

Fast numerical inversion of the attenuated Radon transform with full and partial measurements

Guillaume Bal and Philippe Moireau

Department of Applied Physics and Applied Mathematics, Columbia University, New York, NY 10027, USA

E-mail: gb2030@columbia.edu and philippe.moireau@polytechnique.org

Received 20 November 2003, in final form 24 March 2004

Published 21 May 2004

Online at stacks.iop.org/IP/20/1137

DOI: 10.1088/0266-5611/20/4/009

Abstract

We propose a numerical method to simulate and invert the two-dimensional attenuated Radon transform (AtRT) from full (360°) or partial (180°) measurements. The method is based on an extension of the fast slant stack algorithm developed for the Radon transform. We show that the algorithm offers robust and fast inversion of the AtRT for a wide class of synthetic sources and absorptions. The complexity of the fast algorithm to compute the AtRT of a $n \times n$ image and perform the reconstruction from the AtRT data is $O(Nn^2 \log n)$ operations, with N the number of Fourier modes necessary to accurately represent the absorption map. The algorithm is applied to the reconstruction of the exponential Radon transform, where the absorption coefficient is constant, and of the AtRT when only 180° measurements are available. The reconstruction from partial measurements is based on an iterative scheme introduced recently in Bal (2004 *Inverse Problems* **20** 399–419). Single-photon emission computed tomography is an important medical imaging technique based on the inversion of the AtRT.

1. Introduction

SPECT (single-photon emission computed tomography) is a very popular medical imaging technique to obtain pictures of the brain, the lungs, the liver and the heart for instance. It is based on the detection of gamma particles emitted from injected radioactive atoms. Mathematically, the imaging technique involves reconstructing the spatial structure of the source of gamma particles from boundary measurements. Whereas scattering of the particles can be neglected as a first approximation (for scattering is mostly inelastic so that reemitted particles have a lower energy), absorption of the gamma particles from the source term to the detectors cannot be ignored.

In the absence of absorption, the imaging problem consists of reconstructing a source term from its Radon transform. Inversion formulae and numerical techniques to solve this problem are well known [22, 23, 30]. In the presence of absorption, a recent explicit formula [24, 27, 28] also allows us to reconstruct the source term from its *attenuated Radon transform* (AtRT), the boundary measurements; we also refer to [1, 10] for an alternative approach. The reconstruction is based for two-dimensional problems on measuring the particle density on the whole boundary of a domain in which the support of the source term is included. In practice however one would like to reconstruct the source term from fewer data.

The Radon transform has an obvious redundancy of order two in the sense that it satisfies $g(s, \theta) = g(-s, \theta + \pi)$; see (1). This redundancy is no longer explicit for the attenuated Radon transform. It was shown in [26, 29] that half of the measurements (180° measurements) are sufficient to reconstruct the source term when the absorption is constant. The iterative procedure proposed there generalizes the reconstruction formulae from full measurements derived in [32] for the exponential Radon transform (ERT; the AtRT for a constant absorption map). More recently, it was shown in [6] that 180° measurements are also sufficient to reconstruct compactly supported source terms from their AtRT provided that the absorption satisfies certain smallness conditions. This paper addresses the numerical aspects of the latter reconstruction theory and shows that reconstruction from half measurements is indeed feasible in cases of practical interest.

The numerical technique we consider here is a generalization of the fast slant stack algorithm as it was described in [2, 3] to compute and invert the Radon transform. The slant stack algorithm is also referred to as the linogram algorithm in the literature and has a long history in medical imaging; see [4, 11, 25]. A crucial property of that algorithm is that it offers an exactly invertible discrete Radon transform. Moreover this algorithm is fast (its complexity is in $O(n^2 \log n)$ to calculate and invert the Radon transform of an $n \times n$ image) and accurate (the Radon transform is calculated with spectral accuracy and its inversion can be accurately performed relatively easily). The fast slant stack algorithm uses a discrete version of the Fourier slice theorem, which relates the Fourier transform of the source term to the Fourier transform of the Radon transform. This important feature of the Radon transform no longer holds for the AtRT. We show that by decomposing exponential terms in the ERT and AtRT we can also obtain algorithms that perform the inversion with a complexity of order $O(Nn^2 \log n)$, where the constant N is the number of Fourier modes necessary to accurately represent the absorption map. For the AtRT, this requires preprocessing of the attenuation map for a higher cost of $O(n^3 \log n)$ unless the absorption map is represented as a coarser $n^{2/3} \times n^{2/3}$ image, which can be justified provided that absorption is sufficiently smooth.

There are other fast algorithms to invert the Radon transform [7–9, 12]. There are also many other techniques to invert the AtRT, based on the inversion formula (and the filtered back-projection) [10, 14, 18], or on other methods [16, 17, 25]. One main advantage of the slant stack method is its accuracy for a reasonable computational cost. Provided one is careful in choosing a sufficiently large number of measured directions, the spectral radii of the continuous and discrete operators involved in the calculations are very close to each other. As we shall see, this is particularly important in the inversion of the AtRT (or ERT) from partial measurements. Also provided that the absorption map is sufficiently smooth, the fast version of the algorithm allows us to invert the AtRT quite rapidly.

The rest of the paper is organized as follows. Section 2 introduces the Radon transform in the slant stack variables and the continuous reconstruction formulae. Section 3 recalls the fast slant stack algorithm and details its implementation. Section 4 addresses the calculation of the adjoint operator and the inversion of the Radon transform. Numerical simulations illustrate the method. Sections 2 to 4 closely follow the presentation in [2] and introduce notation and

implementation techniques we need in subsequent sections. In section 5, we generalize the slant stack algorithm to the AtRT with full measurements. Fast algorithms (of complexity $O(Nn^2 \log n)$) for ERT and AtRT are introduced in section 6. Finally we consider in section 7 the reconstruction of source terms from their ERT or AtRT with 180° measurements.

2. Radon transform and slant stack

Radon transform. The Radon transform $g(s, \theta)$ of a function $f(\mathbf{x})$ over \mathbb{R}^2 is defined for $s \in \mathbb{R}$ and $\theta \in (0, 2\pi)$ as

$$g(s, \theta) = Rf(s, \theta) = \int_{\mathbb{R}} f(s\boldsymbol{\theta}^\perp + t\boldsymbol{\theta}) dt = \int_{\mathbb{R}^2} f(\mathbf{x})\delta(\mathbf{x} \cdot \boldsymbol{\theta}^\perp - s) d\mathbf{x}. \quad (1)$$

Throughout we identify $\boldsymbol{\theta} = (\cos \theta, \sin \theta)$ for $\theta \in (0, 2\pi)$ and $\boldsymbol{\theta} \in S^1$. Let us define the Hilbert transform as

$$Hu(t) = \frac{1}{\pi} \int_{\mathbb{R}} \frac{u(s)}{t-s} ds, \quad (2)$$

where the above integrals need be taken in the principal value sense. It is well known that the function $f(\mathbf{x})$ can be reconstructed from its Radon transform $Rf(s, \theta)$ using the formula

$$f(\mathbf{x}) = \frac{1}{4\pi} \int_0^{2\pi} \boldsymbol{\theta}^\perp \cdot \nabla(HRf)(\mathbf{x} \cdot \boldsymbol{\theta}^\perp, \theta) d\theta = \frac{1}{4\pi} \int_0^{2\pi} \left(H \frac{\partial}{\partial s} Rf\right)(\mathbf{x} \cdot \boldsymbol{\theta}^\perp, \theta) d\theta. \quad (3)$$

This can be recast as

$$f(\mathbf{x}) = \frac{1}{4\pi} R^* I^{-1} Rf(\mathbf{x}), \quad (4)$$

where R^* is the formal adjoint to R and I^{-1} the Riesz potential, defined respectively by

$$R^*g(\mathbf{x}) = \int_0^{2\pi} g(\mathbf{x} \cdot \boldsymbol{\theta}^\perp, \theta) d\theta, \quad I^{-1} = \frac{\partial}{\partial s} H = H \frac{\partial}{\partial s}. \quad (5)$$

Let us define the Fourier transform of a function $f(\mathbf{x})$ over \mathbb{R}^d as

$$\hat{f}(\boldsymbol{\xi}) = \int_{\mathbb{R}^d} e^{-i\boldsymbol{\xi} \cdot \mathbf{x}} f(\mathbf{x}) d\mathbf{x}. \quad (6)$$

We still denote by $\hat{f}(\sigma, \theta)$ the Fourier transform of a function $f(s, \theta)$ with respect to the first variable only. A remarkable property of all the operators involved in the reconstruction (4) is their local expression in the Fourier domain. Indeed we have

$$\begin{aligned} \widehat{Rf}(\sigma, \theta) &= \hat{f}(\sigma\boldsymbol{\theta}^\perp), & \widehat{Hf}(\sigma) &= -i \operatorname{sign}(\sigma) \hat{f}(\sigma), \\ \widehat{R^*g}(\boldsymbol{\xi}) &= \frac{1}{|\boldsymbol{\xi}|} \left(\hat{g}\left(|\boldsymbol{\xi}|, \xi - \frac{\pi}{2}\right) + \hat{g}\left(-|\boldsymbol{\xi}|, \xi + \frac{\pi}{2}\right) \right) = \frac{2}{|\boldsymbol{\xi}|} \hat{g}\left(|\boldsymbol{\xi}|, \xi - \frac{\pi}{2}\right), \end{aligned} \quad (7)$$

where $\hat{\boldsymbol{\xi}} = \boldsymbol{\xi}/|\boldsymbol{\xi}| = (\cos \xi, \sin \xi)$, and where the latter equality holds only when $g(s, \theta) = g(-s, \theta + \pi)$, which is the case for the Radon transform (1). These local expressions are very useful when it comes to discretizing the reconstruction formula and solving it numerically. There is a major difficulty however coming from the fact that \widehat{Rf} involves the Fourier transform of f at $\sigma\boldsymbol{\theta}^\perp$. Therefore a discretization of σ and θ provides a ‘polar’ discretization of $\hat{f}(\boldsymbol{\xi})$. The interpolation from polar coordinates back to Cartesian coordinates, turns out to be rather challenging, especially if one wants to perform it fast [22].

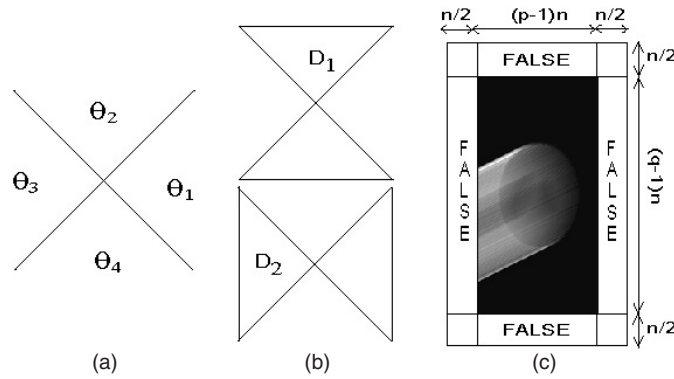


Figure 1. (a) Angular decomposition of $(0, 2\pi)$. (b) In (13), $\widehat{Sf}(\omega, \theta)$ depends only on $\hat{f}(D_1)$ for $\theta \in \Theta_1 \cup \Theta_3$ and on $\hat{f}(D_2)$ for $\theta \in \Theta_2 \cup \Theta_4$. (c) Domain of validity of $C_\theta A_{u,v}$ in (43) below using $pn \times qm$ arrays.

Slant stack. A way to overcome some of the difficulties is to introduce a different transform that is more Cartesian-friendly [2, 3, 11, 25]. We first decompose the angular domain into two subdomains

$$\Theta_1 = \left[-\frac{\pi}{4}, \frac{\pi}{4} \right), \quad \Theta_2 = \left[\frac{\pi}{4}, \frac{3\pi}{4} \right). \tag{8}$$

The integration on lines parametrized with angles in $\Theta_3 = [3\pi/4, 5\pi/4)$ and $\Theta_4 = [5\pi/4, 7\pi/4)$ can be performed by using the symmetry of the Radon transform $g(s, \theta) = g(-s, \theta + \pi)$; see figure 1(a). Up to boundary points, the domain Θ_1 corresponds to those angles θ such that $|\tan \theta| < 1$ whereas Θ_2 corresponds to those angles θ such that $|\cot \theta| < 1$. We then define the Radon transform of a function $f(\mathbf{x})$ over \mathbb{R}^2 in these coordinates as

$$Sf(t, \theta) = \begin{cases} \int_{\mathbb{R}} f(x, x \tan \theta + t) \frac{dx}{\cos \theta}, & \theta \in \Theta_1 \\ \int_{\mathbb{R}} f(y \cot \theta - t, y) \frac{dy}{\sin \theta}, & \theta \in \Theta_2. \end{cases} \tag{9}$$

It is defined for $t \in \mathbb{R}$ and $\theta \in \Theta_1 \cup \Theta_2$ and can be extended to $\theta \in [0, 2\pi)$ by setting $Sf(t, \theta + \pi) = Sf(t, \theta)$, which is equivalent in the variables (t, θ) to $Rf(s, \theta) = Rf(-s, \theta + \pi)$ in the variables (s, θ) . We verify that $Sf(t, \theta) = Rf(t \cos \theta, \theta)$ on Θ_1 and $Sf(t, \theta) = Rf(t \sin \theta, \theta)$ on Θ_2 so that there is a one-to-one correspondence between Rf and Sf . To differentiate it from the Radon transform in the variables (s, θ) we call $Sf(t, \theta)$ the *slant stack* transform, in reference to its use in geophysics. The adjoint operator to Sf (with respect to the usual scalar product in the variables t and θ) is then, using the notation $\mathbf{x} = (x, y)$,

$$S^*g(\mathbf{x}) = \int_{\Theta_1} g(y - x \tan \theta, \theta) \frac{d\theta}{\cos \theta} + \int_{\Theta_2} g(-x + y \cot \theta, \theta) \frac{d\theta}{\sin \theta}, \tag{10}$$

which we decompose with obvious notation as $S_1^*g(\mathbf{x}) + S_2^*g(\mathbf{x})$. Some algebra shows that

$$(HRf)(s, \theta) = (HSf)(t, \theta), \tag{11}$$

where $s = s(t) = t \cos \theta$ on Θ_1 and $s = s(t) = t \sin \theta$ on Θ_2 . Thus for $\theta \in \Theta_1$ with $s = t \cos \theta$, we obtain for instance that

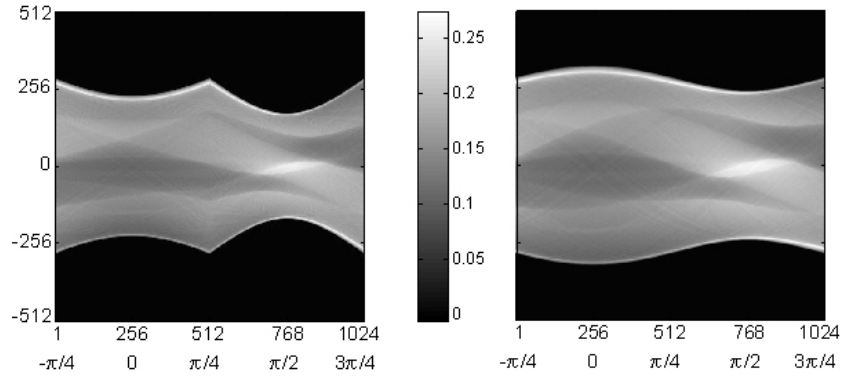


Figure 2. Linogram of the slant stack transform (left) compared with the usual sinogram of the Radon transform (right) for the phantom given in figure 3.

$$\begin{aligned} \left(H \frac{\partial}{\partial s} Rf\right) (\mathbf{x} \cdot \boldsymbol{\theta}^\perp, \theta) &= \frac{1}{\cos \theta} \left(H \frac{\partial}{\partial t} Sf\right) (y - x \tan \theta, \theta) \\ &= \boldsymbol{\theta}^\perp \cdot \nabla(HRf)(\mathbf{x} \cdot \boldsymbol{\theta}^\perp, \theta) \\ &= \frac{1}{\cos \theta} \frac{\partial}{\partial y} (HSf)(y - x \tan \theta, \theta), \end{aligned}$$

and a similar expression for $\theta \in \Theta_2$. Thus (4) in the new variables can be recast as

$$f(\mathbf{x}) = \frac{1}{2\pi} \left(S_1^* \left(\frac{\partial}{\partial y} HSf \right) (\mathbf{x}) + S_2^* \left(\frac{\partial}{\partial x} HSf \right) (\mathbf{x}) \right). \tag{12}$$

The derivatives along the direction $\boldsymbol{\theta}^\perp$ that appear in (3) are now replaced by derivatives along the directions of the main axes. This is one reason why this transform is Cartesian-friendly. We have seen that the operators R and R^* were local in the Fourier domain. This is also the case for the operators S and S^* . Some algebra shows that

$$\widehat{Sf}(\omega, \theta) = \begin{cases} \frac{1}{\cos \theta} \hat{f}(-\omega \tan \theta, \omega), & \theta \in \Theta_1, \\ \frac{1}{\sin \theta} \hat{f}(-\omega, \omega \cot \theta), & \theta \in \Theta_2. \end{cases} \tag{13}$$

$$\widehat{S^*g}(\boldsymbol{\xi}) = 4\pi \left(\frac{\chi(|\xi_x| < |\xi_y|)}{|\xi_y|} + \frac{\chi(|\xi_y| < |\xi_x|)}{|\xi_x|} \right) \hat{f}(\boldsymbol{\xi}) = \frac{4\pi \hat{f}(\boldsymbol{\xi})}{\max(|\xi_x|, |\xi_y|)}.$$

Here we have used the notation $\boldsymbol{\xi} = (\xi_x, \xi_y)$. In the formula for $\widehat{S^*g}$ we recognize the sum $\widehat{S_1^*g} + \widehat{S_2^*g}$. The Fourier transforms thus remain local. Moreover \widehat{Sf} involves the direction of integration θ in only *one* of the variables of $\hat{f}(\boldsymbol{\xi})$. This property is exploited in the numerical implementation of the inversion using the variables t and θ instead of s and θ . Note that $(-\omega \tan \theta, \omega)$ remains in the domain D_1 of figure 1(b) for $\omega \in \mathbb{R}$ since $|\tan \theta| \leq 1$. A comparison of the slant stack transform and the classical Radon transform for the source term given in figure 3 is shown in figure 2.

3. Fast slant stack algorithm

We use (12) and (13) to devise a fast and accurate algorithm to calculate and invert the Radon transform: the fast slant stack algorithm.

We have seen that the slant stack transform takes different expressions for $\theta \in \Theta_1$ and $\theta \in \Theta_2$. We can give them a unified expression by realizing that Θ_2 is the rotation of Θ_1 by $\pi/2$. Let us denote by \mathcal{R} the (counterclockwise) rotation by $\pi/2$ and define $\Theta_k = \mathcal{R}^{k-1}\Theta_1$ for $k = 1, 2, 3, 4$. For $\theta \in [0, 2\pi)$, we define the domain index to which θ belongs by

$$k = k(\theta) \quad \text{such that} \quad 1 \leq k \leq 4 \quad \text{and} \quad \mathcal{R}^{1-k}\theta \in \Theta_1. \tag{14}$$

Let us also define the rotation of the function f by \mathcal{R} as $\mathcal{R}f(\mathbf{x}) = f(\mathcal{R}^{-1}\mathbf{x})$. We then define the slant stack transform as

$$Sf(t, \theta) = \int_{\mathbb{R}} (\mathcal{R}^{1-k}f)(x, x \tan \mathcal{R}^{k-1}\theta) \frac{dx}{\cos \mathcal{R}^{k-1}\theta}. \tag{15}$$

This implies that we can concentrate on the calculation of $Sf(t, \theta)$ for $\theta \in \Theta_1$ and obtain the whole slant stack transform by using the same algorithm after rotation of the function f . In the following, we therefore consider *only* $\theta \in \Theta_1$.

Let us describe how we discretize the function f . Let n be a given power of 2 (to simplify). We consider images composed of $n \times n$ pixels, where the first variable refers to the column index and the second variable to the row index. For the following, we define $m = 2n$. We denote by $F_{u,v}$ for $-n/2 \leq u, v < n/2$ the $\mathbb{C}^{n \times n}$ array of values at the pixels. We define the image F^1 augmented by zeros as

$$F^1_{u,v} = (E^1F)_{u,v}, \tag{16}$$

where E^1 is the operator padding an array $n \times n$ to be an $n \times m$ array by adding $n/2$ rows of zeros ‘above’ and $n/2$ rows of zeros ‘below’. Let us introduce the set of integers for n an even integer:

$$\mathcal{T}_n = \left\{ -\frac{n}{2}, -\frac{n}{2} + 1, \dots, \frac{n}{2} - 1 \right\}. \tag{17}$$

Next we define the interpolant

$$D_m(t) = \frac{1}{m} \sum_{k \in \mathcal{T}_m} e^{i2\pi(k+\frac{1}{2})t} = \frac{\sin m\pi t}{m \sin t}, \tag{18}$$

and the interpolations

$$F^1_u(y) = \sum_{v \in \mathcal{T}_n} F_{u,v} D_m(y - v) = \sum_{v \in \mathcal{T}_m} F^1_{u,v} D_m(y - v). \tag{19}$$

We then define $S_n F_t(\theta)$, the semi-discrete slant stack transform of F , as

$$S_n F_t(\theta) = \frac{1}{\cos \theta} \frac{1}{n} \sum_{u \in \mathcal{T}_n} F^1_u(u \tan \theta + t), \quad \theta \in \Theta_1. \tag{20}$$

The transform is defined for $t \in \mathcal{T}_m$ and for the not yet discretized variable $\theta \in \Theta_1$. We can interpret the above algorithm as follows. Consider $\theta \in \Theta_1$ and introduce the translation operator of length τ acting on \mathbb{C}^m -vectors:

$$(T_\tau V)_u = \sum_{v \in \mathcal{T}_m} V_v D_m(u + \tau - v), \quad u \in \mathcal{T}_m. \tag{21}$$

The *slanted* image is defined by

$$(\mathcal{S}_\theta F)_{u,v} = (T_{u \tan \theta} F^1)_{u,v}, \quad u \in \mathcal{T}_n, v \in \mathcal{T}_m, \tag{22}$$

where the translation by $u \tan \theta$ is applied in the vertical v coordinate for each horizontal index u fixed. We then verify that

$$S_n F_t(\theta) = \frac{1}{\cos \theta} \frac{1}{n} \sum_{u \in \mathcal{T}_n} (\mathcal{S}_\theta F)_{u,t}, \quad t \in \mathcal{T}_m, \theta \in \Theta_1. \tag{23}$$

The semi-discrete slant stack is thus performed following these steps: we first zero-pad F to get F^1 . We next slant F^1 column by column by an amount $-u \tan \theta$ proportional to the column index u . The slant stack transform is finally calculated by summing for each row the terms across the columns.

Semi-discrete Fourier transform and Fourier slice theorem. A key property of the above construction is that it satisfies a discrete version of the Fourier slice theorem, the first equations in (7) and (13). Let us define the semi-discrete Fourier transform of $S_n F$ as

$$\widehat{S_n F}(\sigma, \theta) = \frac{1}{m} \sum_{t \in \mathcal{T}_m} e^{-i\sigma t} S_n F_t(\theta), \quad \sigma \in \mathbb{R}, \quad \theta \in \Theta_1. \quad (24)$$

We verify that

$$\widehat{S_n F}(\sigma, \theta) = \frac{1}{\cos \theta} \frac{1}{nm} \sum_{u \in \mathcal{T}_n} \sum_{v \in \mathcal{T}_m} F_{u,v}^1 \sum_{t \in \mathcal{T}_m} e^{-i\sigma t} D_m(u \tan \theta + t - v).$$

From the definition of D_m we obtain that for a wavenumber of the form $\sigma = \frac{2\pi}{m}(k + \frac{1}{2})$, we have

$$\widehat{S_n F} \left(\frac{2\pi}{m} \left(k + \frac{1}{2} \right), \theta \right) = \frac{1}{\cos \theta} \widehat{F}^1 \left(-\frac{2\pi}{m} \left(k + \frac{1}{2} \right) \tan \theta, \frac{2\pi}{m} \left(k + \frac{1}{2} \right) \right). \quad (25)$$

This is a discrete version of the Fourier slice theorem, where we have defined the semi-discrete Fourier transform:

$$\widehat{F}^1 \left(\frac{2\pi l}{n}, \frac{2\pi(k + \frac{1}{2})}{m} \right) = \frac{1}{nm} \sum_{u \in \mathcal{T}_n} \sum_{v \in \mathcal{T}_m} F_{u,v}^1 \exp \left(-i \left(\frac{2\pi l}{n} u + \frac{2\pi}{m} \left(k + \frac{1}{2} \right) v \right) \right). \quad (26)$$

Discrete and fractional Fourier transform. We see from (25) that the slant stack transform can be computed efficiently in the Fourier domain provided one is capable of estimating the left-hand side in (25) fast. This can be done, using the fractional Fourier transform, provided that $\tan \theta$ is linearly related to the first variable in \widehat{F}^1 . More specifically let us introduce the set

$$\Theta_1^n = \left\{ \theta_l = \arctan \frac{2l}{n}, l \in \mathcal{T}_n \right\}, \quad (27)$$

where $\tan \theta$ is chosen uniformly distributed on $(-1, 1)$. Then we define the fully discrete slant stack transform as

$$S_n F_{t,l} = S_n F_t(\theta_l). \quad (28)$$

We observe that it can be calculated as

$$S_n F_{t,l} = \sum_{k \in \mathcal{T}_m} \exp \left(i \frac{2\pi}{m} \left(k + \frac{1}{2} \right) t \right) \widehat{S_n F}_{k,l}, \quad t \in \mathcal{T}_m, \quad l \in \mathcal{T}_n, \quad (29)$$

where

$$\widehat{S_n F}_{k,l} = \sqrt{1 + \left(\frac{2l}{n} \right)^2} \widehat{F}^1 \left(-\frac{2\pi}{m} \left(k + \frac{1}{2} \right) \frac{2l}{n}, \frac{2\pi}{m} \left(k + \frac{1}{2} \right) \right). \quad (30)$$

The calculation of $S_n F_{t,l}$ for $(t, l) \in \mathcal{T}_m \times \mathcal{T}_n$ can be obtained from $\widehat{S_n F}_{k,l}$ for $(k, l) \in \mathcal{T}_m \times \mathcal{T}_n$ in $O(n^2 \log n)$ by fast Fourier transform (FFT). The fast calculation of $\widehat{S_n F}_{k,l}$ is obtained as follows. Using the FFT we can calculate

$$\tilde{F}_u^1 \left(\frac{2\pi}{m} \left(k + \frac{1}{2} \right) \right) = \frac{1}{m} \sum_{v \in \mathcal{T}_m} \exp \left(-i \frac{2\pi}{m} \left(k + \frac{1}{2} \right) v \right) F_{u,v}^1,$$

for $k \in \mathcal{T}_m$ in $O(n^2 \log n)$ operations.

Let us introduce the discrete *fractional Fourier transform*, which maps a vector $V_u \in \mathcal{T}_n$ to a vector $(\mathcal{F}_\alpha V)_l \in \mathcal{T}_n$ as

$$(\mathcal{F}_\alpha V)_l = \sum_{u \in \mathcal{T}_n} V_u e^{-i \frac{2\pi}{n} \alpha l u}. \tag{31}$$

For a fixed $k \in \mathcal{T}_m$ we observe that as vectors indexed by $l \in \mathcal{T}_n$,

$$\left(\mathcal{F}_{-\frac{2(k+1/2)}{m}} \tilde{F}_u^1 \left(\frac{2\pi}{m} \left(k + \frac{1}{2} \right) \right) \right)_l = \hat{F}^1 \left(-\frac{2\pi}{m} \left(k + \frac{1}{2} \right) \frac{2l}{n}, \frac{2\pi}{m} \left(k + \frac{1}{2} \right) \right). \tag{32}$$

Thus, the calculation of the right-hand side in (30) can be obtained by applying m fractional Fourier transforms on vectors of size n . This can be done in $O(m \times n \log n) = O(n^2 \log n)$.

Remark on the interpolation. We have chosen to use the interpolant (18) and the corresponding de-centred discrete wavenumbers $2\pi(k + \frac{1}{2})$ for $k \in \mathcal{T}_m$ for two reasons. First, the interpolant is real valued so that the slant stack transform $S_n F_{t,l}$ is real valued. Second, the definition of the interpolation involves exactly m discrete wavenumbers (in each dimension), which corresponds to the image we want to consider.

We could choose different interpolants such as

$$C_m(t) = \frac{1}{m} \sum_{k \in \mathcal{T}_m} e^{i2\pi kt} = e^{i \frac{\pi}{m} t} \frac{\sin m\pi t}{m \sin t}, \quad K_m(t) = \frac{1}{m} \sum_{k=-m/2}^{m/2}{}' e^{i2\pi kt} = \frac{\sin(m\pi t)}{m \tan(t/2)},$$

where $'$ means that the boundary coefficient at $k = -m/2$ and $k = m/2$ are halved. The interpolant C_m involves only m values but is not real valued. Even though the phase $e^{i \frac{\pi}{m} t}$ is small for large m , the calculated Radon transform is not real valued. The main disadvantage is that applying the inversion operator to the real part of the calculated Radon transform does not provide as nice results as when it is applied to the complex-valued transform. The operator K_m is real valued but requires consideration of $m + 1$ wavenumbers instead of m , which makes the implementation of the reconstruction algorithm slightly more involved.

Implementation. We have seen that the discrete calculation of $Sf(t, \theta)$ can be performed by considering $\theta \in \Theta_1$ after suitable rotation. Because $n \times n$ images $F_{u,v}$ with n even are considered, the origin of the image turns out to be the pixel $(n/2 + 1, n/2 + 1)$, which is then not invariant by rotation by a multiple of 90° . To overcome the problem we embed the image into a $(n + 1) \times (n + 1)$ image \tilde{F} by adding a column of zeros to the right of F and a row of zeros to the top of F . Moreover we assume that the bottom row and the left column of F are filled with zeros. We can then apply rotations to \tilde{F} and remove the extra column and extra row to come back to an $n \times n$ image. This allows us to implement the operator for $\theta \in \Theta_n^1$ and compute the integrations in other directions by rotation.

For the attenuated Radon transform, we have to consider four operators corresponding to the angles $(-\pi/4, \pi/4)$, $(\pi/4, 3\pi/4)$, $(3\pi/4, 5\pi/4)$ and $(5\pi/7\pi/4)$. The latter three operators are mapped to the first one by appropriate rotation of the original image. The implementation of the first operator is obtained as follows:

- (i) We add zeros to the image F to obtain F^1 .
- (ii) We compute a discrete Fourier transform (DFT) on the columns.
- (iii) We compute a fractional DFT on the rows.
- (iv) We compute an inverse DFT (IDFT) on the columns.

Each of these operations can be performed in $O(n^2 \log n)$ operations. Note also that the calculations on each angular domain Θ_n^k can be performed independently, hence in parallel.

Accuracy of the procedure. We can show that the above procedure has spectral accuracy to estimate the slant stack transform, and equivalently the Radon transform. A proof of this result

can be found in [3]. It can also be obtained from the following observation. Let us define the interpolating function

$$f(x, y) = \sum_{u, v \in \mathcal{T}_m} F(u, v) \operatorname{sinc}\left(\pi\left(x - \frac{u}{m}\right)\right) D_m\left(y - \frac{v}{m}\right),$$

where $\operatorname{sinc}(x) = \sin(x)/x$. A direct calculation shows that for $t \in \mathcal{T}_n$,

$$Sf(\theta, t) = \frac{2\pi}{\cos(\theta)} \sum_{k' \in \mathcal{T}_m} e^{i2\pi(k'+1/2)t} \sum_{u, v \in \mathcal{T}_m} f(u, v) e^{-i\frac{2\pi}{m}(k'+1/2)(v-\tan(\theta)u)} = S_n F_t(\theta).$$

The values of the slant stack transform of suitable polynomial interpolations of the images $F(u, v)$ at the slant stack grid are thus calculated *exactly*.

4. Adjoint transform and inversion

A discretization of the inversion of the slant stack transform can also be obtained fast. We see from (12) that we need to discretize the adjoint operator to S , the Hilbert transform, and the differentiation in y and x . Again, up to appropriate rotation, it is sufficient to consider $\theta \in \Theta_1$. We can thus recast (20)–(28) as

$$S_n F_{t,l} = \sqrt{1 + \left(\frac{2l}{n}\right)^2} \frac{1}{n} \sum_{u \in \mathcal{T}_n} \sum_{v \in \mathcal{T}_m} F_{u,v}^1 D_m(ul + t - v). \quad (33)$$

The adjoint of this operator thus takes the form

$$S_n^* G_{u,v} = \frac{1}{n} \sum_{l \in \mathcal{T}_n} \sum_{t \in \mathcal{T}_m} \left[\sqrt{1 + \left(\frac{2l}{n}\right)^2} G_{t,l} \right] D_m(ul + t - v). \quad (34)$$

The adjoint operator S_n^* has therefore the same structure as S_n with $\sqrt{1 + (2l/n)^2} G_{t,l}$ playing an equivalent role to $F_{u,v}$. It can thus also be estimated with a computational cost in $O(n^2 \log n)$ using discrete Fourier and fractional Fourier transforms.

It remains to discretize the Riesz operator in (12), which for the angles $\theta \in \Theta_1$ corresponds to $\frac{\partial}{\partial y} H$. These operators are diagonal in the Fourier domain and so we define I_n^{-1} , the discretization of the Riesz operator, which acts on vectors, as the succession of the following three steps:

- (i) DFT on vector V_k to obtain \hat{V}_k ;
- (ii) multiplication of \hat{V} by Λ of coefficients $\Lambda_k = \frac{2\pi}{m} \left|k + \frac{1}{2}\right|$ for $k \in \mathcal{T}_m$;
- (iii) IDFT on vector $(\Lambda \hat{V})_k$.

This step can also be computed in $O(n^2 \log n)$ operations.

The contributions coming from $\theta \in \Theta_2$ are calculated similarly after suitable rotation. We therefore have an algorithm that can calculate the Radon transform (or the slant transform) of a $n \times n$ image in $O(n^2 \log n)$ operations. It takes the same order of calculations to discretize the inversion operator.

Exact numerical inversion. With S_n the operator, which to an image F maps the discrete Radon transform $G = S_n F$, our numerical reconstruction consists then in applying $S_n^* I_n^{-1}$ to the data $G = S$. Whereas $S^* I^{-1} S = Id$, the identity operator, for instance in the L^2 sense in the case of continuous functions, this is no longer the case for the discretized operators. We define the operator

$$G_n = S_n^* I_n^{-1} S_n. \quad (35)$$

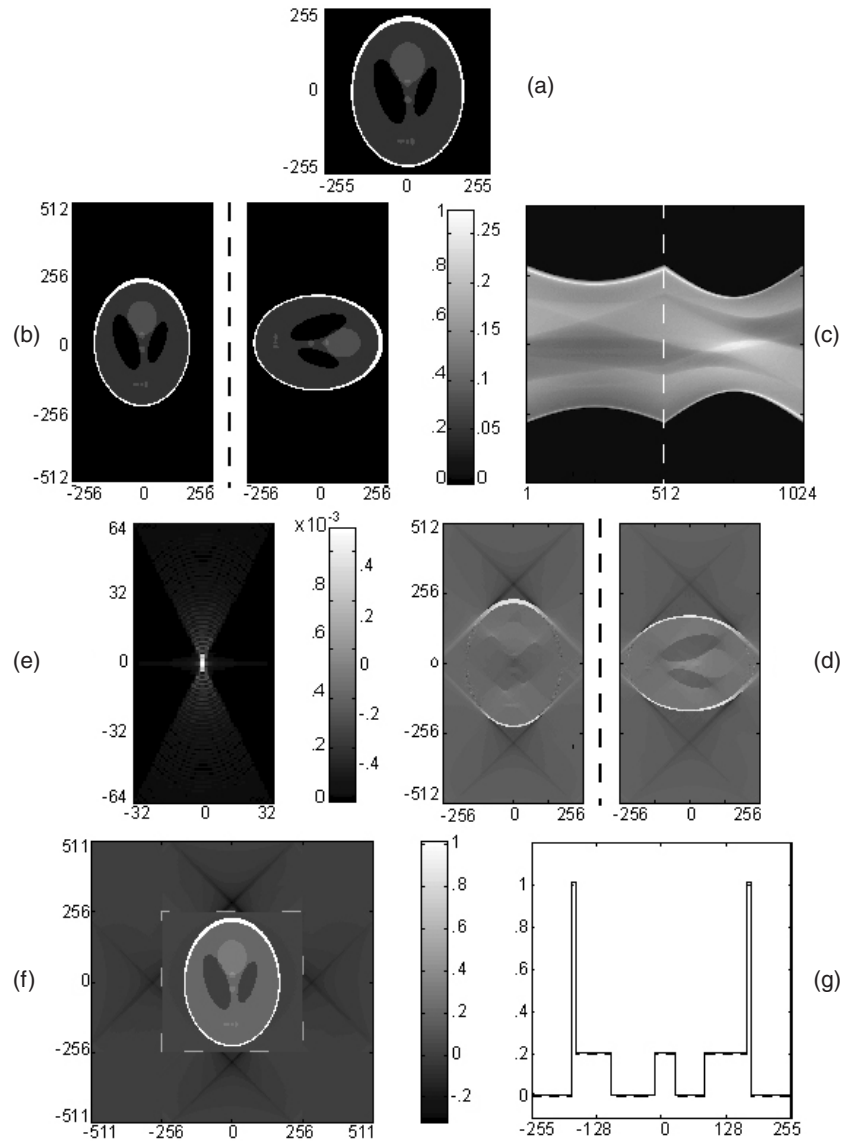


Figure 3. (a) Initial phantom. (b) Same image and rotated image with vertical zero-padding. (c) Slant stack transform of images in (b). (d) Reconstructions using data in (c). (e) Fourier transform of left image in (d) whose support is theoretically in D_1 . (f) Superposition of left and right images in (d) after appropriate rotation. Only the central part is of interest. (g) Cross section of initial and reconstructed images along $y = 0$.

This is a symmetric operator with two principal characteristics: $G_n F$ has a computational cost of $O(n^2 \log n)$, and G_n is spectrally relatively close to identity. It can be inverted by conjugate gradient (CG) [13] as proposed in [2].

Numerical simulations. The fast slant stack algorithm is explained in the numerical simulation of figure 3. We observe that the reconstruction is visually almost perfect. Only the low frequencies (note the relatively constant shift between the initial and reconstructed images on the cross section $y = 0$) are not perfectly reconstructed.

Table 1. Spectral data (three smallest and two largest eigenvalues) for different simulations: n , number of pixels of image; ZP , additional zero-padding such that the algorithm zero-pads the original image into a $2n \times 2n$ image for $ZP = 1$; CG , the number of conjugate gradient iterations to invert G_n .

Case (n, ZP, CG)	First	Second	Third	$n - 1$	Last
16, 0, 0	0.886 86	0.886 86	0.978 78	1.089 7	1.3531
32, 0, 0	0.825 01	0.825 01	0.977 95	1.0984	1.4539
32, 1, 0	0.995 66	0.995 66	0.997 55	1.0204	1.0615
32, 0, 4	0.999 93	1.0001
64, 0, 0	0.759 9	0.759 9	0.962 66	1.1097	1.534
64, 1, 0	0.995 85	0.995 85	0.996 9	1.0212	1.0657
64, 0, 4	0.999 83	1.0004
128, 0, 0	0.696 75	0.696 75	0.938 82	1.1543	1.5977

The quality of the reconstruction is closely related to the quality of the near identity matrix G_n introduced in (35). We have estimated its spectrum numerically using Matlab standard procedures. The results are reported in table 1. We observe that few eigenvalues depart significantly from 1 though this number does not improve as $n \rightarrow \infty$. A better numerical inversion can be obtained in two ways. One can either apply several iterations of conjugate gradient to invert G_n , or zero-pad the image into a bigger image ($2n \times 2n$ in our simulations).

The cost of zero-padding the image into a $2n \times 2n$ image is roughly four times the initial cost since there are twice the directions θ_j and twice the offsets t_k . Note however that the inversion, which is very easy to implement, also requires four times more measured data. The cost of p iterations of conjugate gradient is $2p + 1$ times the initial cost. Indeed the cost I_n^{-1} is asymptotically negligible compared to that of S_n and S_n^* . So the cost of one CG iteration (multiplication by G_n) is roughly twice the cost of a multiplication by S_n^* .

5. AtRT reconstruction from full measurements

Let us introduce the symmetrized beam transform of $a(\mathbf{x})$ as

$$D_\theta a(\mathbf{x}) = \frac{1}{2} \left(\int_0^\infty [a(\mathbf{x} - t\boldsymbol{\theta}) - a(\mathbf{x} + t\boldsymbol{\theta})] dt \right). \quad (36)$$

The AtRT is then the weighted Radon transform defined by

$$R_a f(s, \theta) = R[e^{D_\theta a(\mathbf{x})} f(\mathbf{x})](s, \theta) \equiv R_{a, \theta} f(s). \quad (37)$$

In practice the measured data are given by $e^{-Ra(s, \theta)/2} R_a f(s, \theta)$ [6, 22]. In the numerical calculation of $R_a f(s, \theta)$, a difficulty arises from the fact that the weight $e^{D_\theta a(\mathbf{x})}$ depends on the angular direction θ . It is also the main obstacle to generalizing the reconstruction formula (4). A reconstruction formula was recently obtained in [27]. Let us define

$$i\varphi(\mathbf{x}, \theta) = R_{-a, \theta}^*(2H_a) R_{a, \theta} f(\mathbf{x}), \quad H_a = \frac{1}{2}(C_c H C_c + C_s H C_s), \quad (38)$$

where $R_{a, \theta}^* g(\mathbf{x}) = e^{D_\theta a(\mathbf{x})} g(\mathbf{x} \cdot \boldsymbol{\theta}^\perp)$ and

$$C_c g(s, \theta) = g(s, \theta) \cos\left(\frac{H R a(s, \theta)}{2}\right), \quad C_s g(s, \theta) = g(s, \theta) \sin\left(\frac{H R a(s, \theta)}{2}\right). \quad (39)$$

The operator H_a can be seen as a generalization of the Hilbert transform, with $H_a = H/2$ when $a \equiv 0$. The Novikov reconstruction formula (see [6, 18, 27]) states that

$$f(\mathbf{x}) = \frac{1}{4\pi} \int_0^{2\pi} \boldsymbol{\theta}^\perp \cdot \nabla(i\varphi)(\mathbf{x}, \theta) d\theta. \quad (40)$$

We generalize the slant stack transform and its numerical implementation to the calculation of the attenuated Radon transform $R_a f(s, \theta)$ and to its inversion via (40). Because there is no generalization of the Fourier slice theorem to the case of the attenuated Radon transform, we cannot use the fractional Fourier transform and obtain a computational cost in $O(n^2 \log n)$. However, we can still use the idea that it is easy to sum coefficients along the axis x after slanting the image by the correct amount. This is the technique we have implemented to solve the AtRT. A fast algorithm, based on a Fourier decomposition of the weight $e^{D_\theta a}$, will be presented in section 6.

In order to discretize (37) we need first to calculate $D_\theta a(\mathbf{x})$ and second to perform the one-dimensional integration.

Calculation of the attenuation weight. To calculate $D_\theta a(\mathbf{x})$ it is sufficient to estimate terms of the form

$$C_\theta a(\mathbf{x}) = \int_0^\infty a(\mathbf{x} + t\boldsymbol{\theta}) dt = \int_0^\infty a(x + t, y + t \tan \theta) \frac{dt}{\cos \theta}, \tag{41}$$

which we recast as

$$C_\theta a(\mathbf{x}) = \int_0^\infty (T_t \otimes T_{t \tan \theta}) a(x, y) \frac{dt}{\cos \theta}. \tag{42}$$

Here $T_t a(x) = a(x + t)$. The action of $T_t \otimes T_{t \tan \theta}$ is local in the Fourier domain. We shall use this property to discretize $C_\theta a$. Let $A_{u,v}$ for $u, v \in \mathcal{T}_n$ be the discrete attenuation map and let us define $A_{u,v}^{pq}$ the $pn \times qn$ array composed of the coefficients $A_{u,v}$ surrounded by $(p - 1)n/2$ columns of zeros to the left and to the right and by $(q - 1)n/2$ rows of zeros above and below.

We discretize (42) as

$$C_\theta A_{u,v} = \frac{1}{\cos \theta} \sum_{t=0}^{(q-1)n-1} (T_t \otimes T_{t \tan \theta}) A_{u,v}^{pq}. \tag{43}$$

We verify from the continuous expression (41) that $C_\theta A$ is correctly estimated for $u \in \mathcal{T}_{(p-1)n}$ and $v \in \mathcal{T}_{(q-1)n}$; see figure 1(c). The parameters p and q need to be chosen according to the size of the array on which one wants to obtain an accurate approximation of C_θ . In the Fourier domain we obtain

$$\begin{aligned} \widehat{C_\theta A}_{\xi, \zeta} &= \sum_{t=0}^{(q-1)n-1} \exp\left(i \frac{2\pi t}{m} (p\xi + q \tan \theta \zeta)\right) \frac{\hat{A}_{\xi, \zeta}^{pq}}{\cos \theta} \\ &= \frac{1 - \exp\left(i 2\pi \left(\frac{q-1}{q} \xi + \frac{q-1}{p} \tan \theta \zeta\right)\right) \hat{A}_{\xi, \zeta}^{pq}}{1 - \exp\left(i \frac{2\pi t}{n} \left(\frac{\xi}{q} + \frac{\zeta}{p} \tan \theta\right)\right) \cos \theta}. \end{aligned} \tag{44}$$

For each $\theta \in \Theta_1$ we can thus calculate $C_\theta A_{u,v}$ in $O(n^2 \log n)$ by using the FFT.

The numerical simulation of $C_{\theta_l} A_{u,v}$ for $u, v, l \in \mathcal{T}_m$ can thus be performed in $O(n^3 \log n)$ calculations. Since the array is three dimensional, there is little hope to obtain a better computational cost if one neglects the $\log n$ term. Note that since $\boldsymbol{\theta} \cdot \nabla C_\theta a + a = 0$ with appropriate boundary conditions, we could solve the latter partial differential equation by finite differences in $O(n^3)$ operations [19]. However the line integrations are not necessarily performed very accurately [5, 20]. We thus prefer to use the accurate method based on the same ideas as for the fast slant stack algorithm.

Attenuated slant stack. Let us now come back to the calculations of the AtRT $R_a f(s, \theta)$. Once $e^{D_\theta a}$ is calculated, this can be written in the form of a generalized Radon transform as follows:

$$Rf(s, \theta) = \int_{\mathbb{R}} f(s\boldsymbol{\theta}^\perp + t\boldsymbol{\theta}, \theta) dt. \tag{45}$$

In the slant stack variables, this is equivalent to

$$Sf(t, \theta) = \int_{\mathbb{R}} f(x, x \tan \theta + t, \theta) \frac{dx}{\cos \theta}, \quad (46)$$

assuming that $\theta \in \Theta_1$. We want to perform the calculation by first slanting f and then integrating along a horizontal line. Let us define $F_{u,v}(\theta)$ the image, which now depends on the variable θ , and its extension $F_{u,v}^1(\theta)$ where now $v \in \mathcal{T}_m$. We define the discrete slant stack transform as

$$S_n F_t(\theta) = \frac{1}{\cos \theta} \frac{1}{n} \sum_{u \in \mathcal{T}_n} (\mathcal{S}_\theta F(\theta))_{u,t}, \quad t \in \mathcal{T}_n. \quad (47)$$

Here \mathcal{S}_θ is the slanting operator defined in (22). This operation can be calculated relatively fast in the Fourier domain. Let us define $\tilde{F}_{u,\xi}$ as the Fourier transform with respect to the second variable only. We verify that

$$\widetilde{\mathcal{S}_\theta F_{u,\xi}} = \exp\left(i \frac{2\pi}{m} u \tan \theta \left(\xi + \frac{1}{2}\right)\right) \tilde{F}_{u,\xi}^1, \quad u \in \mathcal{T}_n, \xi \in \mathcal{T}_m. \quad (48)$$

So for each $\theta \in \Theta_1$ we can compute $S_n F_t(\theta)$ in $O(n^2 \log n)$ operations. The global calculation of $S_n F_t(\theta_l)$ for $t \in \mathcal{T}_m$ and $l \in \mathcal{T}_n$ can thus be performed in $O(n^3 \log n)$ operations. In the general case where $f(\mathbf{x}, \theta)$ genuinely depends on θ , there is little hope to do any better. However we show in the following section how we can obtain a fast algorithm when $f(\mathbf{x}, \theta)$ can be written as a sum of products of functions that depend only on space and direction, respectively.

Inversion of the AtRT. We have shown how to calculate the AtRT $R_a f(s, \theta)$ in $O(n^3 \log n)$ operations. We now show that the complexity of the inversion using the Novikov formula is of the same order. Let us define the attenuated slant stack transform as

$$g(t, \theta) = S_{a,\theta} f(t) = S_a f(t, \theta) = \int_{\mathbb{R}} e^{D_{\theta a}(x, x \tan \theta + t)} f(x, x \tan \theta + t) \frac{dx}{\cos \theta}, \quad (49)$$

for $\theta \in \Theta_1$ and similar expressions for $\theta \in \Theta_k$ for $k = 2, 3, 4$ obtained by suitable rotations. Then using (11), we can verify that the Novikov formula in the slant stack variables is given by

$$f(\mathbf{x}) = \frac{1}{4\pi} \left(\frac{\partial}{\partial y} \int_{\Theta_1 \cup \Theta_3} S_{-a,\theta}^* H_a g(\mathbf{x}) d\theta + \frac{\partial}{\partial x} \int_{\Theta_2 \cup \Theta_4} S_{-a,\theta}^* H_a g(\mathbf{x}) d\theta \right). \quad (50)$$

We define the operators $S_a^{k*} = \int_{\Theta_k} S_{a,\theta}^* d\theta$, $k = 1, 2, 3, 4$, as the adjoints to S_a on $\mathbb{R} \times \Theta_k$. The numerical inversion thus requires to approximate $H_a g(t, \theta)$, the derivatives in x and y , and the adjoint operators S_a^{k*} . The operator H_a , involving multiplications and the Hilbert transform H , is dealt with easily in the Fourier domain. The operators $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$ are also easily handled in the Fourier domain as a multiplication by $\frac{2\pi}{m} \left(k + \frac{1}{2}\right)$ for a cost proportional to $O(n^2 \log n)$. It thus remains to deal with the adjoint operators S_a^{k*} . By appropriate rotation, it is sufficient to consider S_a^{1*} , which takes the form

$$S_a^{1*} g(\mathbf{x}) = \int_{\Theta_1} S_{a,\theta}^* g(\mathbf{x}) d\theta = \int_{\Theta_1} e^{D_{\theta a}(x,y)} g(y - x \tan \theta, \theta) \frac{d\theta}{\cos \theta}. \quad (51)$$

The discretization of $S_a f(s, \theta)$ obtained earlier is

$$S_{an} F_{t,l} = \frac{1}{\cos \theta_l} \frac{1}{n} \sum_{u \in \mathcal{T}_n} \mathcal{S}_{\theta_l} [F e^{D_{\theta a}}(\theta_l)]_{u,t} = \frac{1}{\cos \theta_l} \frac{1}{n} \sum_{u \in \mathcal{T}_n} (T_{ul} [F e^{D_{\theta a}}(\theta_l)]^1)_{u,t}. \quad (52)$$

The index $[\cdot]^1$ means here that the array is $n \times m$ and the translation T acts on the second variable, here t . Now we have $T_{ul}^* = T_{-ul}$ so that

$$S_{an}^* G_{u,v} = \frac{1}{n} \sum_{l \in \mathcal{T}_n} \frac{1}{\cos \theta_l} [e^{D_{\theta^a}}(\theta_l)]_{u,v} (T_{-ul} G)_{v,l}. \quad (53)$$

Here the translation T_{-ul} is with respect to the first variable v . We see that the expressions for the discrete operators S_{an} and S_{an}^* are symmetrical. The calculation of $H_{u,v,l} = (T_{-ul} G)_{v,l}$ for $v \in \mathcal{T}_m$, $u, l \in \mathcal{T}_n$, can be performed in $O(n^3 \log n)$ in the Fourier domain. Indeed we verify that

$$\tilde{H}_{u,k,l} = \exp\left(-i2\pi \left(k + \frac{1}{2}\right) ul\right) \hat{G}_{k,l},$$

where \tilde{H} is the discrete Fourier transform of H in the second variable only. Note that we could obtain the summation in l of the above expression for a cost of $O(n^2 \log n)$ operations as we did in the preceding section by using the fractional Fourier transform. Here, however, because the exponential term depends on l , u and v , the summation in l cannot be performed fast. This shows that (53) can also be completed in $O(n^3 \log n)$ operations.

Exact numerical inversion. The generalization of (35) to the AtRT can formally be written

$$G_{an} = (S^* H)_{an} S_{an}, \quad (54)$$

where S_{an} is the discretization of the AtRT and $(S^* H)_{an}$ the discretization of the reconstruction operator. Unlike G_n however, the approximation of identity G_{an} need *not* be symmetric. In order to obtain a better approximation of identity, we thus use the fact that

$$I \sim (G_{an}^* G_{an})^{-1} G_{an}^* G_{an}. \quad (55)$$

This can be solved by conjugate gradient (we assume that the matrix is sufficiently well behaved) as in the case $a \equiv 0$. Other techniques such as GMRES [13] could certainly also be used to invert G_{an} directly. We do not consider them here. The cost of CG becomes relatively high and the implementation of G_{an}^* , which goes along the same lines as that of G_{an} needs to be implemented with some care. Assuming the inverse operator exists, the inversion of the numerical AtRT is thus given by

$$S_{an}^{-1} = (G_{an}^* G_{an})^{-1} G_{an}^* (S^* H)_{an}. \quad (56)$$

We shall see that $G_{an}^* G_{an}$ need not be invertible for certain choices of n and the absorption. To obtain an invertible operator, which is a good approximation of identity, we need to zero-pad the image into a bigger image. We consider below the case of zero-padding into $2n \times 2n$ images, which provides good inversions even for practically very high values of absorption.

Numerical examples. The above calculations show that the attenuated Radon transform can be computed and inverted in $O(n^3 \log n)$ operations. This is not competitive with the computational cost of order $O(n^2 \log n)$ we obtained for the calculation and inversion of the Radon transform. We shall see in section 6 how the inversion can be sped up by using a Fourier decomposition of $e^{D_{\theta^a}}$. We consider two types of absorption maps given in figures 4(i) and (ii). The left figure (i) represents an absorption map used in [18]. The right figure (ii) represents a smooth version of it obtained by multiplication in the Fourier domain by $\hat{g}_p(\xi) = \exp(-np|\xi|^2/2)$ with $p = 4$.

We consider two different experiments with two types of sources given in figure 4(a) and (b). These source profiles have been proposed in [18]. We now consider the reconstruction of the first source terms from its AtRT using the ‘slow’ algorithm in $O(n^3 \log n)$ operations. We use the non-smoothed absorption map (left of figure 4). The numerical simulation is

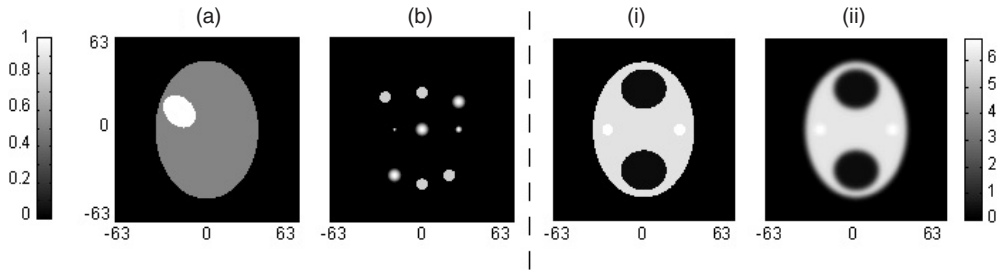


Figure 4. Left: two different source terms: (a) two ellipsoids; (b) nine spots. Right: two attenuation maps: (i) attenuation from [18]; (ii) regularization of the same attenuation map with $p = 4$.

presented in figure 5. Four iterations of conjugate gradient are used in the reconstruction (chosen to obtain a very accurate reconstruction). We observe again that the reconstruction from each quadrangle Θ_i provides only very partial information about the true source but that the sum of the four contributions offers a very good image.

In the absence of conjugate gradient iterations, we observe quite poor reconstructions. The results are shown in figure 6. The structure of the absorption map is clearly noticeable on the reconstructed image though this is only a numerical artefact. Two iterations of conjugate gradient (not shown) would be sufficient to render the absorption map almost unnoticeable. The reason for this behaviour is that the operator G_{an} moves further away from identity as the absorption parameter increases. The spectral analysis of G_{an} is more complicated than that for the classical Radon transform. The results presented in table 2 show extreme singular values of G_{an} , i.e. eigenvalues of the symmetric operator $(G_{an}^* G_{an})^{1/2}$. For large values of the absorption map, several eigenvalues significantly depart from 1. In the case of constant absorption $a \equiv \mu$ on the unit disc, where the operator $G_{\mu n} \equiv G_{an}$ is self-adjoint, we show in table 3 below in section 6 that some eigenvalues of $G_{\mu n}$ even become negative. This implies that for intermediate values of the absorption parameter μ , zero can be an eigenvalue of $G_{\mu n}$, which is therefore no longer invertible. The good properties of the operator G_n (symmetric with positive eigenvalues) obtained in [2] for the Radon transform are therefore no longer true for the attenuated Radon transform.

There are two remedies to this instability. For sufficiently small absorption, where the singular values of G_{an} remain sufficiently far away from 0, conjugate gradient iterations (of $G_{an}^* G_{an}$) bring the spectrum closer to identity. The cost of the inversion with $p \geq 1$ iterations of CG is then roughly $4p + 3$ times more expensive than the cost of the inversion $(S^* H)_{an}$. When several singular values of G_{an} are close to 0, conjugate gradient iterations may not be sufficient (since $(G_{an}^* G_{an})$ may no longer be invertible). The remedy to the latter situation is zero-padding. Although we do not have any theoretical proof, numerical evidence shows that, as for the Radon transform, the inversion is substantially improved by zero-padding the image (hence also the absorption map) into a $2n \times 2n$ image, even at intensities of the absorption map for which many singular values of G_{an} are close to 0. The cost of this zero-padding is then roughly eight times the initial cost for an algorithm of order $O(n^3 \log n)$. For sufficiently large absorptions, zero-padding the image becomes necessary if one wants all the singular values of G_{an} to be far away from 0. Note however that zero-padding not only increases the computational cost but also requires additional measurements. Indeed, in the slant stack method, the number of directions is equal to $4n$ and the number of offsets equal to $2n$. Inversions with zero-padding require $8n$ directions θ_l and $4n$ offsets t_k . Zero-padding will prove quite useful in the iterative reconstructions from 180° measurements proposed in

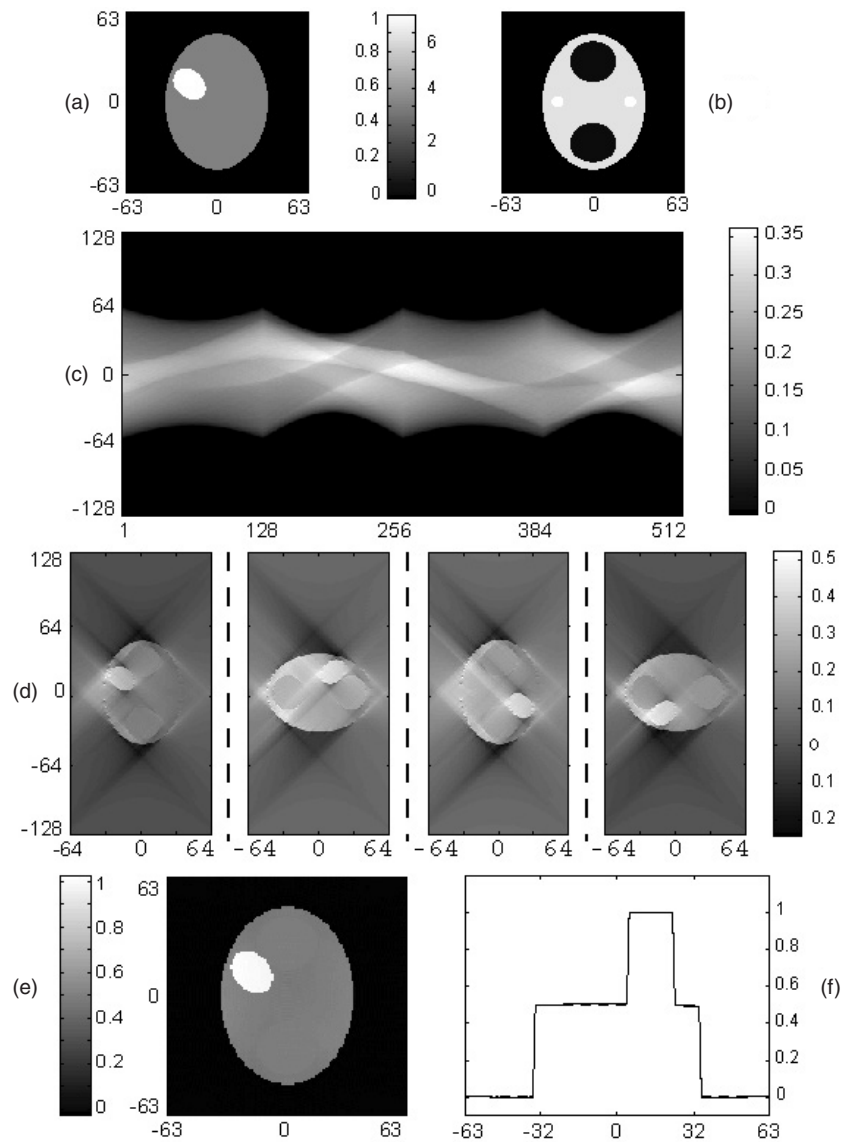


Figure 5. Numerical inversion of AtRT. Top line: image (left) and attenuation (right). Second line: AtRT as a function of t (vertical axis) and θ (horizontal axis); only one out of four measurement points is represented. Because of the zero-padding process into a $2n \times 2n$ image, the measurement space is of size $(2 \times 512) \times (2 \times 256)$. Third line: four partial reconstructions using data in the four domains Θ_k . Bottom line: sum of the previous four partial reconstructions and cross section along the line $x = -23$.

section 7. Figure 7 shows the singular values of G_{an} with $n = 16$ and the same singular values after zero-padding. By oversampling the image, the zero-padding technique allows us to obtain a much better approximation of identity.

At the level of absorption (a) in figure 4, all the singular values of G_{an} are sufficiently far away from 0. We thus do not need to use zero-padding. However to obtain visually almost perfect image reconstructions, we have used four iterations of conjugate gradient to invert the operator $G_{an}^* G_{an}$. The method is thus roughly 19 times more expensive than the method with

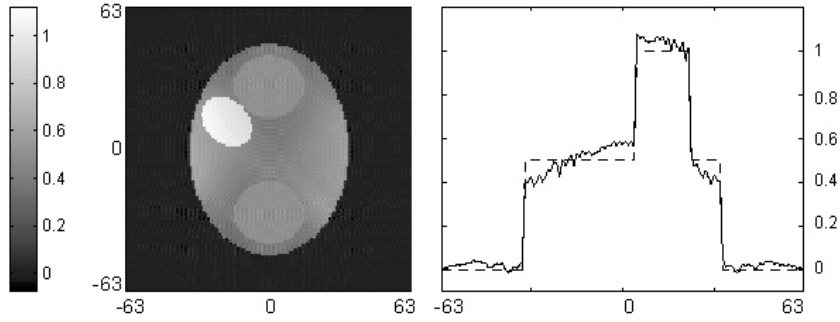


Figure 6. Numerical inversion of the same AtRT as in figure 5 except that no iterations of conjugate gradient have been used.

Table 2. Extremal singular values of G_{an} (three smallest and two largest) for different simulations: n is the number of pixels of the image; $ZP = 1$ corresponds to zero-padding into a $2n \times 2n$ image; CG is the number of conjugate gradient iterations; λ is the multiplicative factor in front of the absorption (a) in figure 4.

Case (λ, n, ZP, CG)	First	Second	Third	$n - 1$	Last
0.5, 16, 0, 0	0.8217	0.8240	0.8540	1.0927	1.3806
0.5, 16, 0, 4	0.9997	0.9998	0.9998	1.0001	1.0001
0.5, 16, 1, 0	0.9671	0.9675	0.9770	1.0225	1.0913
1, 16, 0, 0	0.3747	0.4922	0.5514	1.5186	1.6228
1, 16, 0, 4	0.7209	0.8743	0.9142	1.0146	1.0156
1, 16, 1, 0	0.8678	0.8717	0.9090	1.0633	1.0678
1, 32, 0, 0	0.3548	0.3972	0.5340	1.5317	1.9680
1, 32, 0, 4	0.6507	0.7086	0.8877	1.0162	1.0169
1, 32, 1, 0	0.8671	0.8891	0.8970	1.0701	1.0816

no conjugate gradient iteration. Though CG iteration may not be the fastest available method, it provides accurate reconstructions for moderate values of absorption.

Compatibility conditions. Let us conclude this section with a remark on compatibility conditions. As in the case of the Radon transform where $g(s, \theta) = g(-s, \theta + \pi)$, the measured data have a redundancy of order 2 [6, 28]. In order for the data to be the AtRT of a spatially dependent but angularly independent source term, they must satisfy a compatibility condition, which can be expressed as follows:

$$0 = \int_0^{2\pi} S_{-a, \theta}^* H_a g(\mathbf{x}) d\theta = \sum_{k=1}^4 S_{-a}^{k*} H_a g(\mathbf{x}). \quad (57)$$

This is similar to (50) with no differentiation. We have estimated the right-hand side in (57) numerically in the framework of figure 5. The relative L^2 error of the right-hand side compared to the L^2 norm of the data $g(s, \theta)$ is of the order of 4×10^{-3} , which shows that the slant stack algorithm verifies quite accurately the constraint (57). We have remarked that the compatibility condition was slightly better satisfied (by about 10%) for the smoothed absorption map.

6. Fast reconstruction

The Radon transform can be computed and inverted in $O(n^2 \log n)$ operations using the fast slant stack algorithm. This cost is $O(n^3 \log n)$ in general for the AtRT because of the absence

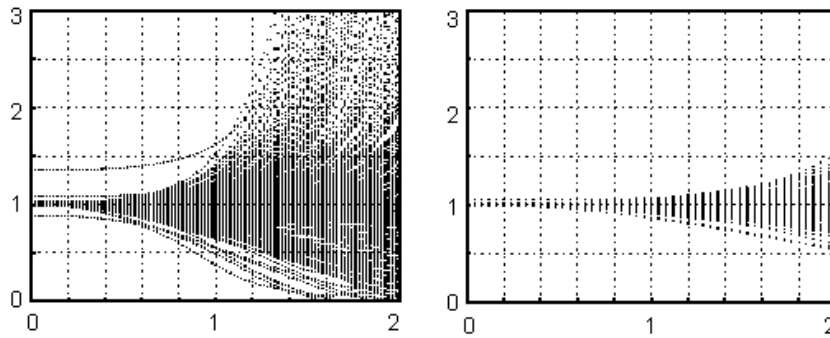


Figure 7. Singular values of the operator G_{an} (left) without zero-padding, (right) with zero-padding into a $2n \times 2n$ image.

of a Fourier slice theorem. In this section, we show that the calculation and inversion of the AtRT can be approximated by solving N slant stack transforms, where N is small compared to n in practice. The AtRT can thus be computed and inverted in $O(Nn^2 \log n)$ operations. We first present the algorithm on the simpler exponential Radon transform, which corresponds to a constant absorption coefficient.

6.1. Fast reconstruction for ERT

The ERT is defined as

$$T_\mu f(s, \theta) = \int_{\mathbb{R}} e^{\mu t} f(s\theta^\perp + t\theta) dt. \tag{58}$$

We can verify that $T_\mu = R_a$ when $a = \mu$ on the unit disc and $a = 0$ elsewhere [6, 18]. An exact inversion formula is based on

$$f(\mathbf{x}) = \frac{1}{4\pi} T_{-\mu}^* I_\mu^{-1} T_\mu f(\mathbf{x}), \tag{59}$$

where the adjoint operator T_μ^* and the generalized Riesz potential I_μ^{-1} are defined by

$$T_\mu^* g(\mathbf{x}) = \int_0^{2\pi} g(\mathbf{x} \cdot \theta^\perp, \theta) e^{\mu \mathbf{x} \cdot \theta^\perp} d\theta, \quad \widehat{I_\mu^{-1} g}(\sigma, \theta) = \begin{cases} |\sigma| \widehat{g}(\sigma, \theta), & |\sigma| > \mu \\ 0 & |\sigma| < \mu. \end{cases} \tag{60}$$

Note that we recover the notation of the Radon transform when $\mu = 0$. There is an extension of the Fourier slice theorem for the ERT. However it involves complex wavenumbers of the image $f(\mathbf{x})$ and its numerical use is difficult (see [26] for a nice application).

Let us now consider $\theta \in \Theta_1$ and introduce the slant stack transform as

$$S_\mu f(t, \theta) = \int_{\mathbb{R}} f(x, x \tan \theta + t) \exp\left(\mu \left(t \sin \theta + \frac{x}{\cos \theta}\right)\right) \frac{dx}{\cos \theta}. \tag{61}$$

We verify that $S_\mu f(t, \theta) = T_\mu f(t \cos \theta, \theta)$. Similarly the adjoint on Θ_1 is

$$S_\mu^{1*} g(\mathbf{x}) = \int_{\Theta_1} g(y - x \tan \theta, \theta) \exp(\mu(x \cos \theta + y \sin \theta)) \frac{d\theta}{\cos \theta}. \tag{62}$$

As for the AtRT, the contributions on Θ_k for $k = 2, 3, 4$ can be obtained using the same algorithm by appropriate rotations. The fast algorithm is based on the following expansion:

$$S_{\mu N} f(t, \theta) = e^{\mu t \sin \theta} \sum_{p=0}^N \frac{\mu^p}{p! \cos^p \theta} S[x^p f(x, y)](t, \theta). \tag{63}$$

Table 3. Extremal eigenvalues of $G_{\mu Nn}$ (three smallest and two largest eigenvalues) for different simulations: $n \times n$ is the number of pixels, μ is absorption, $ZP = 1$ corresponds to zero-padding into a $2n \times 2n$ image.

Case (μ, n, ZP)	First	Second	Third	$n - 1$	Last
1, 16, 0	0.1842	0.1842	0.9169	1.4434	2.1438
1, 16, 1	0.9838	0.9838	0.9912	1.0222	1.0392
3, 16, 0	-1.768	-1.768	0.7081	2.0247	4.6536
3, 16, 1	0.8950	0.9592	0.9592	1.0161	1.1125
3, 32, 0	-3.637	-3.637	0.3983	3.3206	7.0992
3, 32, 1	0.8896	0.9612	0.9612	1.0152	1.1207

We verify that $\lim_{N \rightarrow \infty} S_{\mu N} f = S_{\mu} f$, for instance in the $L^q(\mathbb{R} \times (0, 2\pi))$ sense for $1 \leq q \leq \infty$. The error after simple calculations using $|x^p| \leq 2^{-p}$ and $(\cos \theta)^{-p} \leq 2^{p/2}$ and the Stirling formula is then of the form

$$\left(\frac{(1 + \varepsilon)e\mu}{\sqrt{2N}} \right)^N,$$

for all $\varepsilon > 0$, which goes to 0 exponentially as $N \rightarrow \infty$. This shows that the ERT can be calculated in $O(Nn^2 \log n)$ operations with an error exponentially small in N .

Reconstruction. Let us now consider the discretization of the reconstruction formula (59). The generalized Riesz operator I_{μ}^{-1} is diagonal in the Fourier domain and can thus be estimated in $O(n^2 \log n)$ as in the case $\mu = 0$. It remains to solve T_{μ}^* or equivalently S_{μ}^* in the slant stack variables. This is done by introducing

$$S_{\mu N}^* g(\mathbf{x}) = \sum_{p=0}^N \frac{\mu^p x^p}{p!} S^* \left[\frac{e^{\mu t \sin \theta} g(t, \theta)}{\cos^p \theta} \right] (\mathbf{x}). \quad (64)$$

We observe that $S_{\mu N}^*$ is indeed the adjoint operator to $S_{\mu N}$ and that it converges to S_{μ}^* as $N \rightarrow \infty$. Moreover, as S^* , it can be calculated in $O(Nn^2 \log n)$ operations.

In discrete form, we thus introduce the operator

$$G_{\mu Nn} = S_{-\mu Nn}^* I_{\mu n}^{-1} S_{\mu Nn}, \quad (65)$$

where the operators with index n are the discrete version of the continuous operators using the fast slant stack algorithm presented in section 3. We can verify that $G_{\mu Nn}$ is still a symmetric operator as in the case $\mu = 0$. However, it need no longer have non-negative eigenvalues.

Numerical simulation. Although $G_{\mu Nn}$ remains a symmetric operator as for the Radon transform, its largest and smallest eigenvalues depart further and further from 1 as the absorption parameter increases; see table 3. Without zero-padding, the matrix $G_{\mu Nn}$ is not close to the identity matrix for $n = 16$. By zero-padding the image into a $2n \times 2n$ image, the accuracy of the reconstruction improves drastically.

It is interesting to compare the reconstruction accuracy depending on the number of terms in the expansion (64). We have chosen to calculate the ERT of the phantom in figure 3 with $\mu = 3$ and $N = 50$. In figure 8(a) we show the accuracy of the reconstructed phantom as the number of coefficients N increases. We see a very rapid convergence of the reconstructed solution to the solution reconstructed with $N = 50$ terms. However, in the absence of zero-padding, the latter solution is not very accurate. The fast solution with zero-padding converges quite rapidly to a good approximation of the true image.

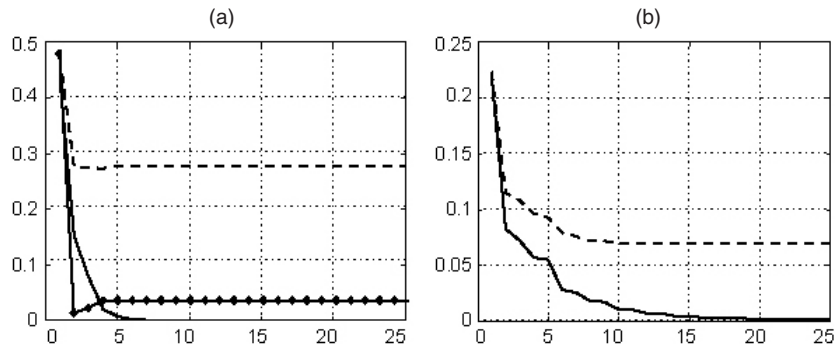


Figure 8. Quality of the reconstruction with an increasing number of terms in relative L^2 norm: (a) fast ERT with $\mu = 3$ and N terms, solid line: error in the reconstruction with $N = 50$ terms; dotted line: error of solution with zero-padding (into a $2n \times 2n$ image) in the original image; dashed curve: error (without zero-padding) in the original image. (b) Fast AtRT with N terms, solid line: error in the reconstruction obtained using the *slow* algorithm of section 5; dashed line: error in the original image.

6.2. Fast reconstruction for AtRT

The structure of the ERT and of the slant stack transform allows us to obtain an efficient algorithm based on (63). There does not seem to be any equivalent structure for the AtRT. Nevertheless one can still devise a fast AtRT provided that sufficient information about $e^{D_\theta a}$ is *pre-calculated*. When the absorption map $a(\mathbf{x})$ is represented as a $n \times n$ image, the cost of the pre-calculation is $O(n^3)$ operations. The term $e^{D_\theta a}$ indeed involves three variables and $O(n^3)$ unknowns once discretized and there is little hope that it can be estimated in less than $O(n^3)$ operations. We can also assume that the absorption map is represented as a $n^{2/3} \times n^{2/3}$ image. In that case, the pre-calculation of $e^{D_\theta a}$ can be obtained in $O(n^2)$ operations.

When $e^{D_\theta a}$ is sufficiently regular, it can be approximated quite accurately by a few terms in its Fourier series, defined by

$$e^{D_\theta a(\mathbf{x})} = \sum_{k \in \mathbb{Z}} w_k(\mathbf{x}) e^{ik\theta}, \quad \text{where} \quad w_k(\mathbf{x}) = \int_0^{2\pi} e^{D_\theta a(\mathbf{x})} e^{-ik\theta} \frac{d\theta}{2\pi}. \quad (66)$$

Using this decomposition we recast the AtRT as

$$S_a f(t, \theta) = \sum_{k \in \mathbb{Z}} e^{ik\theta} S[w_k(\mathbf{x}) f(\mathbf{x})](t, \theta). \quad (67)$$

The discrete approximation of $S[w_k(\mathbf{x}) f(\mathbf{x})](s, \theta)$ can be estimated in $O(n^2 \log n)$ calculations. Assuming that N coefficients are sufficient to obtain the expected accuracy in (66), we obtain a complexity of $O(Nn^2 \log n)$ to estimate the AtRT.

Fast inversion. The adjoint operator on Θ_1 in the reconstruction is given by

$$S_{-a}^* g(\mathbf{x}) = \sum_{k \in \mathbb{Z}} \tilde{w}_k(\mathbf{x}) S^*[e^{-ik\theta} g(t, \theta)](\mathbf{x}), \quad (68)$$

where

$$\tilde{w}_k(\mathbf{x}) = \int_0^{2\pi} e^{-D_\theta a(\mathbf{x})} e^{-ik\theta} \frac{d\theta}{2\pi}. \quad (69)$$

Provided that the weights \tilde{w}_k can be *pre-calculated* (fast or not depending on whether the absorption map is an $n^{2/3} \times n^{2/3}$ image or an $n \times n$ image) and that $e^{-D_\theta a}$ can be

well approximated by N Fourier coefficients, we obtain that S_{-a}^*g can also be estimated in $O(Nn^2 \log n)$ operations. To complete the inversion as in (50) requires the same procedure as in section 5 since the operators $\frac{\partial}{\partial y}$ and H_a can be performed in $O(n^2 \log n)$ operations.

Implementation. The numerical implementation of the above reconstruction formula is relatively straightforward. The weights $e^{\pm D_{\theta}a(\mathbf{x})}$ are approximated using $C_{\theta}A_{u,v}$ introduced in section 5. Note that the values of θ can be chosen arbitrarily in order to calculate the weights $w_k(\mathbf{x})$ that we approximate by $W_{k,u,v}$ in (66) and the weights $\tilde{w}_k(\mathbf{x})$ that we approximate by $\tilde{W}_{k,u,v}$ in (69). Since the functions are periodic in θ we have chosen a uniform discretization $\theta_j = 2\pi j/M$ for $j = 1, \dots, M$ and $M > N$ sufficiently large so that the integrals in (66) and (69) are calculated accurately. The only difficulty arises when one applies the operator $\frac{\partial}{\partial y}$. Since the latter is estimated in the Fourier domain, it requires that at least $m \times m$ coefficients be calculated accurately. Thus we need that $W_{k,u,v}$ and $\tilde{W}_{k,u,v}$ be accurately estimated for $u, v \in \mathcal{T}_m$, which requires taking $p = q = 3$ in the approximation of $C_{\theta}a$; see figure 1(c).

Note also that the Fourier coefficients can be calculated using a coarser spatial discretization of the absorption parameter. Assuming that the absorption is represented by a $n^{2/3} \times n^{2/3}$ image, the total cost of the integrating factor $e^{D_{\theta}a}$ and of the Fourier coefficients can then be performed in $O(n^2 \log n)$ operations. The total cost of the fast algorithm, including the pre-calculation of the Fourier coefficients, is thus of order $O(Nn^2 \log n)$.

Numerical simulation. We have implemented the fast reconstruction within the framework presented in section 5. The absorption map is given in (ii) of figure 4. The source terms are given in (a) and (b) of figure 4. For each source term, we perform three reconstructions from the same AtRT. The first calculation is performed with no noise. The second calculation is performed with 1% uniform noise added to the data multiplied by $e^{Sa(t,\theta)/2}$. The latter weight is included because our computational ‘data’ are $e^{Sa(t,\theta)/2}$ times larger than the physical data that would be measured in a real experiment as we noted at the beginning of section 5. In a third experiment the same noise level is added to the data and a Tikhonov regularization technique is used to smooth out the noise [31, 33] (by smoothing out the derivation at the end of the calculation in the last CG iteration; see (50)). The regularization parameter has been tuned to minimize the error in the reconstructed image. We do not pretend to perform an extensive analysis of noise reduction, and rather want to point out that ‘classical’ regularization techniques used in projection–back-projection or Fourier methods can also be applied here. We refer to [14, 15, 18, 21] for additional information about noise treatment in SPECT. Figures 9 and 10 present the results for the two different source terms. In both calculations, the simulation of the AtRT and of the reconstruction is performed with $N = 21$ terms. All the inversions are performed with four iterations of conjugate gradient to obtain a faithful inversion of the discrete AtRT S_{an} .

The reconstructions are quite satisfactory. To obtain more quantitative information about the fast algorithm we have performed an analysis of the matrix G_{an} for a value of $N = 11$ terms and a zero-padding of the image into an $2n \times 2n$ image. The spectral data of the method are very similar to what we observed in table 2. For the absorption map (ii) in figure 4 multiplied by a coefficient $\lambda > 0$, the extremal singular values of G_{an} in a few cases are given in table 4.

This shows that the spectrum of the fast AtRT algorithm behaves very similarly to that of the slow AtRT algorithm. It is also interesting to compare the reconstructions versus the number of Fourier coefficients used to represent $e^{D_{\theta}a}$. Starting with a forward calculation with $N = 25$ terms we compare in figure 8(b) the reconstruction for varying N . We also observe a relatively rapid convergence to the exact solution although this convergence is not as fast as for the ERT.

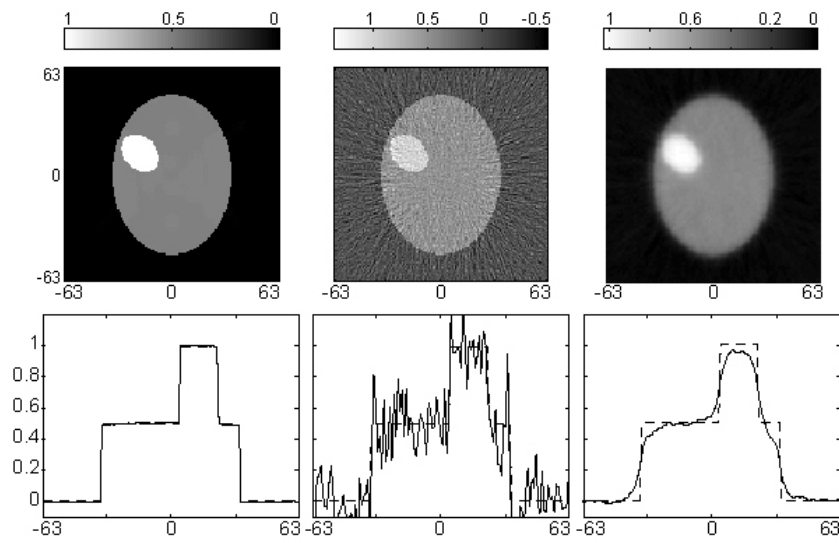


Figure 9. Fast AtRT with ellipsoids. First column: reconstruction with $N = 21$ terms. Second column: reconstruction with 1% noise on the data multiplied by $e^{Sa(t,\theta)/2}$. Third column: same noise level and Tikhonov regularization. The bottom figures present the cross section along $x = -23$.

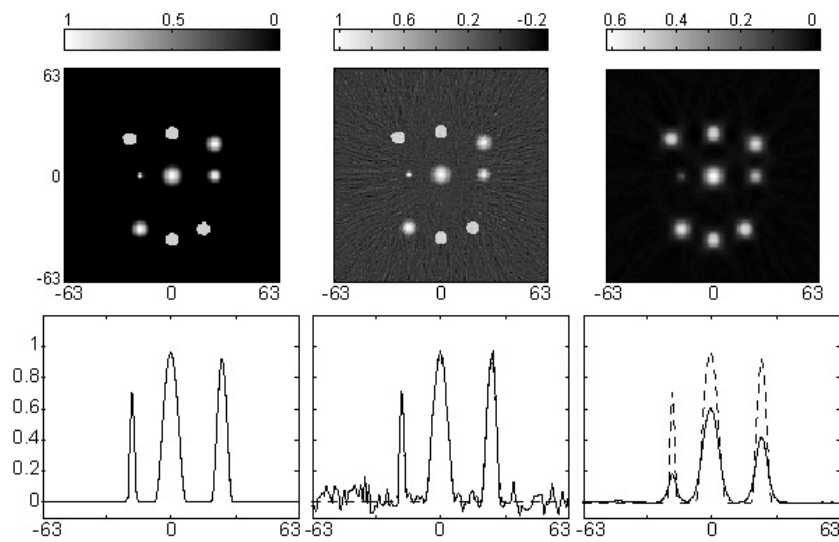


Figure 10. Fast AtRT with spots. First column: reconstruction with $N = 21$ terms. Second column: reconstruction with 1% noise on the data multiplied by $e^{Sa(t,\theta)/2}$. Third column: same noise level and Tikhonov regularization. The bottom figures present the cross section along $y = 0$.

7. Reconstruction from 180° measurements

Let us assume that the measured data $R_a f(s, \theta)$ are available for $\theta \in M = \Theta_1 \cup \Theta_2$ and not on $M^c = \Theta_3 \cup \Theta_4 = M + \pi$. We consider in this section the reconstruction of $f(\mathbf{x})$ from these measurements following the method proposed in [6]. Let us define the reconstruction

Table 4. Extremal singular values of G_{an} (three smallest and two largest) for different simulations: n is the number of pixels of the image; $ZP = 1$ corresponds to zero-padding into a $2n \times 2n$ image; CG is the number of conjugate gradient iterations; λ is the multiplicative factor in front of the absorption (ii) in figure 4.

Case (λ, n, ZP, CG)	First	Second	Third	$n - 1$	Last
0.5, 16, 1, 0	0.9820	0.9828	0.9841	1.0205	1.0559
1, 16, 0, 0	0.6330	0.6506	0.6867	1.1567	1.3903
1, 16, 0, 2	0.7546	0.7800	0.8975	1.0189	1.0541
1, 16, 0, 4	0.9960	0.9827	0.9925	1.0062	1.0068
1, 16, 1, 0	0.9278	0.9321	0.9332	1.0399	1.0561
1, 32, 1, 0	0.8987	0.9060	0.9065	1.0722	1.0619

operators

$$\begin{aligned} F_\theta &= F_{1,\theta} + F_{2,\theta}, & F_{1,\theta} &= R_{-a,\theta}^* \frac{\partial}{\partial s} H_a R_{a,\theta}, \\ F_{2,\theta} &= \left(\boldsymbol{\theta}^\perp \cdot \nabla R_{-a,\theta}^* - R_{-a,\theta}^* \frac{\partial}{\partial s} \right) H_a R_{a,\theta}. \end{aligned} \quad (70)$$

The reconstruction formula (40) is equivalent to $I = \frac{1}{2\pi} \int_0^{2\pi} F_\theta \, d\theta$. It is recast in [6] as

$$I = F^d + F^a + F^s. \quad (71)$$

The operators in (71) are defined as

$$\begin{aligned} F^d &= \frac{1}{2\pi} \int_M F_\theta \, d\theta + \frac{1}{2\pi} \int_{M^c} F_{1,\theta}^* \, d\theta \\ F^a &= \frac{1}{2\pi} \int_{M^c} (F_{1,\theta} - F_{1,\theta}^* + F_{2,\theta}^a) \, d\theta, & F^s &= \frac{1}{2\pi} \int_{M^c} F_{2,\theta}^s \, d\theta, \end{aligned} \quad (72)$$

with $F_{2,\theta}^s = \frac{1}{2}(F_{2,\theta} + F_{2,\theta}^*)$ and $F_{2,\theta}^a = \frac{1}{2}(F_{2,\theta} - F_{2,\theta}^*)$. Because

$$F_{1,\theta}^* = R_{a,\theta}^* H_a \frac{\partial}{\partial s} R_{-a,\theta}, \quad (73)$$

and $R_{-a,\theta} f(s) = R_{a,\theta+\pi} f(-s)$, we verify that the operator F^d depends on the measured data $R_{a,\theta} f(s)$ for $\theta \in M$. We thus define $d(\mathbf{x}) = F^d f(\mathbf{x})$. The operator F^a is bounded in $L^2(B)$, where B is the unit ball in which the support of f is assumed to be included, and skew-symmetric. The operator F^s is self-adjoint and compact.

Iterative reconstruction. The reconstruction is then performed as follows [6]. Let us first assume that $F^s = 0$, as is the case when a is constant on the unit disc and vanishes outside. We aim to solve

$$f(\mathbf{x}) = d(\mathbf{x}) + F^a f(\mathbf{x}). \quad (74)$$

This is done iteratively (with $f^0 = 0$ unless a better initial guess is available) as

$$f^{k+1}(\mathbf{x}) = \gamma d(\mathbf{x}) + [(1 - \gamma)I + \gamma F^a] f^k(\mathbf{x}), \quad (75)$$

where $\gamma = (1 + \|F^a\|_2^2)^{-1}$, with $\|F\|_2$ the $L^2(B)$ norm of an operator F . The norm of the operator $[(1 - \gamma)I + \gamma F^a]$ is given by $\|F^a\|_2 / \sqrt{1 + \|F^a\|_2^2} < 1$. This ensures the convergence of f^k to the solution f of (74).

In the general case, F^s does not vanish and we aim to solve

$$f(\mathbf{x}) = d(\mathbf{x}) + F^a f(\mathbf{x}) + F^s f(\mathbf{x}). \quad (76)$$

It is shown in [6] that an iterative reconstruction is possible provided that $\|F^s\|_2 < 1$. When there exists a value of γ such that $\|(1 - \gamma)I + \gamma(F^a + F^s)\|_2 < 1$ we can use the same algorithm as above with F^a replaced by $F^a + F^s$. In all practical cases considered in this paper, we have observed that we could use this simple algorithm with $\gamma = 1$ since the radius $F^a + F^s = I - F^d$ was always found to be less than 1.

Let us mention that when no such γ exists and we have $\|F^s\|_2 < 1$, we can still solve (76) by posing

$$\begin{aligned} f(\mathbf{x}) &= (I - F^s)^{-1/2}h(\mathbf{x}) \\ h(\mathbf{x}) &= (I - F^s)^{-1/2}d(\mathbf{x}) + (I - F^s)^{-1/2}F^a(I - F^s)^{-1/2}h(\mathbf{x}). \end{aligned} \tag{77}$$

The equation for h can be solved iteratively as it has the form (74) since $(I - F^s)^{-1/2}F^a(I - F^s)^{-1/2}$ is skew-symmetric. Since $I - F^s$ has a real valued and positive spectrum, the operator $(I - F^s)^{-1/2}$ exists and admits the converging (in the $L^2(B)$ sense) expansion $(I - F^s)^{-1/2} = \sum_{k=0}^{\infty} 4^{-k} \binom{2k}{k} (F^s)^k$. Solving for $f(\mathbf{x})$ in (77) is therefore feasible although expensive numerically. We do not consider this method further here.

Implementation. Compared to the reconstruction from full measurements, the solution of (76) requires discretization of the operators $F_{1,\theta}$, $F_{2,\theta}$, and their adjoints. In the variables (t, θ) , the operator F^1 consisting of the integration only on Θ_1 is given by $F^1 f(\mathbf{x}) = S_{-a}^{1*} H_a S_a f(\mathbf{x})$, where S_a^{1*} is defined in (51). As in earlier sections, the integration over Θ_k for $k = 2, 3, 4$ is dealt with similarly using appropriate rotations. The operator can be decomposed as $F^1 = F_1^1 + F_2^1$, which is the integration of the first line of (70) over Θ_1 . In the variables (t, θ) these operators take the form

$$\begin{aligned} F_1^1 f(\mathbf{x}) &= \int_{\Theta_1} S_{-a,\theta}^* \frac{\partial}{\partial t} H_a S_a f(\mathbf{x}) d\theta, \\ F_2^1 f(\mathbf{x}) &= \int_{\Theta_1} \left(\frac{\partial}{\partial y} e^{-D_\theta a(\mathbf{x})} \right) H_a S_a f(y - x \tan \theta, \theta) \frac{d\theta}{\cos \theta}. \end{aligned} \tag{78}$$

The operator F_1^1 and its adjoint F_1^{1*} pose no specific implementation difficulty. The only difference with respect to F is that the derivation $\frac{\partial}{\partial t}$ is now taken just before or just after the operator H_a . The implementation of F^d in (72) to calculate $d(\mathbf{x})$ is therefore very similar to earlier implementations in sections 5 and 6.2. The operator F_2^1 is also very similar. We have to make sure that $e^{-D_\theta a(\mathbf{x})}$ is sufficiently well approximated so that we can take its derivative in the variable y on an array of size $n \times m$ to use the algorithm presented in section 5 and $m \times m$ for the algorithm in section 6.2. This requires choosing $p = 2$ and $q = 3$ and $p = 3$ and $q = 3$, respectively, in the approximation of $C_\theta a$ calculated in section 5. The adjoint operator F_2^{1*} is defined by

$$F_2^{1*} f(\mathbf{x}) = \int_{\Theta_1} S_{a,\theta}^* H_a S \left[\left(\frac{\partial}{\partial y} e^{-D_\theta a(\mathbf{x})} \right) f(\mathbf{x}) \right] (s, \theta) d\theta. \tag{79}$$

Its implementation is quite similar to that of the operator F . Note that the fast implementation of F_2^1 and F_2^{1*} as in section 6.2 requires replacing the coefficients w_k and \tilde{w}_k by their derivatives in the variable y .

Similarly to what we obtained in earlier sections, the cost of a single iteration is $O(Nn^2 \log n)$ for the AtRT. The cost of the full algorithm will then be of order $O(\mathcal{N}Nn^2 \log n)$, where \mathcal{N} is the total number of iterations. The latter number depends on the spectral radius $\rho(F^a + F^s)$ or $\rho(F^s)$. In the applications considered in this paper, the radius $\rho = \rho(F_a + F_s) < 1$, which implies an accuracy of the reconstruction after \mathcal{N} steps of order $O(\rho^{\mathcal{N}})$.

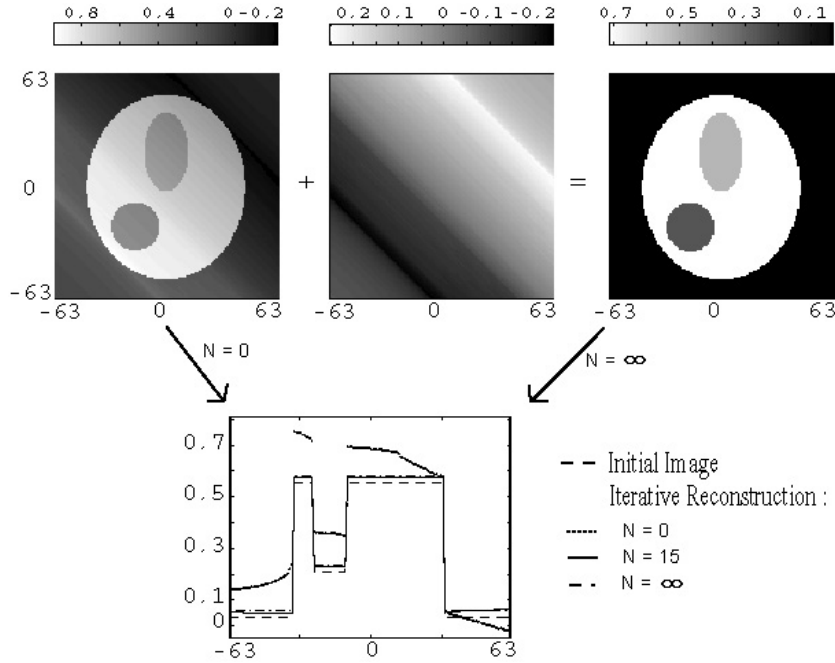


Figure 11. ERT reconstruction with fast algorithm ($N = 20$) and zero-padding $ZP = 1$ (except for the calculation of $d(\mathbf{x})$) after $\mathcal{N} = 0$ and $\mathcal{N} = 15$ iterations and at convergence. The bottom figure represents a cross section along $x = -23$.

7.1. Numerical simulation for ERT

In the case where $a = \mu$ is constant on the unit disc and 0 elsewhere, we can show [6] that $F_\theta^* = F_{\theta+\pi}$ and that $F_2 \equiv 0$. We are thus in the situation where

$$F^d = \frac{1}{\pi} \int_M F_\theta d\theta, \quad F^a = \frac{1}{2\pi} \int_{M+\pi} (F_\theta - F_{\theta+\pi}) d\theta. \tag{80}$$

With different notation, this is the procedure implemented in [26, 29]. All we have to do is to multiply the data we have on M by 2 and to subtract what we have added as a contribution on $M + \pi$. Since the operator F^a is skew-symmetric, we are in the situation where we need to solve for $f(\mathbf{x})$ in (74) with $d(\mathbf{x}) = F^d f(\mathbf{x})$ can be reconstructed from the measured data.

Numerical results. We have considered the benchmark proposed in [26]. The source term is reproduced in figure 11. The absorption parameter is $\mu = 3$ and the number of terms in the reconstruction (and simulation of the synthetic data) is $N = 20$. As we saw in table 3 the eigenvalues of $G_{\mu N n}$ depart substantially from 1 in the absence of zero-padding. It is important to ensure that no spurious numerical modes arbitrarily increase the radius of the operator F^a in (80) if one wants to use a physical value of γ in (75). We have therefore performed the iterations of the algorithm with zero-padding ($ZP = 1$) so that the operators act on images of size $2n \times 2n = 256 \times 256$. The calculation of $d(\mathbf{x})$ has however been obtained *with no* zero-padding though four iterations of CG have been used. It is thus calculated from measurements on 256 directions (for half of the measurements) and 256 offsets and *not* four times that amount. Once $d(\mathbf{x})$ is calculated, the operator F^a is calculated on $2n \times 2n$ images. By doing so, we have obtained a spectral radius of the discretization of F^a of roughly 0.8, which allows us to choose (the non-optimal but very convenient) $\gamma = 1$. The spectral radius

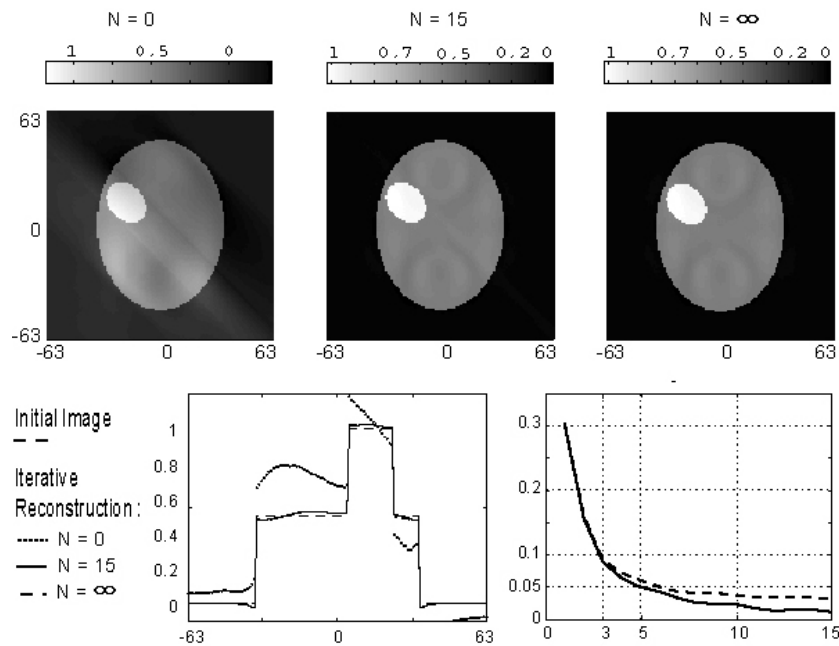


Figure 12. AtRT reconstruction from partial measurements with fast algorithm ($N = 21$) and zero-padding $ZP = 1$ (except for the calculation of $d(\mathbf{x})$). The images after $\mathcal{N} = 0$ and $\mathcal{N} = 15$ iterations and at convergence are represented. Bottom: (left) cross section along $y = -23$ and (right) relative L^2 error compared to the true image (dashed line) and compared to the solution at convergence (solid line).

of the discretization of F^a without zero-padding is above 1 and requires choosing $\gamma < 1$ as in [26]. Figure 11 displays the image reconstructed from data on $M = \Theta_1 \cup \Theta_2$ after one iteration, corresponding to $\mathcal{N} = 1$ in (75), and the image after $\mathcal{N} = 15$ iterations.

7.2. Numerical simulation for AtRT

In the examples of source and absorption maps presented earlier in the paper, we have observed that the spectral radius of the discretization of the operator $F^a + F^s$ always remained below 1 provided that zero-padding was used in their estimation. This is to be compared to the results shown in figure 7 and implies that we can use the algorithm (75) (with F^a replaced by $F^a + F^s$) with $\gamma = 1$. As before the calculation of the approximation of $d(\mathbf{x})$ is obtained *without* zero-padding the image. We have observed that the radius of $F^a + F^s$ becomes greater than 1 when no zero-padding is used. The main advantage of zero-padding the image is that it avoids the purely numerical increase of the radius of the operators involved in the iterative technique. Let us mention that the radii of these operators become larger than 1 for sufficiently large absorptions independently of the discretization. We do not consider this case further here.

Let us consider the first source term (a) and the regularized absorption map (ii) in figure 4. The reconstruction after $\mathcal{N} = 0$ and $\mathcal{N} = 15$ is compared to the true image in figure 12. We observe that the discontinuities of the source term are well reproduced after the first iteration. However the values of the intensity are poorly reconstructed. The solution after $\mathcal{N} = 5$ iterations however is almost perfect and visually perfect after $\mathcal{N} = 15$ iterations. The

convergence of the iterative procedure to the true image as \mathcal{N} increases is represented in figure 12.

8. Conclusions

We have extended the fast slant stack algorithm to the calculation of the exponential Radon transform and the attenuated Radon transform. The calculation is accurate and can be performed in $O(Nn^2 \log n)$ operations, where N depends on the smoothness of a and the required accuracy, provided that the Fourier coefficients of $e^{D_0 a}$ are pre-calculated. The pre-calculation can also be performed in $O(n^2 \log n)$ operations when the sufficiently smooth absorption map is represented as a $n^{2/3} \times n^{2/3}$ image. This is automatically the case for the ERT.

We have then used the Novikov formula and results in [6] to address the reconstruction of the source term from its ERT or AtRT with full or partial measurements. The slant stack algorithm has been shown to provide robust and accurate reconstructions provided that conjugate gradient iterations and possibly zero-padding techniques are used. Under the assumption that the Fourier coefficients of $e^{-D_0 a}$ are pre-calculated or that the absorption map is represented as a $n^{2/3} \times n^{2/3}$ image, the reconstruction can also be performed in $O(Nn^2 \log n)$ operations.

How large N has to be in practical reconstructions remains to be determined. Assuming that N is not too large, the present algorithm offers an interesting technique to invert the AtRT quite rapidly. A practical determination of N will also depend on the noise level in the measurements (the higher the noise level, the smaller N in some sense since the accuracy of the absorption map matters less). How the method behaves as the noise level increases remains to be investigated.

Acknowledgments

This work was supported in part by NSF grant DMS-0239097. The first author also acknowledges support from an Alfred P Sloan fellowship.

References

- [1] Arbuzov E V, Bukhgeim A L and Kazantsev S G 1998 Two-dimensional tomography problems and the theory of A —analytic functions *Sib. Adv. Math.* **8** 1–20
- [2] Averbuch A, Coifman R R, Donoho D L, Israeli M and Waldén J 2001 Fast slant stack: a notion of Radon transform for data in a Cartesian grid which is rapidly computable, algebraically exact, geometrically faithful and invertible *Preprint*
- [3] Averbuch A, Coifman R R, Donoho D L, Israeli M and Waldén J 2003 The pseudopolar FFT and its applications *Preprint*
- [4] Axel L, Herman G T, Roberts D A and Dougherty L 1990 Linogram reconstruction for magnetic resonance imaging (MRI) *IEEE Trans. Med. Imaging* **9** 447–9
- [5] Bal G 2001 Fourier analysis of the diamond discretization in particle transport *Calcolo* **38** 141–72
- [6] Bal G 2004 On the attenuated Radon transform with full and partial measurements *Inverse Problems* **20** 399–419
- [7] Basu S and Bresler Y 2000 $O(N^2 \log_2 N)$ filtered backprojection reconstruction algorithm for tomography *IEEE Trans. Image Process.* **9** 1760–73
- [8] Boag A, Bresler Y and Michielssen E 2000 A multilevel domain decomposition algorithm for fast $O(N^2 \log N)$ reprojection of tomographic images *IEEE Trans. Image Process.* **9** 1573–82
- [9] Brandt A, Mann J, Brodski M and Galun M 2000 A fast and accurate multilevel inversion of the Radon transform *SIAM J. Appl. Math.* **60** 437–62
- [10] Bukhgeim A A and Kazantsev S G 2002 Inversion formula for the fan-beam attenuated Radon transform in a unit disk *Sobolev Institute of Mathematics Preprint* N 99

- [11] Edholm P, Herman G T and Roberts R A 1988 Image reconstruction from linograms: implementation and evaluation *IEEE Trans. Med. Imaging* **7** 239–46
- [12] Fessler J A and Sutton B P 2003 Nonuniform fast Fourier transforms using min–max interpolation *IEEE Trans. Signal Process* **51** 560–74
- [13] Golub G H and van Loan C F 1996 *Matrix Computations* (Baltimore, MD: Johns Hopkins University Press)
- [14] Guillement J-P, Jauberteau F, Kunyansky L, Novikov R G and Trebossen R 2002 On single-photon emission computed tomography imaging based on an exact formula for the nonuniform attenuated correction *Inverse Problems* **18** L11–9
- [15] Guillement J-P and Novikov R G 2004 A noise property analysis of single-photon emission computed tomography data *Inverse Problems* **20** 175–98
- [16] Heike U 1986 Single-photon emission computed tomography by inverting the attenuated Radon transform with least-squares collocation *Inverse Problems* **2** 307–30
- [17] Kunyansky L A 1992 Generalized and attenuated Radon transforms: restorative approach to the numerical inversion *Inverse Problems* **8** 809–19
- [18] Kunyansky L A 2001 A new SPECT reconstruction algorithm based on the Novikov’s explicit inversion formula *Inverse Problems* **17** 293–306
- [19] Lewis E E and Miller W F Jr 1984 *Computational Methods of Neutron Transport* (New York: Wiley)
- [20] Mathews K A 1999 On the propagation of rays in discrete ordinates *Nucl. Sci. Eng.* **132** 155–80
- [21] Metz C E and Pan X 1995 A unified analysis of exact methods on inverting the 2-D exponential Radon transform *IEEE Trans. Med. Imaging* **14** 643–58
- [22] Natterer F 1986 *The Mathematics of Computerized Tomography* (New York: Wiley)
- [23] Natterer F 1999 Numerical methods in tomography *Acta Numer.* **8** 107–41
- [24] Natterer F 2001 Inversion of the attenuated Radon transform *Inverse Problems* **17** 113–9
- [25] Natterer F and Wübbeling F 2001 *Mathematical Methods in Image Reconstruction (SIAM Monographs on Mathematical Modeling and Computation)* (Philadelphia, PA: SIAM)
- [26] Noo F and Wagner J-M 2001 Image reconstruction in 2D SPECT with 180° acquisition *Inverse Problems* **17** 1357–71
- [27] Novikov R G 2002 An inversion formula for the attenuated x-ray transformation *Ark. Math.* **40** 145–67 (Rapport de Recherche 00/05–3, Université de Nantes, Laboratoire de Mathématiques)
- [28] Novikov R G 2002 On the range characterization for the two-dimensional attenuated x-ray transformation *Inverse Problems* **18** 677–700
- [29] Pan X, Kao C-M and Metz C E 2002 A family of π -scheme exponential Radon transforms and the uniqueness of their inverses *Inverse Problems* **18** 825–36
- [30] Radon J 1917 Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten (Berichte Sächsische Akademie der Wissenschaften, Leipzig) *Math. Phys. Kl.* **69** 262–7
- [31] Tikhonov A V and Arsenin V Y 1977 *Solutions of Ill-Posed Problems* (New York: Wiley)
- [32] Tretiak O and Metz C 1980 The exponential Radon transform *SIAM J. Appl. Math.* **39** 341–54
- [33] Vogel C R 2002 *Computational Methods for Inverse Problems (SIAM Frontiers in Applied Mathematics)* (Philadelphia, PA: SIAM)