 'iradon' function. At least visually all three reconstructions seem to be decent as we can clearly make out the shape of a circle.

Looking at the data, we see that the minimum of $-2\ln f_{Y|\Lambda=\lambda}(y)$ is attained for $\lambda = 1$. This agrees with the minimal MSE, which is also lower than the MSE of the 'iradon' reconstructed sinogram. The minimum for the EMD is attained for $\lambda = 0.5$ but this maximum is still much larger than the EMD of the 'iradon' reconstructed sinogram.

Let us now turn to the rectangle.

λ	$-2\ln f_{Y\|\Lambda=\lambda}(y)$	MSE	EMD
'iradon'	–	2.700	2.376
$\lambda_1 = 0.01$	2113.204	2.292	13.016
$\lambda_2 = 0.1$	1704.254	2.296	13.372
$\lambda_3 = 0.3$	1598.948	2.304	12.729
$\lambda_4 = 0.5$	1611.152	2.312	11.395
$\lambda_5 = 1$	1758.124	2.336	13.176
$\lambda_6 = 1.5$	1962.192	2.362	12.683
$\lambda_7 = 2$	2189.350	2.390	13.718
$\lambda_8 = 3$	2676.345	2.455	13.188
$\lambda_9 = 5$	3717.030	2.999	13.813
$\lambda_{10} = 10$	6373.919	3.499	13.554

Table 2: Comparing different function values for the rectangle

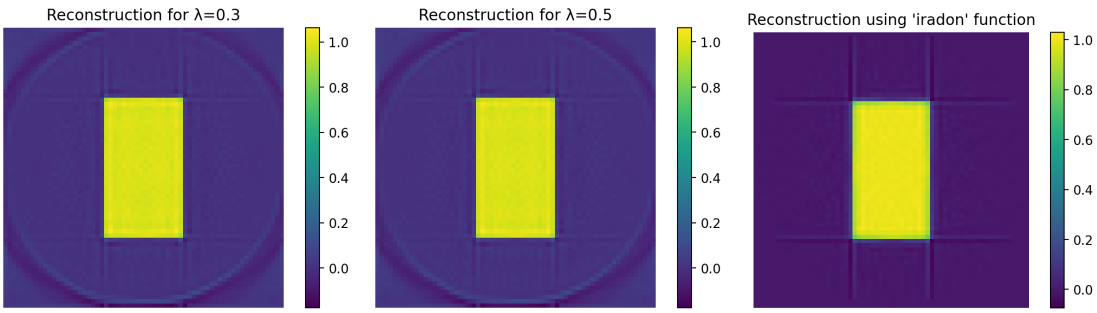


Figure 20: Three reconstructions of the object

Here we see that the minimum of $-2\ln f_{Y|\Lambda=\lambda}(y)$ is attained for $\lambda = 0.3$, while the minimal MSE is attained for $\lambda = 0.01$ and the minimal EMD for $\lambda = 0.5$. The MSE for $\lambda = 0.3$ is at least smaller than the MSE of the 'iradon' reconstruction, but the EMD is much lower for the 'iradon' reconstruction.

Lastly, let us look at the two circles.

λ	$-2\ln f_{Y\|\Lambda=\lambda}(y)$	MSE	EMD
'iradon'	–	7.422	11.640
$\lambda_1 = 0.01$	2089.808	3.481	39.201
$\lambda_2 = 0.1$	1651.481	3.485	36.218
$\lambda_3 = 0.3$	1482.552	3.599	37.545
$\lambda_4 = 0.5$	1429.727	3.624	36.543
$\lambda_5 = 1$	1413.550	3.650	37.355
$\lambda_6 = 1.5$	1452.950	3.560	39.079
$\lambda_7 = 2$	1518.506	3.698	39.747
$\lambda_8 = 3$	1680.006	3.789	35.792
$\lambda_9 = 5$	2058.665	3.978	36.890
$\lambda_{10} = 10$	3101.060	4.670	38.356

Table 3: Comparing different function values for the two circles

The minimum of $-2\ln f_{Y|\Lambda=\lambda}(y)$ is attained for $\lambda = 1$, the minimal MSE is attained for $\lambda = 1.5$ and the minimal EMD for $\lambda = 3$. We have a similar effect to the rectangle where for all λ the MSE is lower than the MSE of the 'iradon' reconstruction while the EMD is significantly larger.

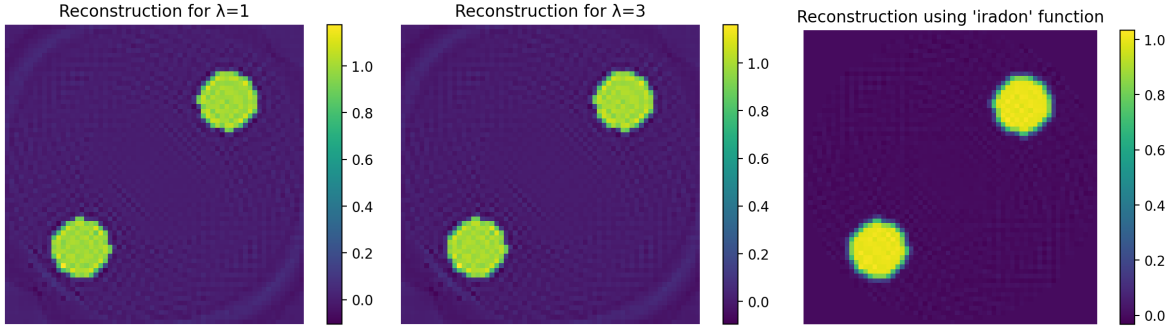


Figure 21: Reconstruction for $\lambda = 1$, $\lambda = 3$, and for the 'iradon' function. All three clearly display the two circles, however the 'iradon' reconstruction has the highest constrast between the objects and the background (looking at the colorbar).

The original derivation of the used formula encompasses a noisy measurement, hence we will now add some Gaussian noise to the sinograms to see, if our algorithm then outperforms the SciPy variant of the

reconstruction. Specifically, we will add noise in the following way:

$$y = Ax + \nu,$$

where $N = \nu$ is a realization of $N \sim \mathcal{N}(0, I_n)$.

We will further lower the dimension of the objects to 25×25 and to 25×30 for the sinograms. Since we are adding random noise to the images, we will not just compare single images, but instead take a sample of 10 pictures and compare averages. Defining $-2 \ln f_{Y|\Lambda=\lambda}(y) =: \phi(\lambda)$, we get the following table:

	Circle			Rectangle			Two Circles		
λ	$\phi(\lambda)$	MSE	EMD	$\phi(\lambda)$	MSE	EMD	$\phi(\lambda)$	MSE	EMD
'iradon'	–	5.087	5.881	–	7.422	11.640	–	9.048	12.408
$\lambda_1 = 0.3$	1213.243	121.514	65.809	1223.309	109.548	26.808	1224.876	118.453	40.223
$\lambda_2 = 1$	1050.277	51.623	54.844	1089.808	48.042	20.726	1066.091	49.640	35.238
$\lambda_3 = 1.5$	1009.550	39.641	52.423	1070.157	37.402	18.445	1028.693	38.585	31.919
$\lambda_4 = 2$	987.591	33.058	49.050	1069.654	31.427	17.183	1010.266	32.431	30.359
$\lambda_5 = 3$	969.620	25.703	46.089	1094.411	24.588	15.776	999.402	25.467	28.408
$\lambda_6 = 5$	975.920	18.531	41.543	1184.944	17.763	13.667	1019.388	18.740	24.930
$\lambda_7 = 10$	1057.621	11.654	35.069	1468.078	11.453	10.756	1131.560	12.430	21.837
$\lambda_8 = 20$	1258.319	7.781	28.944	2040.344	8.511	8.718	1381.083	9.407	19.524
$\lambda_9 = 50$	1769.851	7.254	21.191	3492.539	10.543	6.860	1987.215	10.534	15.708
$\lambda_{10} = 100$	2358.295	9.916	18.441	5290.044	16.713	6.786	2659.888	14.807	15.179

Table 4: Comparing different function values for the circle, rectangle, and two circles

Not once we have that the minimizer of ϕ beats the 'iradon' reconstruction. For the circle no regularization actually beats the 'iradon' reconstruction, neither in terms of MSE nor in terms of EMD. For the rectangle at least a couple regularizations beat the 'iradon' reconstruction in one of the two metrics. For the two circles we again have no regularization beating the 'iradon' reconstruction, but at least a couple come close. However these are not the ones that would be chosen by minimizing ϕ.

So while our Bayesian approach gives us an objective criterion to choose the regularization parameter, it does not perform optimally for our examples and our performance metrics.

Conclusion and Outlook In this work, we addressed the processing challenges in Nano Computed Tomography caused by inherent kinetic energy, which leads to significant noise in the sinogram data. By employing Bayesian statistics, we implemented a method for selecting the regularization parameter λ and used the Tikhonov regularization with the Conjugate Gradient method for image reconstruction. Our approach successfully mitigates noise and improves image quality although we were unable to outcompete existing Python code.

There are many different approaches and ideas that could be followed to improve results. One could look at different Bayesian estimators than the MAP to see if they would provide better estimators. One could also change the noise modelling, leading to a different Bayesian integral and different formula for λ.

It is also possible to not only consider translation noise but also a rotating object, for more information we refer to [ERMSS20].

Looking at the applied side, it is needed to further investigate the jumps that were found in some of the processed sinograms. They could be possibly related to the low dimensions of the objects, but this needs further exploration.

Improvements in the code that implements the algorithms are also further possible. With more efficient code we could have also improved upon the grid search that we performed here to possibly select a λ with even less error.

6 References

[DH19] Peter Deuflhard and Andreas Hohmann. *Numerische Mathematik 1.* De Gruyter, Berlin, Boston, 2019.

[Dur19] Rick Durrett. *Probability: Theory and Examples.* Cambridge Series in Statistical and Probabilistic Mathematics, Cambridge, May 2019.

[ERMSS20] Peter Elbau, Monika Ritsch-Marte, Otmar Scherzer, and Denise Schmutz. Motion reconstruction for optical tomography of trapped objects. *Inverse Problems*, 36(4):044004, 2020.

[HF16] Naor Huri and Meir Feder. Selecting the lasso regularization parameter via bayesian principles. *2016 IEEE International Conference on the Science of Electrical Engineering (ICSEE)*, pages 1–5, 2016.

[Kel76] Joseph B. Keller. Inverse problems. *The American Mathematical Monthly*, 83(2):107–118, 1976.

[LO07] Haibin Ling and Kazunori Okada. An efficient earth mover's distance algorithm for robust histogram comparison. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):840–853, 2007.

[MNP19] François Monard, Richard Nickl, and Gabriel P. Paternain. Efficient nonparametric bayesian inference for x-ray transforms. *The Annals of Statistics*, 47(2):1113–1147, 2019.

[Nat86] Frank Natterer. *The Mathematics of Computerized Tomography.* Vieweg+Teubner Verlag Wiesbaden, 1 edition, 1986.

[NW06] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization.* Springer Series in Operations Research and Financial Engineering. Springer New York, NY, 2 edition, 2006.

[Rad17] Johann Radon. Über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiten. *Berichte über die Verhandlungen der Königlich-Sächsischen Akademie der Wissenschaften zu Leipzig. Mathematisch-Physische Klasse*, 69:262–277, 1917.

[SS18] Jakub Sevcik and Vaclav Smidl. A comparison of sparse bayesian regularization methods on computed tomography reconstruction. *Journal of Physics: Conference Series*, 1047(1):012013, 2018.

[Wer92] Jochen Werner. *Numerische Mathematik.* Vieweg+Teubner Verlag Wiesbaden, 1 edition, 1992.