Development of optical tomography methods with discretized path integral (離散化経路積分による散乱トモグラフィー)

Yuan Bingzhi

Supervisor: Dr. Toru Tamaki
Dr. Kazufumi Kaneda
Dr. Takio Kurita
Dr. Bisser Raytchev

Visual Information Science Laboratory
Department of Information Engineering
Graduate Schoole of Engineering
Hiroshima University

This thesis submitted in fulfilment of the requirements for the degree of
Doctor of Engineering

Hiroshima University January 2018

This thesis is dedicated to my loving parents and families.
For their love, support and encouragement during the hard time.

Declaration

I, Bingzhi Yuan, declare that this thesis titled, "Development of optical tomography methods with discretized path integral" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help
- Where the thesis is based on work done by myself jointly with others, I gave made clear exactly what was done by others and what I have contributed myself.

BingzhiYuan
January 2018

Acknowledgements

This dissertation would not have been possible without the guidance and the help of several individuals who in one way or another contributed and extended their valuable assistance in the preparation and completion of this study.

First and foremost, my utmost gratitude to my supervisor Dr. Toru Tamaki, whose sincerity and encouragement I will never forget. Dr. Toru Tamaki has inspired me as as I hurdle all the obstacles in the completion of my PhD life.

Also, my sincere gratitude to my sub-supervisors, Dr. Kazufumi Kaneda, Dr. Takio Kurita and Dr. Bisser Raytchev, who provide me many useful and important suggestions on my research and this dissertation. Their unselfish and unfailing support will help me all my research life.

And my sincere gratitude to Dr. Yasuhiro Mukaigawa, Dr. Hiroyuki Kubo and Mr. Takahiro Kushida, who are the co-authors of the papers included in this thesis.

And then, my sincere gratitude to Dr. Tsubasa Hirakawa, who give me help in technical paper writing. My gratitude to my laboratory members. Thanks Dr. Marcos Slomp and Dr. Michihiro Mikamo for sharing variant new ideas with me. Thanks Mr. Fangda Zhao for the discussion of my research.

This research is supported in part by a grant from the Japan Society for the Promotion of Science (JSPS) through the "Funding Program for Next Generation World-Leading Researchers (NEXT Program)" initiated by the Council for Science and Technology Policy (CSTP), and by JSPS KAKENHI Grant Number 26280061.

I also would like to thank the open source initiatives of EIDORS[1] for their generosity and excellence in their open source contributions that provide the comparison results contained in this thesis, as well as the writing of this document. Last but not the least, my family and my parents, for all your kindness, understanding and supports, during the hard time. Without you, I will not have confidence to go abroad and finish my study. Thank you very much.

[1] http://eidors3d.sourceforge.net/

Abstract

Recent the demand of a safe medical tomography is increasing dramatically, as the radiation issue of X-ray CT and the limited working area of NMRI. Optical tomography is known for its safeness. Therefore, an optical tomography method to estimate the inside of an object seems to be useful and important.

In this thesis, we present a forward model for optical tomography based on a path integral. Instead of directly solving the radiative transport equations, which has been widely used in optical tomography, we use a path integral that has been developed for rendering participating media based on the volume rendering equation in computer graphics. After creating a discretized two-dimensional layered grid, we obtained the forward model and the inverse problem for the optical tomography.

We develop an algorithm to estimate the extinction coefficients of each voxel with an Log-barrier interior point approach. Numerical simulation results are shown to demonstrate that the proposed method works well.

In order to improve the performance, we propose the Primal-dual approach to solve the inverse problem formulated as a constraint optimization problem. We further develop new efficient formulations for computing Jacobian and Hessian of the cost function of the constraint non-linear optimization problem. Numerical experiments show that the combination of Primal-dual approach and the new efficient formulations can reduce the computation time than the Log-barrier interior point approach while keeping the quality of the estimation results.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

Optical tomography[10, 8, 33, 7, 9, 12, 58, 18] is known as a safer alternative to X-ray tomography. Usually tomography consists of a light source generating penetrative light and a detector capturing the light, which allows to estimate the inside of an object in which the light is passing through. The most important application is X-ray Computed Tomography (CT) where X-rays are used due to their penetrative property. The balance between the radiation exposure of the human body and the quality of the obtained results has been debated since the early days when X-ray CT was invented. Therefore, there is an urgent demand for a safer medical tomography, such as optical tomography.

Modeling the behavior of light plays an important role in optical tomography, and in the mesoscale, in which the wavelength of light is close to the scale of tissue, the *Radiative Transport Equation* (RTE) is used for describing the behavior of light scattering[9, 43]. At the macroscale[12], the time-independent or dependent RTE is often approximated with a diffusion equation.

Similarly, the computer graphics community has the used time-independent RTE, and in contrast to the (surface) rendering equation[39, 41], often call it the *volume rendering equation* (VRE)[39, 53]

$$(\omega \cdot \nabla)L(x,\omega) = -\sigma_t(x)L(x,\omega) + \sigma_s(x)\int_{S^2} f_p(x,\omega,\omega')L(x,\omega')d\omega', \tag{1.1}$$

where notation will be introduced in the following sections. The use of VRE enables us to render volumes of participating media such as fog, cloud, and fire through which light is penetrating, and to obtain realistic volume-rendering images of such scenes[17, 45]. The *path integral*, which can be considered as a discrete version of the continuous Feynman path integral [26, 66], has been recently employed to solve the VRE in an efficient way

with Monte Carlo integration such as Metropolis light transport[52, 68] or bidirectional path tracing[47].

1.1 Thesis Overview

In this thesis, we propose an optical tomography method using path integral as a forward model and solving a non-linear inverse problem that minimizes the discrepancy between measurements and model predictions in a least-squares sense.

In the forward model, we simplify the path integral with some assumptions. The path integral, as the name suggests, gathers (or integrates) the contributions of all possible paths of light[52, 68, 67, 38, 63, 44]. We approximate the integral of infinite number of paths with a sum of finite number of paths, and discretize a continuous medium into voxels of a regular grid, and continuous light paths into discrete ones (i.e., polylines). We deal with anisotropic scattering having a peak in the forward direction, which is different from other discretization methods using discrete ordinate or spherical harmonics[17, 1, 59]. Then we further focus on estimating the spatially varying extinction coefficient $\sigma_t(x)$ at each discretized voxel location of the medium while fixing scattering properties (e.g., scattering coefficients σ_s and phase functions f_p). By separating the scattering properties from our problem, we formulate optical tomography as an optimization problem with inequality constraints.

To solve the optimization problem with inequality constraints, we implemented two types of approaches, the Log-barrier Interior Point approach and the Primal-dual Interior Point approach. These two types of approaches were tested by the numerical simulation and obtained fair results.

Till here, we have described the optical tomography method we proposed. In next section, we will introduce the our contribution and the thesis structures.

1.2 Thesis Structures

In this thesis, we proposed the forward problem using discretized path integral which has not been used in optical tomography before. And two types of approaches to solve the inverse problem obtained from the forward problem. The structures of the thesis is as follows:

In Chapter 1, we have introduced the motivation of our research.

In Chapter 2, we will introduce the previous works both related to optical tomography and the Statistical Monte Carlo methods.

In Chapter 3, we will show how to use path integral method to present the mathematical model of the light transport in the optical tomography, and simplify the expression with

reasonable constraints. The simplified model presented in this chapter is used by "Optical tomography with discretized path integral" [71] and "Layered optical tomography of multiple scattering media with combined constraint optimization"[72].

In Chapter 4, we will describe the process of using interior point approach to solve the optimization problem which is obtained from the simplified model in Chapter 3. The works described in Chapter 4 were published in the paper "Optical tomography with discretized path integral" [71] and "Layered optical tomography of multiple scattering media with combined constraint optimization"[72].

Chapter 5 shows the primal-dual approach which solve the same optimization problem in Chapter 4. We will also introduce the tricks used in the approach which reduce the computational cost dramatically. The works described in Chapter 5 were published in the paper "Primal-dual approach to optical tomography with discretized path integral with efficient formulations"[73] and "Optical tomography with discretized path integrals: a comparison with log-barrier and primal-dual methods"[70].

Chapter 6 presents our conclusions.

Chapter 2

Related Work

In this section, we briefly review related work on optical tomography and path integrals in computer graphics.

Optical tomography[7, 9] (or inverse transport[12, 58], inverse scattering[49], scattering tomography[27, 28]) is a problem in medical imaging using light sources to reconstruct the optical properties of tissue from measurements (time-dependent or stationary, angular-dependent or independent) at the surface boundary. Solving the RTE (1.1) with boundary conditions analytically is however difficult, and approximations, such as discrete ordinates and Nth order spherical harmonics (P_N approximation), are often used and solved numerically by, for example, finite element methods (FEM) or finite difference methods (FDM). The famous diffuse approximation[9, 12] (DA) is a P_1 (thus 1st order) approximation with the assumption on a phase function being isotropic. The DA is an approximation to RTE at macroscopic scale when scattering is large while absorption is low, and scattering is not highly peaked. Diffuse Optical Tomography (DOT) is based on DA and nowadays represents the frontier of optical tomography[62, 15] with many clinical applications[29]. The DA however does not often hold in realistic participating (scattering) media; absorption may not be small compared to scattering, and the shapes of the phase functions can be highly peaked in the forward direction which is often modeled by Henyey-Greenstein[35], Schlick[14] or Mei and Rayleigh phase functions[20, 50, 53, 39]. Experimental evidence [42] also suggests a highly peaked shape of the phase functions in biological media. DOT works but is still limited, therefore, other methods have been also studied for cases when DA does not hold.

Statistical Monte-Carlo methods are used for media in which the assumptions do not hold, however they are computationally intensive and inefficient for solving the forward problem [7, 9, 12, 58, 29], i.e., solving the RTE with given parameters. Therefore Monte-Carlo based approaches have been used for estimating the spatially constant (not varying) parameters in homogeneous media such as paper[24, 25], clouds[21], liquids[40], plastics[11], or uniform

material samples[30]. Another difficulty of Monte-Carlo based inverse methods is that an analytical forward model prediction is hard to obtain when we want to minimize the difference between the prediction and measurements except for very special structures[5, 48]. A gradient based least square approach has been proposed but only for spatially constant parameter estimation[23–25], while model-free approaches have relied on genetic algorithms[21, 11], numerical perturbation[32, 31], voting[36], or even simple back-projection[13]. One of the contributions of the current paper is to enable us to use a gradient based optimization approach for estimating spatially varying parameters, which is extensible by using many optimization methods.

Similar to optical tomography, modeling light transport plays very important role in computer graphics. Our own work on optical tomography is inspired by Monte-Carlo based statistical methods. In the last two decades, methods based on path integrals[57, 52, 68, 47, 56, 46] have provided models of light transport for efficient volume rendering. For solving RTE, a path integral has been used for a forward problem solver[66, 37, 69], and also applied to optical tomography but under the diffusion assumption[60, 61]. Our proposed method is based on a path integral to express the forward model prediction explicitly, which is very suitable to solve the inverse problem with gradient-based methods. This is an advantage of our method over existing methods because the paths used in the forward model can be generated by either a deterministic or statistical (Monte-Carlo) method. To achieve an efficient forward model, we introduce a simplified layered scattering model that uses a limited number of deterministic paths instead of Monte-Carlo simulated ones.

Chapter 3

Forward problem in the optical tomography

As mentioned in the Chapter 1, in the optical tomography method we proposed, we solve a non-linear inverse problem that minimizes the discrepancy between measurements and model predictions in a least-squares sense. Therefore, we deal with the following optical tomography problem[1]

$$\min_{\boldsymbol{\sigma}_t} \sum_{i,j} \left| I_{ij} - P_{ij}(\boldsymbol{\sigma}_t) \right|^2, \tag{3.1}$$

where $\boldsymbol{\sigma}_t$ is a vector representing the spatial distribution of the extinction coefficients to be estimated. We divide our discussion into two parts; forward and inverse problems. The forward problem focuses on building a mathematical model $P_{ij}(\boldsymbol{\sigma}_t)$ of the light transport between a light source i and a detector j. We will make some assumptions on the light transport and the medium to simplify the forward model. An inverse problem minimizes the difference between the observations I_{ij} of the detector and the forward model to estimate the spatial distribution of the extinction coefficients $\boldsymbol{\sigma}_t$.

3.1 Forward model

In the forward problem, as we mentioned before, we use a path integral to build a mathematical model for the light transport. Here, we follow the notation developed in the computer graphics literature [57, 52, 22, 44] to introduce the path integral. The next section will show the simplified model we propose.

[1] This is a conceptual formulation and the actual problem is shown in Eq. (4.2).

Given a space $\Re^3$, there are a light source located at $x_0 \in \Re^3$and a detector at $x_{M+1} \in \Re^3$, and in-between participating media $v \subset \Re^3$ with boundary ∂v and interior volume $v_0 := v \setminus \partial v$. A light path $\tilde{x}$ connecting x_0 and x_{M+1} of length $M+2$ consists of $M+2$ vertices $x_m \in \Re^3$ for $m = 0, 1, \ldots, M+1$, denoted by $\tilde{x} = x_0 x_1 \cdots x_M x_{M+1}$. Thus, absorption, scattering or reflection events happen at $x_1, \ldots, x_M$. The set of all paths of length M is denoted by Ω_M. The path space Ω is the countable set of all paths Ω_M of finite length,

$$\Omega = \bigcup_{M=2}^{\infty} \Omega_M. \tag{3.2}$$

A direction is denoted by $\omega \in S^2$, where S^2 is a unit sphere in $\Re^3$. A unit vector $\omega_{x_m, x_{m+1}}$ is the direction from vertex x_m to vertex x_{m+1} in a path $\tilde{x}$.

Veach [67] introduced a framework representing the rendering equation in the form of a path integral for scenes without participating media (i.e., no scattering), and later Pauly et al.[52] extended it to the volume rendering equation with scattering. The amount of light I observed by the detector is given by the path integral

$$I = \int_{\Omega} f(\tilde{x}) d\mu(\tilde{x}), \tag{3.3}$$

an integral over the path space. Here $\mu(\tilde{x})$ is a measure of path $\tilde{x}$,

$$d\mu(\tilde{x}) = \prod_{m=0}^{M+1} d\mu(x_m), \quad d\mu(x_m) = \begin{cases} dA(x_m), & x_m \in \partial v, \\ dV(x_m), & x_m \in v_0, \end{cases} \tag{3.4}$$

where $d\mu(x_m)$ denotes the differential measure at vertex x_m. $f(\tilde{x})$ is a measurement contribution function defined as follows;

$$f(\tilde{x}) = L_e(x_0, x_1) G(x_0, x_1) \left[\prod_{m=1}^{M} f_f(x_{m-1}, x_m, x_{m+1}) G(x_m, x_{m+1}) \right] W_e(x_M, x_{M+1}), \tag{3.5}$$

where $W_e(x_M, x_{M+1})$ is the camera response function, and $L_e(x_0, x_1)$ is the intensity of the light emitted from the light source x_0 to vertex x_1. $f_f(x_{m-1}, x_m, x_{m+1})$ is a scattering kernel at x_m with respect to the locations of vertices x_{m-1} and x_{m+1},

$$f_f(x_{m-1}, x_m, x_{m+1}) = \begin{cases} f_s(x_{m-1}, x_m, x_{m+1}), & x_m \in \partial v, \\ \sigma_s(x_m) f_p(x_{m-1}, x_m, x_{m+1}), & x_m \in v_0. \end{cases} \tag{3.6}$$

Here, the bidirectional scattering distribution function (BSDF) $f_s(x_{m-1}, x_m, x_{m+1})$ is used for locations on the surface of objects, and the scattering coefficient $\sigma_s(x_m)$ at x_m and phase function $f_p(x_{m-1}, x_m, x_{m+1})$ are used for those inside the medium. $G(x_m, x_{m+1})$ is a generalized geometric term:

$$G(x_m, x_{m+1}) = T(x_m, x_{m+1}) g(x_m, x_{m+1}), \tag{3.7}$$

where $g(x_m, x_{m+1})$ is a geometric term

$$g(x_m, x_{m+1}) = \begin{cases} \frac{|\boldsymbol{n}_g(x_m) \cdot \omega_{x_m, x_{m+1}}|}{\|x_m - x_{m+1}\|^2}, & x_m \in \partial V, \\ \frac{1}{\|x_m - x_{m+1}\|^2}, & x_m \in V_0, \end{cases} \tag{3.8}$$

with unit normal $\boldsymbol{n}_g(x_m)$ of the surface at $x_m \in \partial V$. $T(x_m, x_{m+1})$ is a transmittance which describes the attenuation when light passes through the medium;

$$T(x_m, x_{m+1}) = \begin{cases} e^{-\tau(x_m, x_{m+1})}, & \{x_m, x_{m+1}\} \subset V_0 \cup \partial V, \\ 0, & \text{otherwise}, \end{cases} \tag{3.9}$$

$$\tau(x_m, x_{m+1}) = \int_0^1 \sigma_t((1-s)x_m + s x_{m+1}) ds, \tag{3.10}$$

where $\sigma_t(x_m)$ is the extinction coefficient at vertex x_m.

Putting all together, we have a path integral of the following infinite sum of all possible path contributions;

$$I = \sum_{M=2}^{\infty} \sum_{k \in \Omega_M} L_e(x_0, x_1) G(x_0, x_1) \left[\prod_{m=1}^{M} f_f(x_{m-1}, x_m, x_{m+1}) G(x_m, x_{m+1}) \right] W_e(x_M, x_{M+1}) \prod_{m=0}^{M+1} d\mu(x_m). \tag{3.11}$$

Note that all vertices $\{x_m\}$ depend on a path k; different paths have different sets of vertices. In the equation above however we omit the path index k for simplicity. Later we will use k as path index again.

3.2 Assumptions on the path integral formulation

As our target is optical tomography, we restrict the model to deal with inside participating media. To do so, we assume that the light source x_0 and detector x_{M+1} are located on the surface, and the other vertices $x_1, x_2, \ldots, x_M, x_{M+1}$ are inside the medium; that is, $x_0, x_{M+1} \in$

$\partial\nu$ and $x_1,\ldots,x_M \in \nu_0$. Then the transmittance is simplified as

$$T(x_m, x_{m+1}) = e^{-\tau(x_m, x_{m+1})}. \tag{3.12}$$

Furthermore, we assume that the observations are ideal and the camera response function is the identity; $W_e(x_M, x_{M+1}) = 1$.

Apart from the assumptions above, we rewrite the geometric term and the differential measure. The definitions above use area measures $dA(x_m)$ and volume measures $dV(x_m)$ along with the squared distance geometric term[57, 52, 44], however steradian measures $d\omega(x_m)$ and the identity geometric term are equivalent and also widely used[39, 22, 53],

$$g(x_m, x_{m+1}) d\mu(x_m) = d\omega(x_m). \tag{3.13}$$

Therefore, we employ the steradian measures and rewrite as follows:

$$g(x_m, x_{m+1}) = 1 \tag{3.14}$$

$$d\mu_k(x_m) = \begin{cases} dA(x_0), & m = 0, \\ d\omega(x_m), & m = 1,\ldots,M+1. \end{cases} \tag{3.15}$$

Now Eq.(3.11) is written as

$$I = \sum_{M=2}^{\infty} \sum_{k \in \Omega_M} L_e(x_0, x_1) T(x_0, x_1) dA(x_0) \left[\prod_{m=1}^{M} f_f(x_{m-1}, x_m, x_{m+1}) T(x_m, x_{m+1}) d\omega(x_m) \right] d\omega(x_{M+1}). \tag{3.16}$$

3.3 Discretization of the forward model

For numerical computation, we first discretize the medium into voxels of a regular grid, where each voxel has its own extinction coefficient $\sigma_t[b]$ (b is the index of the voxel) as shown in Figure 3.1.

With this voxelization, the paths of light are also divided into segments, as explained below. First we explain the integral (3.10) along a single segment $x_m x_{m+1}$ of a path $\tilde{x}$. It describes the attenuation of light along the segment due to the extinction coefficients of the voxels involved. Because of the discretization of the medium, the integral (3.10) can be

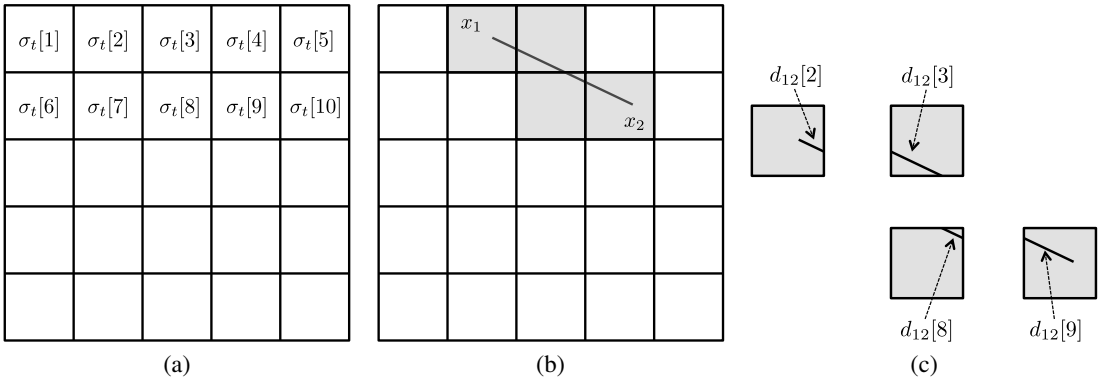


Fig. 3.1 Illustration of a discretization example. (a) Voxelization of the medium into a regular grid of size 5×5. Voxels are indexed in raster scan order in this example; from left to right, and top to bottom. Each voxel b has extinction coefficient $\sigma_t[b]$. (b) A path segment between vertices x_1 and x_2. Voxels involved in the segment are shaded. (c) Lengths $d_{12}[b]$ of the involved voxels $b = 2,3,8,9$. Here we denote $d_{12}[b]$ instead of $d_{x_1,x_2}[b]$ for simplicity.

written as a sum of voxel-wise multiplications;

$$\tau(x_m, x_{m+1}) = \int_0^1 \sigma_t((1-s)x_m + sx_{m+1})ds = \sum_{b \in \mathscr{B}_{x_m,x_{m+1}}} \sigma_t[b] d_{x_m,x_{m+1}}[b] = \boldsymbol{\sigma}_t^T \boldsymbol{d}_{x_m,x_{m+1}}. \tag{3.17}$$

For the second equality, b is the index of a set $\mathscr{B}_{x_m,x_{m+1}}$ of all voxels involved by segment $x_m x_{m+1}$, and $d_{x_m,x_{m+1}}[b]$ is the length of the part of the segment $x_m x_{m+1}$ passing through voxel b. This is illustrated in Fig.3.1(c). The extinction coefficient σ_t is now a peace-wise constant function because of the voxelization, then the integral turns into a sum[2].

This simplifies the computation, however the sum over a set $\mathscr{B}_{x_m,x_{m+1}}$ is not preferable in terms of implementation and optimization. We propose here to use a vector representation of both extinction coefficients and segment lengths, which is the third equality of the above equation. The first vector $\boldsymbol{\sigma}_t$ stores the values of the extinction coefficients $\sigma_t[b]$ of all voxels. This vector can be generated by serializing the voxels on the grid in a certain order. The second vector $\boldsymbol{d}_{x_m,x_{m+1}}$ contains the values of the lengths $d_{x_m,x_{m+1}}[b]$ for all voxels. We should note that this vector is very sparse; most of the voxels have no intersection with the segment $x_m x_{m+1}$. Hence, only few elements in $\boldsymbol{d}_{x_m,x_{m+1}}$ have non-zero values, and the other elements are zero because those voxels b have no intersection and $d_{x_m,x_{m+1}}[b] = 0$.

[2]The idea that this integral can be turned into a sum, has been discussed before[6], however not in the context of tomography.

This sparsity of the vector facilitates the construction of a whole path $\tilde{x}$ because path segments can be "added" as follows;

$$\boldsymbol{D}_k = \sum_{m=0}^{M} \boldsymbol{d}_{x_m, x_{m+1}}, \tag{3.18}$$

where $\boldsymbol{D}_k$ is the vector of a complete path k of length $M+2$; the b-th element can be interpreted as the length of the segment when the path passes through voxel b. This notation simplifies a part of Eq.(3.16) as follows;

$$\prod_{m=0}^{M} T(x_m, x_{m+1}) = \prod_{m=0}^{M} e^{-\tau(x_m, x_{m+1})} = e^{-\sum_{m=0}^{M} \tau(x_m, x_{m+1})} = e^{-\sum_{m=0}^{M} \boldsymbol{\sigma}_t^T \boldsymbol{d}_{x_m, x_{m+1}}} = e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_k}. \tag{3.19}$$

Using this notation to rewrite Eq.(3.16), we have

$$I = \sum_{M=2}^{\infty} L_e(x_0, x_1) \sum_{k \in \Omega_M} H_k e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_k} = L_e(x_0, x_1) \sum_{k \in \Omega} H_k e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_k}, \tag{3.20}$$

where the factor H_k defined as

$$H_k = dA(x_0) d\omega(x_{M+1}) \prod_{m=1}^{M} f_f(x_{m-1}, x_m, x_{m+1}) d\omega(x_m), \tag{3.21}$$

describes the contributions of the scattering coefficients and phase functions, and the exponential factor represents attenuation due to absorption (and out-scattering) over the path.

3.4 2D layered model of forward scattering

As a first attempt, we design a 2D layered grid, instead of the 3D one. Since we voxelize the medium into a regular grid, the 2D medium consists of parallel layers. Hereafter, a 3D direction ω between vertices is written as a 2D direction θ, and a steradian measure $d\omega$ as an angle measure $d\theta$.

As shown in Fig.3.2, we assume a particular layer scattering having the following properties. First, vertices $x_1 \cdots x_M$ of path $\tilde{x}$ are located at the centers of each voxel. Light source x_0 is located on the boundary of the top surface of the voxels in the top layer. Similarly, detector x_{M+1} is located on the boundary of the bottom surface of the voxels in the bottom layer. Second, directions θ_{x_0, x_1} and $\theta_{x_M, x_{M+1}}$ at the beginning and end of a path are

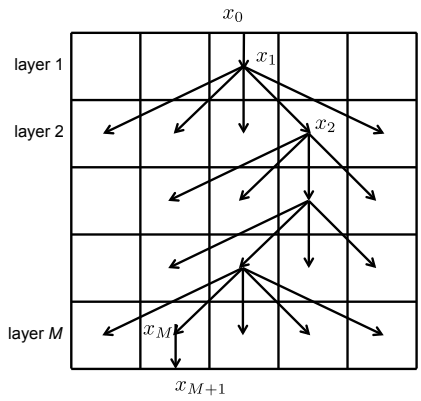


Fig. 3.2 Proposed 2D layered model of scattering. This example shows path $\tilde{x}$ consisting of vertices $x_1 \cdots x_M$ located at the centers of voxels in a grid with M parallel layers. x_0 is a light source located on the top surface, and x_{M+1} is a detector at the bottom. At each vertex, the light scatters to voxels in the next layer, and possible scattering directions are indicated by arrows.

perpendicular to the boundary. This means that scattering begins at x_1 and ends at x_M. Third, forward scattering happens layer by layer. More specifically, light is scattered at the center of a voxel in a layer, then goes to the center of a voxel in the next (below) layer. Scattering is assumed to happen every time the light traverses voxel centers. Even if the next voxel is just below the current voxel and the path segment is straight, it is regarded as scattering. Fourth, the scattering coefficient is uniform; $\sigma_s(x) = \sigma_s$.

By ignoring paths exiting from the sides of the grid, the number of all possible paths is N^M, where M is the number of layers and N is the number of voxels in one layer.

3.5 Approximating the phase function with a Gaussian

We use a Gaussian model $f_p(\theta, \sigma^2)$ as an approximation of the phase function;

$$f_p(x_{m-1}, x_m, x_{m+1}) \equiv f_p(\theta_m, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-\theta_m^2}{\sigma^2}\right), \quad -\frac{\pi}{2} < \theta_m < \frac{\pi}{2}, \tag{3.22}$$

where the variance σ^2 controls the scattering property; larger values of σ^2 mean strong forward scattering. This Gaussian approximation is convenient in our case because of the following two reasons.

First, existing phase function models [35, 14, 20, 50, 53, 39] are those for three dimensional scattering, not for 2D. This means that those functions are normalized for integrals over the unit sphere S^2: $\int_{S^2} f_p(\omega) d\omega = 1$. Most of the phase functions assume isotropy

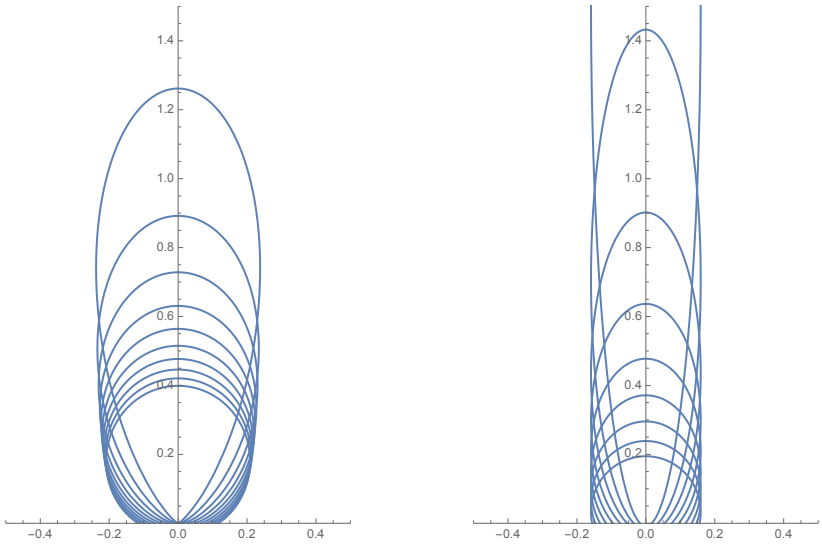


Fig. 3.3 Comparison of two-dimensional phase functions. The upward vertical direction is $\theta = 0$, and horizontal directions are $\theta = \pm\frac{\pi}{2}$. (a) Gaussian approximated phase functions with $\sigma^2 = 0.1, 0.2, \ldots, 1.0$. The tallest and narrowest shape corresponds to $\sigma^2 = 0.1$, and the shape becomes shorter and rounder for larger values of σ^2. (b) Heino's two-dimensional analogs[34] of Henyey-Greenstein's phase function with parameter $g = 0.1, 0.2, \ldots, 1.0$. The tallest and narrowest shape corresponds to $g = 1.0$, and the shape becomes shorter and close to a hemisphere for smaller values of g.

(rotational symmetry) and hence the function has a form taking angle θ as an argument, however $\int_{-\pi}^{\pi} f_p(\theta)d\theta \neq 1$. These functions therefore are not adequate for our case.

Second, our assumption of layer-wise forward scattering does not allow scattering to happen backward or sideway, and the Gaussian model is suitable for it. As shown in Figure 3.3, the Gaussian model has the form of forward-only scattering (no backward or sideway) in a reasonable range of σ^2, and it is almost normalized; $\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} f_p(\theta, \sigma^2)d\theta \approx 1$. Other two-dimensional phase functions exist, which are not forward-only. For example, Heino et al.[34] introduced a two-dimensional analog of Henyey-Greenstein's phase function[35], shown in Fig.3.3. Although the parameters are different, the two functions in Fig.3.3 have similar shapes. The most important difference is that Heino's function has backward scattering, but our Gaussian model doesn't. More realistic scattering rather than the layer-wise forward scattering introduced here needs Heino's or Henyey-Greenstein's phase function.

We should note one further simplification in our layer-wise forward scattering model. The angle θ_m in the phase function is usually defined between θ_{x_{m-1},x_m} and $\theta_{x_m,x_{m+1}}$, that is, the difference of directions changed by the scattering event. Instead of dealing with

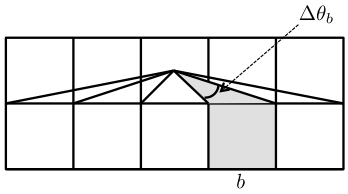


Fig. 3.4 An illustration of angle measure $\Delta\theta_b$ for voxel b in the next layer. For the center voxel of the upper layer, voxel b (shaded) in the next layer subtends an angle of $\Delta\theta_b$, which is used for the angle measure in Eq. (3.23).

such an exact difference of directions, we use the angle between $\theta_{x_m,x_{m+1}}$ and the vertical (downward) direction for efficiency of computation. This assumption enables us to discretize the Gaussian phase function much easier. Since $f_p(\theta)$ integrates to (approximately) one, such a normalization can be discretized with a sum as follows;

$$\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} f_p(\theta,\sigma^2)d\theta \approx \sum_{b\in\mathscr{B}_n} f_p(\theta_b,\sigma^2)\Delta\theta_b \approx 1, \tag{3.23}$$

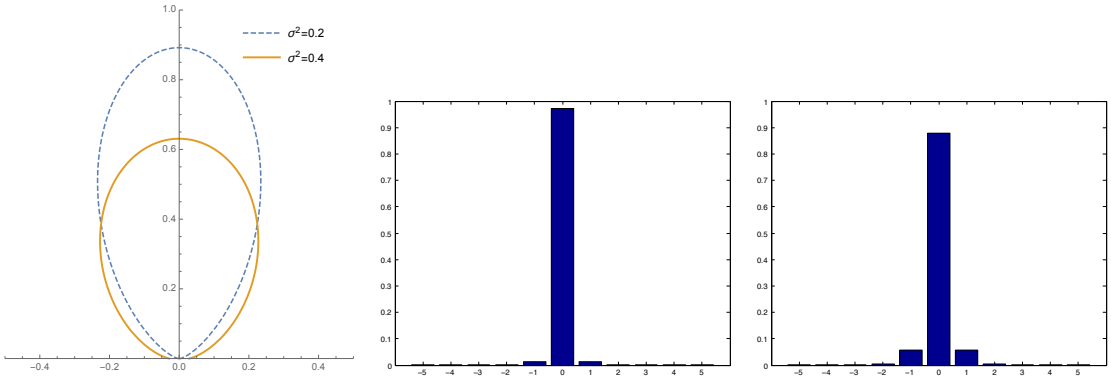
where $\mathscr{B}$ is a set of voxel indices in the next layer n, θ_b is an alternative form of the corresponding $\theta_{x_m,x_{m+1}}$, and $\Delta\theta_b$ is the angle measure as shown in Figure 3.4.

The equation above can be considered as the energy distribution from a voxel in one layer to the voxels in the next layer. For a voxel b at direction θ_b, the value of $f_p(\theta_b,\sigma^2)\Delta\theta_b$ describes what percentage of the energy will be scattered to this voxel. Figure 3.5 shows plots of the values corresponding to two phase functions with different parameters. We can see that due to forward scattering most of the energy is concentrated in the voxel just below, and a small part goes to the adjacent voxels.

The contribution H_k in Eq. (3.21) now needs to be rewritten so that it deals with the Gaussian phase function and the discretized energy distribution discussed above. First we reorder the measure

$$H_k = dA(x_0)d\theta(x_{M+1})\prod_{m=1}^{M} f_f(x_{m-1},x_m,x_{m+1})d\theta(x_m) \tag{3.24}$$

$$= dA(x_0)d\theta(x_1)\prod_{m=1}^{M} f_f(x_{m-1},x_m,x_{m+1})d\theta(x_{m+1}), \tag{3.25}$$



(a) (b) (c)

Fig. 3.5 (a) The phase functions with parameter $\sigma^2 = 0.2$ (dashed line) and $\sigma^2 = 0.4$ (solid line). (b,c) Plot of the value $f_p(\theta_b, \sigma^2)\Delta\theta_b$ for each voxel b for (b) $\sigma^2 = 0.2$ and (c) $\sigma^2 = 0.4$. Note that index b is relative to the voxel in the next layer just below the voxel in consideration. The voxel just below is $b = 0$, the voxel on its right side is $b = 1$, and on the left side is $b = -1$.

then replace the factors with the Gaussian phase function;

$$H_k = dA(x_0)\Delta\theta_{x_0,x_1}\sigma_s^M \prod_{m=1}^{M} f_p(\theta_{x_m,x_{m+1}}, \sigma^2)\Delta\theta_{x_m,x_{m+1}}. \tag{3.26}$$

Note that the factor $dA(x_0)\Delta\theta_{x_0,x_1}\sigma_s^M$ is common for all paths because we assumed that the grid is uniform so that $dA(x_0)$ is constant, and the direction θ_{x_0,x_1} (or ω_{x_0,x_1}) is perpendicular to the top surface, and σ_s is constant.

3.6 Observation model

Suppose the 2D layered medium is an $M \times N$ grid; it has M layers each of which is made of N voxels. We now construct an observation model of the light transport between a light source and a detector: emitting light to each of the voxels at the top layer, and capturing light from each voxel from the bottom layer. More specifically, let $i \in \mathscr{B}_1$ and $j \in \mathscr{B}_M$ be voxel indices of the light source and detector locations, respectively. By restricting the light paths only to those connecting i and j, the observed light I_{ij} is written as follows;

$$I_{ij} = I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}}, \tag{3.27}$$

Algorithm 1: Computing contribution H_{ijk} and omitting low contribution path by thresholding.

Input: Threshold th, path $\tilde{x} = x_0 \cdots x_{M+1}$.
Output: Contribution H_{ijk}.

```
H_ijk = 1;
for m = 1 to M do
    H_ijk = H_ijk f_p(θ_{x_m,x_{m+1}}, σ^2) Δθ_{x_m,x_{m+1}}
    if H_ijk ≤ th then
        stop;
        omit this path;
accept this path;
return H_ijk;
```

where H_{ijk} and $\boldsymbol{D}_{ijk}$ are the same as in Eqs.(3.26) and (3.20), respectively, but restricted to paths connecting i and j, and $I_0 = L_e(x_0, x_1)$ assuming the light source being constant.

In the above equation, k indexes the light paths which share the same i and j. Due to the layered scattering model in the $N \times M$ grid, the number of different paths between i and j is $N_{ij} = N^{M-2}$. This is however too large even for small N and M, e.g. $N = M = 10$. Therefore we exclude paths having small contributions from the computation. This is done by a simple thresholding while computing H_{ijk} as shown in Algorithm 1. This results in generating fewer paths; $N_{ij} \leq N^{M-2}$. For example, there are $N_{ij} = 742$ paths for $N = M = 20$ with $\sigma^2 = 0.4$ when $th = 0.001$, which enable us to reduce the computation cost.

3.7 Summary

In this chapter, we proposed an optical tomography method using path integral as a forward model. After discretization and adding constraints and assumptions, we obtained a forward model.

Suppose the 2D layered medium is an $M \times N$ grid, let $i \in \mathscr{B}_1$ and $j \in \mathscr{B}_M$ be voxel indices of the light source and detector locations, respectively. By restricting the light paths only to those connecting i and j, the observed light I_{ij} is written as follows;

$$I_{ij} = I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}}, \tag{3.28}$$

where H_{ijk} and $\boldsymbol{D}_{ijk}$ are the same as in Eqs.(3.26) and (3.20), respectively, but restricted to paths connecting i and j, and $I_0 = L_e(x_0, x_1)$ assuming the light source being constant. k indexes the shrinkked light paths which share the same i and j.

Chapter 4

Log-barrier Interior Point approach

In chapter 3, we obtained a forward model under constraints. Suppose the 2D layered medium is an $M \times N$ grid, let $i \in \mathscr{B}_1$ and $j \in \mathscr{B}_M$ be voxel indices of the light source and detector locations, respectively. By restricting the light paths only to those connecting i and j, the observed light I_{ij} is written as follows;

$$I_{ij} = I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}}, \tag{4.1}$$

where H_{ijk} and $\boldsymbol{D}_{ijk}$ are the same as in Eqs.(3.26) and (3.20), respectively, but restricted to paths connecting i and j, and $I_0 = L_e(x_0, x_1)$ assuming the light source being constant. k indexes the shrinkked light paths which share the same i and j.

4.1 Method: Inverse problem

Next, we propose a method for the inverse problem of the forward model (4.1) to estimate the extinction coefficients of the 2D layered model. As we mentioned before, we fix the light paths and assume that the scattering coefficients and parameters of the Gaussian phase function are uniform and known in advance.

4.1.1 Cost function

In the $M \times N$ 2D layered medium described in the last section, we had assumed a configuration of a light source and detector as the one show in the left-most figure of Fig.4.1; the light source is located above the medium and the detector is below, and the observed light is I_{ij} where i, j are the voxel indices of the light source and detector locations. By sliding the

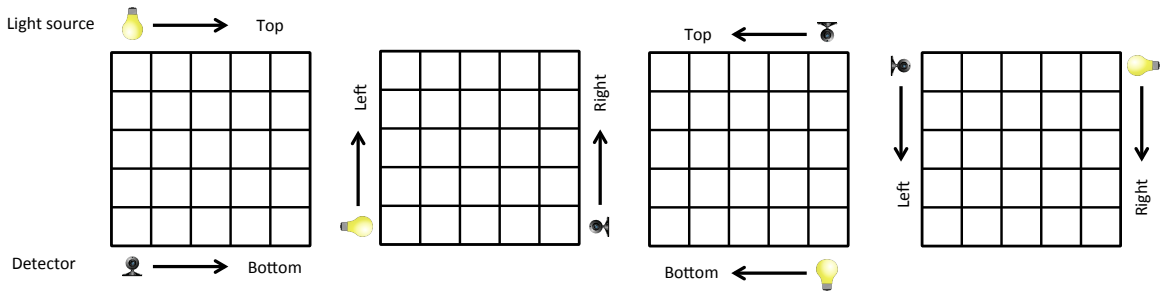


Fig. 4.1 Four configurations of light sources and detectors. From left to right, we call configurations T2B (top-to-bottom), L2R (left-to-right), B2T (bottom-to-top), R2L (right-to-left) which represent locations of light sources and detectors.

light source and the detector, we can obtain N^2 observations, resulting in the following least squares problem

$$\min_{\boldsymbol{\sigma}_t} f_0, \quad f_0 = \sum_{i=1}^{N}\sum_{j=1}^{N}\left| I_{ij} - I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}} \right|^2, \tag{4.2}$$

under $2MN$ constraints

$$0 \preceq \boldsymbol{\sigma}_t \preceq u \tag{4.3}$$

where $\preceq$ denotes the generalized inequality, i.e. all elements in the vector must satisfy the inequality. The lower bound 0 comes from the fact that any media must have positive extinction coefficients, while the upper bound u is used for numerical stability to exclude unrealistic values to be estimated.

Furthermore, as shown in Fig.4.1, we have four configurations of light sources and detectors by changing their positions. This gives us four different sets of observations I_{ij} and paths ijk. These four different sets lead to four objective functions (f_{T2B}, f_{L2R}, f_{B2T}, f_{R2L}) as shown in Fig.4.1. Since the four objective functions share the same variables $\boldsymbol{\sigma}_t$, we can use all of them at the same time by adding them to form a new single function f_0 at the expense of additional (factor of four) computation cost;

$$\min_{\boldsymbol{\sigma}_t} f_0 \quad f_0 = f_0^{T2B} + f_0^{L2R} + f_0^{B2T} + f_0^{R2L} \tag{4.4}$$

$$\text{subject to} \quad 0 \preceq \boldsymbol{\sigma}_t \preceq u. \tag{4.5}$$

4.1.2 Optimization problem with inequality constraints

Since the inverse problem (4.5) is non-linear, we employ an interior point method[**?**]; an iterative optimization algorithm for problems with constraints. Here we first review several key points in optimization, then we will develop an algorithm to solve 4.5 along with required first and second order derivatives of the cost function (4.5).

Unconstrained problem: Quasi-Newton

First we review optimization without constraints, which is used inside the interior point method. The general form of unconstrained optimization is

$$\min_{\boldsymbol{\sigma}_t} f(\boldsymbol{\sigma}_t) \tag{4.6}$$

where $\boldsymbol{\sigma}_t \in \Re^{N\times M}$ is a real vector and $f : \Re^{N\times M} \to \Re$ is an objective function which is twice continuously differentiable.

To solve it, an iterative procedure begins with an initial guess ${\boldsymbol{\sigma}_t}^0$ and generates a sequence $\{{\boldsymbol{\sigma}_t}^k\}_{k=0}^{\infty}$. It stops when the change of solutions is small enough. The information about function f at ${\boldsymbol{\sigma}_t}^k$ or even previous estimates ${\boldsymbol{\sigma}_t}^0, {\boldsymbol{\sigma}_t}^1, \cdots, {\boldsymbol{\sigma}_t}^{k-1}$ are used to calculate a direction $\boldsymbol{p}_k$ to move with a step size α_k. Line search is often used to determine the step size by searching along the direction starting from ${\boldsymbol{\sigma}_t}^k$ for finding ${\boldsymbol{\sigma}_t}^{k+1}$ with the least value of the objective function;

$$\min_{\alpha_k>0} f({\boldsymbol{\sigma}_t}^k + \alpha_k \boldsymbol{p}_k) \tag{4.7}$$

Once we find the step size, the estimate ${\boldsymbol{\sigma}_t}^{k+1}$ is updated as ${\boldsymbol{\sigma}_t}^{k+1} \leftarrow {\boldsymbol{\sigma}_t}^k + \alpha_k \boldsymbol{p}_k$. The direction is $\boldsymbol{p}_k = -B_k \nabla f({\boldsymbol{\sigma}_t}^k)$ for the Newton's method, where $B_k = \nabla^2 f({\boldsymbol{\sigma}_t}^k)^{-1}$ is the inverse of the Hessian.

The Newton's method is well known for its second order convergence and accuracy. However, when the dimension of the problem is large, calculating the Hessian and its inverse is computationally expensive. Therefore Quasi-Newton methods are often used, where the inverse Hessian is updated by incremental approximations in order to reduce the computation

Algorithm 2: The Quasi-Newton method with BFGS update rule.

Input: A feasible initial solution $\boldsymbol{\sigma}_t{}^0$, and $B_0 \succ 0$.
Result: An estimate $\boldsymbol{\sigma}_t{}^\star$.

```
repeat
    Compute the Quasi-Newton direction: p^k = −B_k ∇f(σ_t^k).
    Find step length α_k with line search.
    Update estimate σ_t^{k+1} ← σ_t^k + α_k p^k.
    Update B_k with BFGS.
until convergence;
```

cost. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) update rules are well known[51];

$$\boldsymbol{s} = \boldsymbol{\sigma}_t{}^k - \boldsymbol{\sigma}_t{}^{k-1} \tag{4.8}$$

$$\boldsymbol{y} = \nabla f(\boldsymbol{\sigma}_t{}^k) - \nabla f(\boldsymbol{\sigma}_t{}^{k-1}) \tag{4.9}$$

$$B_k = \left(I - \frac{\boldsymbol{s}\boldsymbol{y}^T}{\boldsymbol{y}^T\boldsymbol{s}}\right) B_{k-1} \left(I - \frac{\boldsymbol{y}\boldsymbol{s}^T}{\boldsymbol{y}^T\boldsymbol{s}}\right) + \frac{\boldsymbol{s}\boldsymbol{s}^T}{\boldsymbol{y}^T\boldsymbol{s}}. \tag{4.10}$$

When the conditions $\boldsymbol{y}^T\boldsymbol{s} > 0$ and $B_0 \succ 0$ (where $\succ 0$ means positive definite) are satisfied, the BFGS update guarantees the positive definiteness of B_k. Algorithm 2 shows the Quasi-Newton method.

Constrained problem: Interior point

Here we introduce a constrained optimization with inequality constraints of the form;

$$\min_{\boldsymbol{\sigma}_t} f_0(\boldsymbol{\sigma}_t) \quad \text{subject to} \quad f_i(x) \leq 0, \quad i = 1, \cdots, m, \tag{4.11}$$

where $\boldsymbol{\sigma}_t \in \Re^{N\times M}$ is a real vector and $f_0, \cdots, f_m : \Re^{N\times M} \to \Re$ are twice continuously differentiable.

The idea is to approximate it as an unconstrained problem. Using Lagrange multipliers, we can first rewrite problem (4.11) as

$$\min_{\boldsymbol{\sigma}_t} f_0(\boldsymbol{\sigma}_t) + \sum_{i=1}^{m} I(f_i(\boldsymbol{\sigma}_t)), \tag{4.12}$$

where $I : \Re \to \Re$ is an indicator function which keeps the solution inside the feasible region;

$$I(f) = \begin{cases} 0, & f \leq 0 \\ \infty, & f > 0. \end{cases} \tag{4.13}$$

The problem (4.12) now has no inequality constraints, while it is not differentiable due to I.

The barrier method[16] is an interior point method which introduces a logarithmic barrier function to approximate the indicator function I as follows;

$$\hat{I}(f) = -(1/t)\log(-f), \tag{4.14}$$

where $t > 0$ is a parameter to adjust the accuracy of approximation. The log barrier function goes to infinity rapidly as f goes close to 0 while it is close to 0 when f are far away from 0. Since $\hat{I}(f)$ is differentiable, we have

$$\min_{\boldsymbol{\sigma}_t} f_0(\boldsymbol{\sigma}_t) + \sum_{i=1}^{m} -(1/t)\log(-f_i(\boldsymbol{\sigma}_t)), \tag{4.15}$$

or equivalently,

$$\min_{\boldsymbol{\sigma}_t} t f_0(\boldsymbol{\sigma}_t) - \sum_{i=1}^{m} \log(-f_i(\boldsymbol{\sigma}_t)). \tag{4.16}$$

The barrier method solves (4.16) iteratively by increasing the parameter t. At the limit of $t \to \infty$, the above problem coincides with the original problem (4.12).

4.1.3 Algorithm for solving the inverse problem

Algorithm 3 shows the our algorithm which uses a barrier method with Quasi-Newton for solving the inverse problem. We should mention the following parts where we have modified the original algorithm[16].

Warm start For each inner loop, the Quasi-Newton method needs initial guess of the inverse Hessian B_0. Instead of fixing B_0 for every inner loop, we reuse B_k of the last inner loop to accelerate the convergence (shown in Lines 4 and 19 in Algorithm 3).

Checking feasibility Since the Quasi-Newton method and line search estimate without constraints, the next estimate ${\boldsymbol{\sigma}_t}^{k+1}$ may go beyond the constraints; in our case, each element ${\sigma_t}^{k+1}[b]$ in ${\boldsymbol{\sigma}_t}^{k+1}$ must be inside $[0, u]$ after step size has been determined. Therefore in Line

8 we check the feasibility of the estimate ${\boldsymbol{\sigma}_t}^{k+1}$ for the current step size α_k. If it exceeds the boundary of the feasible region, we pull the estimate back into the feasible region by halving the step size. If it is still outside of the feasible region, then the step size is halved again. Why don't we just set the step size so that ${\boldsymbol{\sigma}_t}^{k+1}$ is exactly on the boundary? The reason is the log-barrier: if ${\boldsymbol{\sigma}_t}^{k+1}$ is on the boundary, in other words, ${\sigma_t}^{k+1}[b]$ is either 0 or u, then $\log(\sigma_t[b])$ or $\log(u-\sigma_t[b])$ becomes infinite, which results in numerical instability. Therefore, the procedure described above is needed.

Checking for positive definiteness The BFGS update rules guarantee B_k to be positive definite if $\boldsymbol{y}^T\boldsymbol{s} > 0$ and $B \succ 0$ are satisfied. While the latter is satisfied by giving an appropriate initial guess, the former however depends on the updates at each iteration. If it is not satisfied, then the BFGS updates is no longer valid and we reset the inverse Hessian B_k to a scaled identity [51] at line 16.

Jacobian

Here we represent the Jacobian of the objective function f_0 in Eq. (4.2). Note that the objective function f_0 in Eq. (4.5) can be derived in the same manner.

We first rewrite the objective function f_0 as follows;

$$f_0 = \sum_{i=1}^{N}\sum_{j=1}^{N}\left| I_{ij} - I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}} \right|^2 \tag{4.18}$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{N}\left(I_{ij}^2 - 2I_{ij}I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}} + I_0^2 \sum_{k=1}^{N_{ij}}\sum_{l=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}} H_{ijl} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijl}} \right) \tag{4.19}$$

$$= \sum_{i=1}^{N}\sum_{j=1}^{N}\left(I_{ij}^2 - 2I_{ij}I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}} + I_0^2 \sum_{k=1}^{N_{ij}}\sum_{l=1}^{N_{ij}} H_{ijk} H_{ijl} e^{-\boldsymbol{\sigma}_t^T (\boldsymbol{D}_{ijk}+\boldsymbol{D}_{ijl})} \right), \tag{4.20}$$

an the gradient of f_0 is given by

$$\frac{\partial f_0}{\partial \boldsymbol{\sigma}_t} = \sum_{i=1}^{N}\sum_{j=1}^{N}\left(2I_{ij}I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}} \boldsymbol{D}_{ijk} - I_0^2 \sum_{k=1}^{N_{ij}}\sum_{l=1}^{N_{ij}} H_{ijk} H_{ijl} e^{-\boldsymbol{\sigma}_t^T (\boldsymbol{D}_{ijk}+\boldsymbol{D}_{ijl})} (\boldsymbol{D}_{ijk}+\boldsymbol{D}_{ijl}) \right). \tag{4.21}$$

Algorithm 3: Barrier method of interior point with Quasi-Newton solver.

Data: Parameters $\mu > 1$, $\varepsilon > 0$, and $t = t_{\text{init}} > 0$.
Input: A feasible initial estimate ${\boldsymbol{\sigma}_t}^0$, and $B \succ 0$.
Result: An estimate ${\boldsymbol{\sigma}_t}^\star$.

while $\frac{2MN}{t} \geq \varepsilon$ **do** `// outer loop: barrier method`
  $t \leftarrow \mu t$.
  Set a log-barriered cost function;

$$f(t) = tf_0 - \sum_b \big(\log(\sigma_t[b]) + \log(u - \sigma_t[b])\big) \tag{4.17}$$

  $k \leftarrow 0$, $B_k \leftarrow B$, ${\boldsymbol{\sigma}_t}^k \leftarrow \boldsymbol{\sigma}_t$.
  repeat `// inner loop: Quasi-Newton`
    Compute the Quasi-Newton direction: $\boldsymbol{p}^k = -B_k \nabla f({\boldsymbol{\sigma}_t}^k)$.
    Find step length α_k with line search.
    while ${\boldsymbol{\sigma}_t}^k + \alpha_k \boldsymbol{p}^k$ *is not feasible* **do**
      Halve the step size: $\alpha_k \leftarrow \alpha_k/2$.
    Update estimate ${\boldsymbol{\sigma}_t}^{k+1} \leftarrow {\boldsymbol{\sigma}_t}^k + \alpha_k \boldsymbol{p}^k$.
    $\boldsymbol{s} = {\boldsymbol{\sigma}_t}^{k+1} - {\boldsymbol{\sigma}_t}^k$.
    $\boldsymbol{y} = \nabla f({\boldsymbol{\sigma}_t}^{k+1}) - \nabla f({\boldsymbol{\sigma}_t}^k)$.
    if $\boldsymbol{y}^T \boldsymbol{s} > 0$ **then**
      Update B_{k+1} with BFGS (4.10).
    else
      Reset $B_{k+1} \leftarrow \frac{\boldsymbol{y}^T \boldsymbol{s}}{\boldsymbol{y}^T \boldsymbol{y}} I$.
    $k \leftarrow k+1$.
  until $\frac{1}{2} \nabla f({\boldsymbol{\sigma}_t}^{k+1})^T B_{k+1} \nabla f({\boldsymbol{\sigma}_t}^{k+1}) \leq \varepsilon$;
  $B \leftarrow B_{k+1}$, $\boldsymbol{\sigma}_t \leftarrow {\boldsymbol{\sigma}_t}^k$.

To simplify the equation, we use the following notation;

$$E = \begin{bmatrix} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ij1}} \\ e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ij2}} \\ \vdots \\ e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijN_{ij}}} \end{bmatrix}, \quad H = \begin{bmatrix} H_{ij1} \\ H_{ij2} \\ \vdots \\ H_{ijN_{ij}} \end{bmatrix} \tag{4.22}$$

$$\boldsymbol{D}_{ij} = \begin{bmatrix} \boldsymbol{D}_{ij1} \\ \boldsymbol{D}_{ij2} \\ \vdots \\ \boldsymbol{D}_{ijN_{ij}} \end{bmatrix}, \quad \widetilde{\boldsymbol{D}_{ij}} = \begin{bmatrix} \boldsymbol{D}_{ij1} + \boldsymbol{D}_{ij1} & \boldsymbol{D}_{ij1} + \boldsymbol{D}_{ij2} & \cdots & \boldsymbol{D}_{ij1} + \boldsymbol{D}_{ijN_{ij}} \\ \boldsymbol{D}_{ij2} + \boldsymbol{D}_{ij1} & \boldsymbol{D}_{ij2} + \boldsymbol{D}_{ij2} & \cdots & \boldsymbol{D}_{ij2} + \boldsymbol{D}_{ijN_{ij}} \\ \vdots & \vdots & \cdots & \vdots \\ \boldsymbol{D}_{ijN_{ij}} + \boldsymbol{D}_{ij1} & \boldsymbol{D}_{ijN_{ij}} + \boldsymbol{D}_{ij2} & \cdots & \boldsymbol{D}_{ijN_{ij}} + \boldsymbol{D}_{ijN_{ij}} \end{bmatrix}. \tag{4.23}$$

Now f_0 and the gradient can be represented as

$$f_0 = \sum_{i=1}^{N}\sum_{j=1}^{N}\left(I_{ij}^2 - 2I_{ij}I_0E^TH + I_0^2(E^TH)^2\right) \tag{4.24}$$

$$\frac{\partial f_0}{\partial \boldsymbol{\sigma}_t} = \sum_{i=1}^{N}\sum_{j=1}^{N}\left(2I_{ij}I_0\,\mathrm{sum}[(E\times H)\otimes \boldsymbol{D}_{ij}] - I_0^2\,\mathrm{sum}[((E\times H)(E\times H)^T)\otimes \widetilde{\boldsymbol{D}_{ij}}]\right), \tag{4.25}$$

where sum[] stands for the sum over the elements of the container (5.25) of vectors, $\times$ is the element-wise product, and $\otimes$ denotes the tensor product defined as

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \cdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix}, \quad B = \begin{bmatrix} \boldsymbol{b}_{11} & \boldsymbol{b}_{12} & \cdots & \boldsymbol{b}_{1m} \\ \boldsymbol{b}_{21} & \boldsymbol{b}_{22} & \cdots & \boldsymbol{b}_{2m} \\ \vdots & \vdots & \cdots & \vdots \\ \boldsymbol{b}_{n1} & \boldsymbol{b}_{n2} & \cdots & \boldsymbol{b}_{nm} \end{bmatrix} \tag{4.26}$$

$$A\otimes B = \begin{bmatrix} a_{11}\boldsymbol{b}_{11} & a_{12}\boldsymbol{b}_{12} & \cdots & a_{1m}\boldsymbol{b}_{1m} \\ a_{21}\boldsymbol{b}_{21} & a_{22}\boldsymbol{b}_{22} & \cdots & a_{2m}\boldsymbol{b}_{2m} \\ \vdots & \vdots & \cdots & \vdots \\ a_{n1}\boldsymbol{b}_{n1} & a_{n2}\boldsymbol{b}_{n2} & \cdots & a_{nm}\boldsymbol{b}_{nm} \end{bmatrix}. \tag{4.27}$$

4.2 Numerical simulations

In this section, we report the results obtained by numerical simulations using the proposed model.

The following parameters have been used in Algorithm 3: $t_{init} = 1.0$, $\mu = 1.5$, $\varepsilon = 10^{-2}$. For the line search, the range for the step size was $\alpha_k \in [0, 100]$. For the initial guess we used $B = I$, $\boldsymbol{\sigma}_t{}^0 = \mathbf{0}$. For the 2D layered medium, the grid size was set to $N = M = 20$ with square voxels of size 1 [mm], i.e. the medium is 20 [mm] $\times$ 20 [mm], and $dA = 1$ [mm]. The values of the extinction coefficients are set between 1.05 and 1.55 [mm^{-1}], and the upper bound (4.3) is set to $u = 2.0$ [mm^{-1}]. The parameter of the Gaussian phase function is 0.2 or 0.4, and the scattering coefficient is set to $\sigma_s = 1$ [mm^{-1}]. The threshold for excluding low contribution paths is $th = 0.001$.

The ground truth and the estimated extinction coefficients are shown in Figure 4.2. The matrix plots in the top row of the figure represents five different media (from (a) to (e)) used for the simulation. Each voxel b is shaded in gray according to the values of the extinction coefficient $\sigma_t[b]$, and darker gray represents larger values of $\sigma_t[b]$. Also the values of $\sigma_t[b]$ are displayed at each voxel. In the same manner, the middle and bottom rows show the estimated

results when the following values of the parameter of the Gaussian phase function were used: $\sigma^2 = 0.2$ and 0.4. Figure 4.3 shows the observations I_{ij} in a matrix form, from which the extinction coefficients are estimated. Each element in these plots is now an observation I_{ij}. We can see observations with higher values (shown in darker shades of gray in the plots) on the diagonal. The observations obtained for $\sigma^2 = 0.4$ seem to be fainter than those obtained for $\sigma^2 = 0.2$ due to the larger amount of scattering.

The left-most column of Fig. 4.2(a) shows the simplest case: the medium has almost homogeneous extinction coefficients of value 1.05 (voxels shaded in light gray) except few voxels with much higher coefficients of 1.2 (voxels shaded in dark gray), which means that those voxels absorb much more light than other voxels. The coefficients are estimated reasonably well as shown in the middle and bottom rows, and the root-mean-squared error (RMSE) shown in Table 4.1 is small enough with the relative error of $0.0075/1.05 = 0.7\%$ to the background coefficient value. The other media, shown in columns (b)–(e), have more complex distributions of the extinction coefficients. We summarize the quality of the estimated results in terms of RMSE in Table 1. Numbers in the brackets are relative errors of RMSE to the background extinction coefficient values (i.e., 1.05). Computation time is also shown in Table 1. Note that our proposed method has been currently implemented in Matlab, which can be accelerated further by using C++.

The values of the cost function f_0 over iterations of the outer loop in Algorithm 3 are shown in Figure 4.4 for each medium. These curves show that the proposed method effectively minimizes the original objective function (4.5) for five different types of media shown here and probably for other media. Figure 4.5 demonstrates how the log-barriered cost function f in Eq. (4.17) evolves over all iterations of the inner loop; the number of iterations in the horizontal axis accumulates all inner iterations of the Quasi-Newton method. We can see that each inner loop successively minimizes the log-barriered function and the warm start (reusing the Hessian from the previous outer loop) may help the gap of values between inner loops.

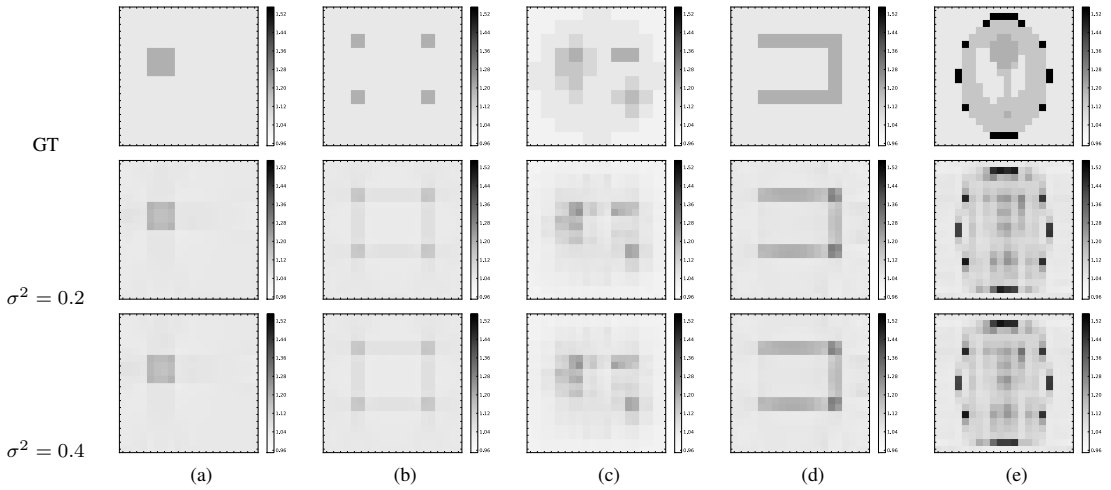


Fig. 4.2 Numerical simulation results for a grid of size 20×20. Darker shades of gray represent larger values (more light is absorbed at the voxel). The bars on the side show extinction coefficient values in greyscale. The first row shows ground truth for five different types of media (a)–(e) used for the simulation. The second and third rows show estimated results for $\sigma^2 = 0.2$ and $\sigma^2 = 0.4$, respectively, of the Gaussian phase function.

Table 4.1 RMSEs and computation time for the numerical simulations for five different types of media (a)–(e) with grid size of 20×20, for two different Gaussian phase function parameter values. Numbers in the brackets are relative errors of RMSE to the background extinction coefficient values (i.e., 1.05).

		(a)	(b)	(c)	(d)	(e)
RMSE	$\sigma^2 = 0.2$	0.0067506 (0.643%)	0.014253 (1.36%)	0.017771 (1.69%)	0.016220 (1.54%)	0.057692 (5.49%)
	$\sigma^2 = 0.4$	0.0075305 (0.717%)	0.014369 (1.37%)	0.017704 (1.69%)	0.015692 (1.49%)	0.058464 (5.57%)
Computation time [s]	$\sigma^2 = 0.2$	142	113	297	190	269
	$\sigma^2 = 0.4$	127	110	186	156	267

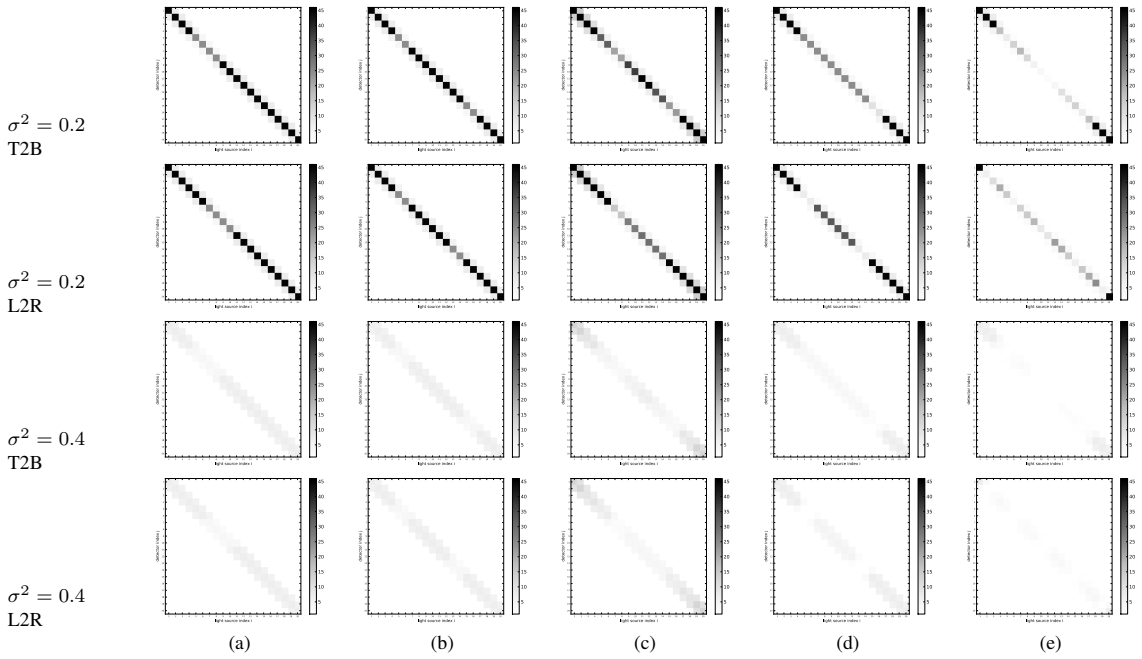


Fig. 4.3 Visualization of the observations I_{ij} in a matrix form. Each matrix shows I_{ij} in its i-th row and j-th column. The horizontal index i indicates the location of the light source, and the vertical index j the location of the detector. Hence, I_{ij} is the light intensity with the detector at j and the light source at i. Darker shades of grey represent larger observation values (brighter light is observed). (left to right columns) Five different media (a)–(e) used for the simulation in the same order as in Fig. 4.2. (top to bottom rows) I_{ij} for T2B and L2R configurations for $\sigma^2 = 0.2$ and $\sigma^2 = 0.4$.

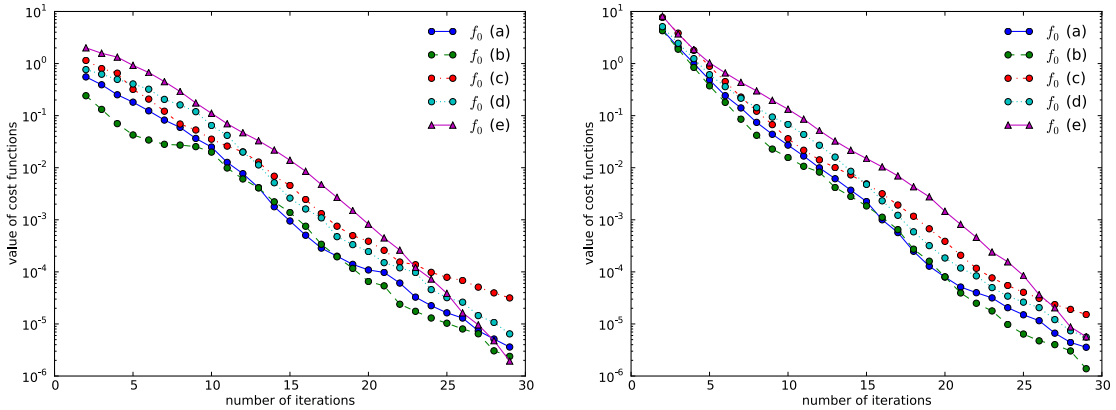


Fig. 4.4 Original cost function values f_0 over iterations of the outer loop of Algorithm 3 with $\sigma^2 = 0.2$ (left) and 0.4 (right). The horizontal axis shows the number of outer iterations, and the vertical axis represents the log of the original cost function values. Different plots indicate five different types of media (a)–(e) used for the simulation.

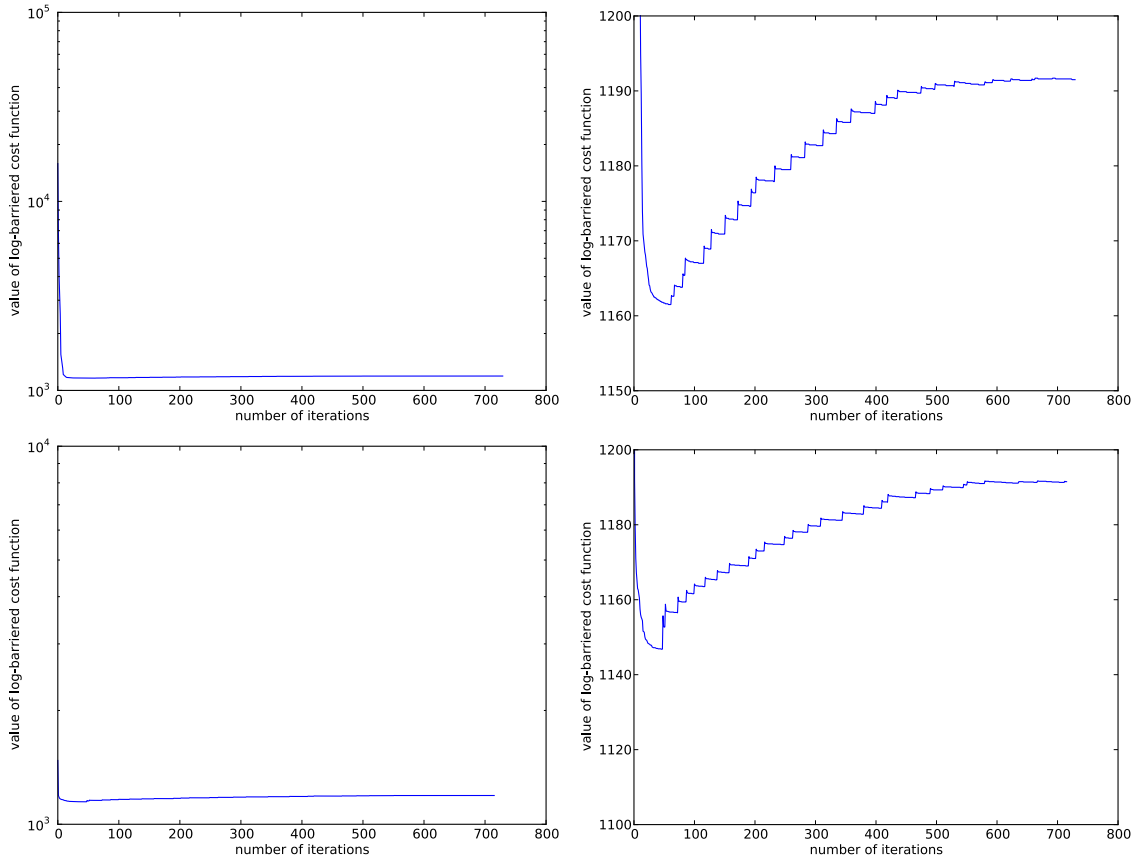


Fig. 4.5 Log-barriered cost function values f over iterations of all inner loops of Algorithm 3 for medium (e) with $\sigma^2 = 0.2$ (top) and 0.4 (bottom). The horizontal axis shows the number of total inner iterations accumulated across different outer loops. The vertical axis represents the original cost function values (left) in log scale and (right) in linear scale.

4.2.1 Comparison results

We compare our method to a standard DOT with Finite Element Methods (FEM) [54, 55] using different optimization methods implemented in the Electrical Impedance Tomography and Diffuse Optical Tomography Reconstruction Software (EIDORS) [54, 55]. The ground truth used in this comparison is shown in the top row of Figure 11 (a) – (e); $N = M = 24$ medium of the size 24 [mm] $\times$ 24 [mm] with extinction coefficient distributions almost the same as those shown in Fig.4.2 (a) – (e).

For solving DOT by EIDORS, we used $24 \times 24 \times 24 = 1152$ triangle meshes (i.e., each voxel is divided into two triangle meshes), and for the boundary condition we placed 16 light sources and 16 detectors at the same interval around the medium. We chose two solvers: Gauss-Newton (GN) method and Primal-Dual (PD) interior point method. We used $\boldsymbol{\sigma}_t{}^0 = \mathbf{0}$ as the initial guess for both our method and EIDORS.

The results obtained by our method ($\sigma^2 = 0.4$) and DOT with GN and PD are shown in Fig.5.1. The results obtained by the proposed method are shown in the second row, which are similar to those in the third row of Fig.4.2. The third row in Fig.5.1 shows results for DOT with GN. This kind of blurred results are typical for DOT estimation due to its diffusion approximation. The last row shows results for DOT with PD, which look better than those obtained for DOT with GN, but still have a tendency of overestimating the high coefficient value areas.

We summarize RMSE values and computation time for each method in Table 4.2 in the same format with Table 4.1. RMSE values of our method are 2 to 5 times smaller than those of DOT, and this demonstrates that the proposed method can achieve much more accurate results.

The current disadvantage is its large computation cost, as our method takes up to 1000 times longer than DOT. We plan to reduce the computation cost by optimizing the code using C++ and adopting other solvers.

4.3 Summary

Combining chapter 3 and chapter 4, we have proposed a path integral based approach to optical tomography for multiple scattering in discretized participating media. Assuming the scattering coefficients and phase function are known and uniform, the extinction coefficients at each voxel in a 2D layered medium are estimated by using an interior point method.

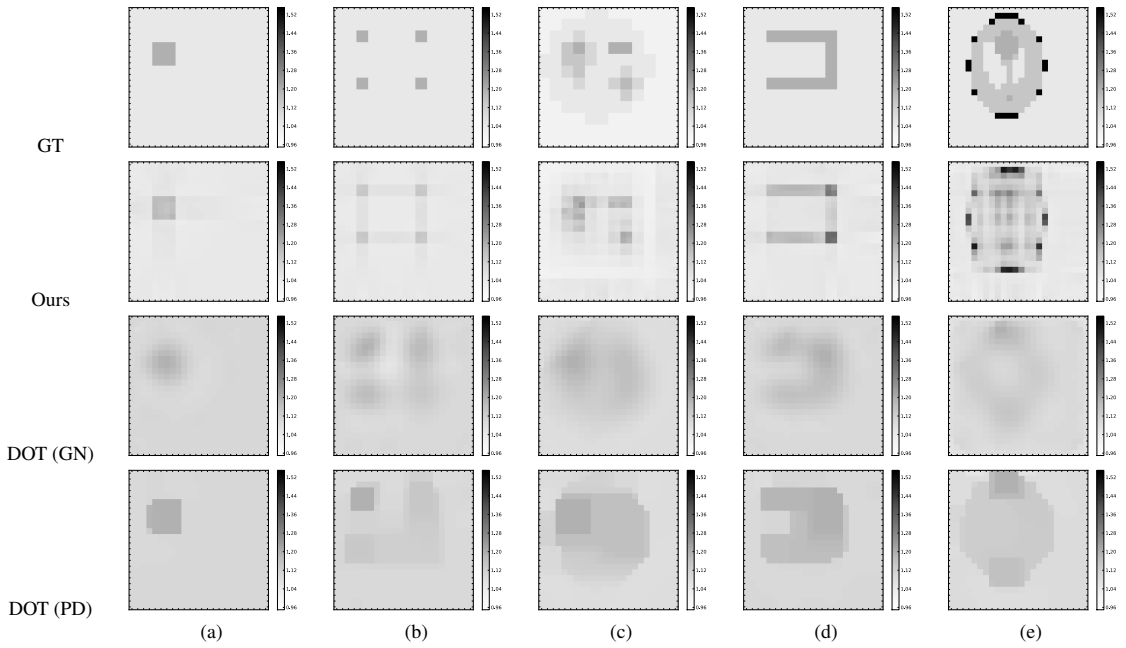


Fig. 4.6 Numerical simulation results for a grid of size 24×24, comparing our method to DOT with two solvers. Darker shades of gray represent larger values (more light is absorbed at the voxel). The bars on the side show extinction coefficient values in greyscale. First row shows the ground truth for five different types of media (a)–(e) used for the simulation. Second row shows the estimated results of the proposed method. Third and fourth rows show estimated results for DOT by using Guass-Newton (GN) and Primal-Dual (PD) interior point solvers.

A large part of the computational cost of the proposed method comes from the forward model prediction (3.27), which appears in the gradient computation (4.2). It depends on the number of paths N_{ij}; we use currently about 700 paths out of all 20^{18} possible paths, and for each path we need to compute path vectors $\boldsymbol{D}_{ijk}$, $\boldsymbol{D}_{ijk} + \boldsymbol{D}_{ijl}$, and factors H_{ijk}. A possible acceleration is the precomputation of these variables but this would lead to a trade-off with storage cost. Each $\boldsymbol{D}_{ijk}$ has dimension of $20 \times 20 = 400$, each pair of ij has about 700 vectors of $\boldsymbol{D}_{ijk}$, and the number of pairs ij (hence observations) is $20 \times 20 = 400$. In total, about 450MB memory would be required even if single precision floating numbers were used for storing all $\boldsymbol{D}_{ijk}$. Fortunately, these vectors are necessarily sparse, and we have used sparse matrices to store them. However, the increase will be linear in the number of paths N_{ij} and quadratic with the grid size $\max(N,M)$. Therefore we make another efficient implementations while maintain the accuracy of the proposed method in the next chapter.

Table 4.2 RMSEs and computation time for the numerical simulations for five different types of media (a)–(e) with grid size of 24×24, for the proposed method and DOT with two solvers. Numbers in the brackets are relative errors of RMSE to the background extinction coefficient values (i.e., 1.05).

		(a)	(b)	(c)	(d)	(e)
RMSE	Ours $\sigma^2 = 0.4$	0.007662 (0.730%)	0.01244 (1.18%)	0.026602 (2.53%)	0.021442 (2.04%)	0.051152 (4.87%)
	DOT (GN)	0.053037 (5.05%)	0.060597 (5.77%)	0.7605 (7.53%)	0.059534 (5.67%)	0.0855 (8.14%)
	DOT (PD)	0.052466 (5.25%)	0.0626 (5.97%)	0.081081 (8.11%)	0.066042 (6.60%)	0.080798 (8.08%)
Computation time [s]	Ours $\sigma^2 = 0.4$	257	217	382	306	504
	DOT (GN)	0.397	0.390	0.407	0.404	0.453
	DOT (PD)	1.11	1.09	1.14	1.08	1.15

Chapter 5

Primal-dual Interior Point approach

In this Chapter 5, following the same inverse problem in Chapter 4, we will use primal-dual interior-point approach to solve the following optimization problem:

$$\min_{\boldsymbol{\sigma}_t} f \quad \text{s.t.} \quad \sigma_{tl} \leq \boldsymbol{\sigma}_t \leq \sigma_{tu}, \tag{5.1}$$

where f is the cost function

$$f = \sum_{i=1}^{N} \sum_{j=1}^{N} \left| I_{ij} - I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_i^T \boldsymbol{D}_{ijk}} \right|^2, \tag{5.2}$$

and σ_{tl} and σ_{tu} are lower and upper bounds, respectively. i is the position of the light source i and j is the position the detector j. $\{I_{ij}\}$ is a set of observations we obtained. The box constraints are due to the nature of the extinction coefficient being positive (i.e., $\sigma_{tl} > 0$), and the numerical stability of excluding unrealistic large values.

5.1 Primal-dual interior point method of the inverse problem

Here we develop a primal-dual interior point method to solve the inverse problem (5.1). It is an inequality constraint optimization with box constraints, which is straightforward to apply a standard primal-dual method [16]. However, we can use the structure of the box constraints, hence we derive an efficient algorithm by using the problem structure.

5.1.1 Primal-dual method

We first rewrite the inequality constraint problem to an equivalent problem with equality constraints with slack variables $\boldsymbol{s} = (s_1, \ldots, s_{2NM})^T$ as follows;

$$\min_{\boldsymbol{\sigma}_t} f \quad \text{s.t.} \quad \boldsymbol{c} - \boldsymbol{s} = \mathbf{0},\ 0 \leq \boldsymbol{s}, \tag{5.3}$$

where $\boldsymbol{c}$ is a vector of the box constraints

$$\boldsymbol{c} = \begin{pmatrix} c_1(\boldsymbol{\sigma}_t) \\ \vdots \\ c_{2NM}(\boldsymbol{\sigma}_t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{\sigma}_t - \sigma_{tl}\mathbf{1} \\ \sigma_{tu}\mathbf{1} - \boldsymbol{\sigma}_t \end{pmatrix}. \tag{5.4}$$

Here c_i is the i-th constraint, and $\mathbf{1}$ is a vector of ones.

The Lagrangian L of the above problem is

$$L(\boldsymbol{\sigma}_t, \boldsymbol{s}, \boldsymbol{z}) = f - \sum_{i=1}^{2NM} z_i(c_i - s_i) = f - \boldsymbol{z}^T(\boldsymbol{c} - \boldsymbol{s}), \tag{5.5}$$

where $\boldsymbol{z}$ is a vector of Lagrangian multipliers, or dual variables.

The KKT conditions of Eq. (5.3) with duality gap μ is written as

$$\nabla f - A^T\boldsymbol{z} = \mathbf{0}, \quad S\boldsymbol{z} - \mu\mathbf{1} = \mathbf{0}, \quad \boldsymbol{c} - \boldsymbol{s} = \mathbf{0}, \quad \boldsymbol{s} \geq \mathbf{0}, \quad \boldsymbol{z} \geq \mathbf{0}, \tag{5.6}$$

where $S = \mathrm{diag}(\boldsymbol{s})$, and

$$A = \nabla\boldsymbol{c} = \begin{pmatrix} \nabla c_1 \\ \vdots \\ \nabla c_{2NM} \end{pmatrix} = \begin{pmatrix} I \\ -I \end{pmatrix}. \tag{5.7}$$

Here I is an identity matrix.

To solve the system of the KKT conditions by using Newton's method, we have the following system of equations;

$$\begin{pmatrix} \nabla^2 L & \mathbf{0} & -A^T \\ \mathbf{0} & Z & S \\ A & -I & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{p}_\sigma \\ \boldsymbol{p}_s \\ \boldsymbol{p}_z \end{pmatrix} = - \begin{pmatrix} \nabla f - A^T\boldsymbol{z} \\ S\boldsymbol{z} - \mu\mathbf{1} \\ \boldsymbol{c} - \boldsymbol{s} \end{pmatrix}, \tag{5.8}$$

where $Z = \mathrm{diag}(\boldsymbol{z})$, and $\nabla^2 L$ is the Hessian of the Lagrangian.

5.1.2 Solving the system efficiently

The matrix in Eq. (5.8) is of the size $5NM \times 5NM$, which is sparse but large, and computationally expensive to solve. We therefore develop an efficient way to solve th system by using the problem structure.

First the system is explicitly written as follows:

$$\begin{cases} \nabla^2 L \boldsymbol{p}_\sigma - A^T \boldsymbol{p}_z = -\nabla f + A^T \boldsymbol{z} \\ Z\boldsymbol{p}_s + S\boldsymbol{p}_z = -S\boldsymbol{z} + \mu \mathbf{1} \\ A\boldsymbol{p}_\sigma - \boldsymbol{p}_s = -\boldsymbol{c} + \boldsymbol{s}. \end{cases} \tag{5.9}$$

Substituting the last equation into the second one yields

$$A^T S^{-1} Z(A\boldsymbol{p}_\sigma + \boldsymbol{c} - \boldsymbol{s}) + A^T \boldsymbol{p}_z = -A^T \boldsymbol{z} + \mu A^T S^{-1} \mathbf{1}, \tag{5.10}$$

and we add the both side to the first equation to obtain

$$\nabla^2 L \boldsymbol{p}_\sigma + A^T S^{-1} Z(A\boldsymbol{p}_\sigma + \boldsymbol{c} - \boldsymbol{s}) = -A^T \boldsymbol{z} + \mu A^T S^{-1} \mathbf{1} - \nabla f + A^T \boldsymbol{z}. \tag{5.11}$$

Here we define $\boldsymbol{w} = \boldsymbol{s}^{-1} \odot \boldsymbol{z}$ and $\boldsymbol{y} = \mu \boldsymbol{s}^{-1} - \boldsymbol{w} \odot \boldsymbol{c} + \boldsymbol{z}$, where $\odot$ is the Hadamard (element-wise) product, $\boldsymbol{s}^{-1}$ is a vector of element-wise reciprocals of $\boldsymbol{s}$. Then we have

$$\boldsymbol{p}_\sigma = (\nabla^2 L + A^T S^{-1} Z A)^{-1}(-\nabla f + A^T \boldsymbol{y}) \tag{5.12}$$

$$\boldsymbol{p}_z = \mu \boldsymbol{s}^{-1} - \boldsymbol{w} \odot \boldsymbol{p}_s - \boldsymbol{z}. \tag{5.13}$$

By exploiting the structure of matrix A, and defining $\boldsymbol{w} = (\boldsymbol{w}_l^T, \boldsymbol{w}_u^T)^T$ to split a vector into two parts corresponding to lower and upper bound constraints, we have

$$A^T S^{-1} Z A = \begin{pmatrix} I & -I \end{pmatrix} \operatorname{diag}(\boldsymbol{w}) \begin{pmatrix} I \\ -I \end{pmatrix} = \operatorname{diag}(\boldsymbol{w}_l + \boldsymbol{w}_u). \tag{5.14}$$

Similarly, we define $\boldsymbol{y} = (\boldsymbol{y}_l^T, \boldsymbol{y}_u^T)^T$ to simplify $A^T \boldsymbol{y}$ as $A^T \boldsymbol{y} = \boldsymbol{y}_l - \boldsymbol{y}_u$ and then

$$A\boldsymbol{p}_\sigma = \begin{pmatrix} I \\ -I \end{pmatrix} \boldsymbol{p}_\sigma = \begin{pmatrix} \boldsymbol{p}_\sigma \\ -\boldsymbol{p}_\sigma \end{pmatrix}. \tag{5.15}$$

Now the solution is given by

$$\begin{cases} \boldsymbol{p}_\sigma = (\nabla^2 L + \mathrm{diag}(\boldsymbol{w}_l + \boldsymbol{w}_u))^{-1}(-\nabla f + \boldsymbol{y}_l - \boldsymbol{y}_u) \\ \boldsymbol{p}_s = \begin{pmatrix} \boldsymbol{p}_\sigma \\ -\boldsymbol{p}_\sigma \end{pmatrix} + \boldsymbol{c} - \boldsymbol{s} \\ \boldsymbol{p}_z = \mu \boldsymbol{s}^{-1} - \boldsymbol{w} \odot \boldsymbol{p}_s - \boldsymbol{z}, \end{cases} \tag{5.16}$$

which involves the inversion of the size $NM \times NM$.

5.1.3 Update variables

Once $\boldsymbol{p}_\sigma, \boldsymbol{p}_s$, and $\boldsymbol{p}_z$ are obtained, we then estimate the step length to update the parameters [16]. The maximum of the step lengths is given by the following rule;

$$\begin{cases} \alpha_s^{max} = \max\{\alpha \in [0,1] \mid \boldsymbol{s} + \alpha \boldsymbol{p}_s \geq (1-\tau)\boldsymbol{s}\} \\ \alpha_z^{max} = \max\{\alpha \in [0,1] \mid \boldsymbol{z} + \alpha \boldsymbol{p}_z \geq (1-\tau)\boldsymbol{z}\}, \end{cases} \tag{5.17}$$

with $\tau \in (0,1)$ are used (for example, $\tau = 0.995$). This prevents variables $\boldsymbol{s}$ and $\boldsymbol{z}$ from approaching to the lower boundary.

Next we perform the backtracking line search [51] to estimate acceptable step lengths α_s and α_z. To this end, we use the following exact merit function ϕ with $\eta \in (0,1)$;

$$\phi(\boldsymbol{\sigma}_t, \boldsymbol{s}) = f - \mu \sum_{i=1}^{2MN} \log(s_i) + v\|\boldsymbol{c}(x) - \boldsymbol{s}\|, \tag{5.18}$$

and make a sufficient decrease requirement

$$\phi(\boldsymbol{\sigma}_t + \alpha_s \boldsymbol{p}_\sigma, \boldsymbol{s} + \alpha_s \boldsymbol{p}_s) \leq \phi(\boldsymbol{\sigma}_t, \boldsymbol{s}) + \eta \alpha_s D_\phi(\boldsymbol{\sigma}_t, \boldsymbol{s}; \boldsymbol{p}_\sigma, \boldsymbol{p}_s), \tag{5.19}$$

where $D_\phi(\boldsymbol{\sigma}_t, \boldsymbol{s}; \boldsymbol{p}_\sigma, \boldsymbol{p}_s)$ denotes the directional derivative of ϕ in the direction $(\boldsymbol{p}_\sigma, \boldsymbol{p}_s)$. The step lengths α_s and α_z are found in the ranges $\alpha_s \in (0, \alpha_s^{max}]$ and $\alpha_z \in (0, \alpha_z^{max}]$ so that Eq. (5.19) is satisfied.

Then the parameters $\boldsymbol{\sigma}_t$, $\boldsymbol{s}$ and $\boldsymbol{z}$ are updated as

$$\begin{cases} \boldsymbol{\sigma}_t \leftarrow \boldsymbol{\sigma}_t + \alpha_s \boldsymbol{p}_\sigma \\ \boldsymbol{s} \leftarrow \boldsymbol{s} + \alpha_s \boldsymbol{p}_s \\ \boldsymbol{z} \leftarrow \boldsymbol{z} + \alpha_z \boldsymbol{p}_z. \end{cases} \tag{5.20}$$

Algorithm 4: Primal-Dual interior point with line search.

Data: $\mu > 0, \sigma \in (0,1), \varepsilon_{TOL} > 0, \varepsilon_\mu, \eta \in (0,1), k = 0$.
Input: A feasible initial estimates $\boldsymbol{\sigma}_t$, $\boldsymbol{s} > 0$, and $\boldsymbol{z} > 0$.
Input: $B_0 = I$ `// Only for Quasi-Newton`
Result: Estimates of $\boldsymbol{\sigma}_t$

repeat
  repeat `// inner loop`
    Compute the decent direction $\boldsymbol{p} = (\boldsymbol{p}_\sigma, \boldsymbol{p}_s, \boldsymbol{p}_z)$
    Compute the step lengths α_s and α_z
    Update $\boldsymbol{\sigma}_t, \boldsymbol{s}, \boldsymbol{z}$
    Update the approximation B_k `// Only for Quasi-Newton`
    Set $k \leftarrow k+1$
  until $E(\boldsymbol{\sigma}_{tk}, \boldsymbol{s}_k, \boldsymbol{z}_k; \mu) \leq \varepsilon_\mu$;
  $\mu \leftarrow \sigma\mu$
  $\varepsilon_\mu \leftarrow \mu$
until $E(\boldsymbol{\sigma}_{tk}, \boldsymbol{s}_k, \boldsymbol{z}_k; 0) \leq \varepsilon_{TOL}$;

Once following error function is smaller than a given threshold, the primal-dual interior point method stops;

$$E(\boldsymbol{\sigma}_t, \boldsymbol{s}, \boldsymbol{z}; \mu) = \max\{\|\nabla f - A^T \boldsymbol{z}\|, \|S\boldsymbol{z} - \mu \mathbf{1}\|, \|\boldsymbol{c} - \boldsymbol{s}\|\}. \tag{5.21}$$

Algorithm 4 summaries the primal-dual interior point method developed above. Note that the Hessian $\nabla^2 L$ can be approximated as B_k at each iteration k by the Quasi-Newton method, instead of the full Hessian used by the Newton's method. We will compare the Newton's method and the Quasi-Newton method in the section of experiments.

5.2 Efficient formulations

The most computationally intensive part of the primal-dual algorithm shown above is the computation of Hessians for Newton's method and Jacobians for Newton's and Quasi-Newton methods. We propose here efficient formulations of Hessian and Jacobian of the problem, whose computational cost is much smaller than naive formulations used in the previous approaches.

First we show the naive and old formulations of Hessian and Jacobian, then introduce our new formulations, followed by discussions on computational cost.

5.2.1 Previous old formulations for inverse problem

Here we show how the previous approaches [65, 72, 71] do. We call these the *old* formulations.

Jacobian: old formulation

Remind that the objective function f to be minimized is

$$f = \sum_{i=1}^{N}\sum_{j=1}^{N}\left(I_{ij}^2 - 2I_{ij}I_0\sum_{k=1}^{N_{ij}} H_{ijk}e^{-\boldsymbol{\sigma}_t^T\boldsymbol{D}_{ijk}} + I_0^2\sum_{k=1}^{N_{ij}}\sum_{l=1}^{N_{ij}} H_{ijk}H_{ijl}e^{-\boldsymbol{\sigma}_t^T(\boldsymbol{D}_{ijk}+\boldsymbol{D}_{ijl})}\right). \tag{5.22}$$

The gradient of f is given as follows by taking the derivative of the objective function;

$$\begin{aligned}\frac{\partial f}{\partial \boldsymbol{\sigma}_t} = \sum_{i=1}^{N}\sum_{j=1}^{N}\Bigg(&2I_{ij}I_0\sum_{k=1}^{N_{ij}} H_{ijk}e^{-\boldsymbol{\sigma}_t^T\boldsymbol{D}_{ijk}}\boldsymbol{D}_{ijk} \\ &- I_0^2\sum_{k=1}^{N_{ij}}\sum_{l=1}^{N_{ij}} H_{ijk}H_{ijl}e^{-\boldsymbol{\sigma}_t^T(\boldsymbol{D}_{ijk}+\boldsymbol{D}_{ijl})}(\boldsymbol{D}_{ijk}+\boldsymbol{D}_{ijl})\Bigg).\end{aligned} \tag{5.23}$$

To simplify the equation, the following notations are introduced;

$$\boldsymbol{E}_{ij} = \begin{pmatrix} e^{-\boldsymbol{\sigma}_t^T\boldsymbol{D}_{ij1}} \\ e^{-\boldsymbol{\sigma}_t^T\boldsymbol{D}_{ij2}} \\ \vdots \\ e^{-\boldsymbol{\sigma}_t^T\boldsymbol{D}_{ijN_{ij}}}\end{pmatrix}, \boldsymbol{H}_{ij} = \begin{pmatrix} H_{ij1} \\ H_{ij2} \\ \vdots \\ H_{ijN_{ij}}\end{pmatrix}, D_{ij} = \begin{pmatrix} \boldsymbol{D}_{ij1} \\ \boldsymbol{D}_{ij2} \\ \vdots \\ \boldsymbol{D}_{ijN_{ij}}\end{pmatrix} \tag{5.24}$$

$$\widetilde{D_{ij}} = \begin{pmatrix} \boldsymbol{D}_{ij1}+\boldsymbol{D}_{ij1} & \cdots & \boldsymbol{D}_{ij1}+\boldsymbol{D}_{ijN_{ij}} \\ \boldsymbol{D}_{ij2}+\boldsymbol{D}_{ij1} & \cdots & \boldsymbol{D}_{ij2}+\boldsymbol{D}_{ijN_{ij}} \\ \vdots & \ddots & \vdots \\ \boldsymbol{D}_{ijN_{ij}}+\boldsymbol{D}_{ij1} & \cdots & \boldsymbol{D}_{ijN_{ij}}+\boldsymbol{D}_{ijN_{ij}}\end{pmatrix}. \tag{5.25}$$

Now f and the gradient are rewritten as follows:

$$f = \sum_{i=1}^{N}\sum_{j=1}^{N}\left(I_{ij}^2 - 2I_{ij}I_0\boldsymbol{E}_{ij}^T\boldsymbol{H}_{ij} + I_0^2(\boldsymbol{E}_{ij}^T\boldsymbol{H}_{ij})^2\right) \tag{5.26}$$

$$\begin{aligned}\frac{\partial f}{\partial \boldsymbol{\sigma}_t} = \sum_{i=1}^{N}\sum_{j=1}^{N}\Big(&2I_{ij}I_0\,\mathrm{sum}[(\boldsymbol{E}_{ij}\odot\boldsymbol{H}_{ij})\otimes D_{ij}] \\ &- I_0^2\,\mathrm{sum}[((\boldsymbol{E}_{ij}\odot\boldsymbol{H}_{ij})(\boldsymbol{E}_{ij}\odot\boldsymbol{H}_{ij})^T)\otimes\widetilde{D_{ij}}]\Big),\end{aligned} \tag{5.27}$$

where sum[] stands for the sum over the elements of the container (5.25) of vectors, and $\otimes$ denotes the tensor product.

Hessian: old formulation

We define the following notations;

$$\boldsymbol{\beta}_{ijx} = \begin{pmatrix} D_{ij1x} \\ D_{ij2x} \\ \vdots \\ D_{ijkx} \end{pmatrix} \tag{5.28}$$

$$\boldsymbol{\gamma}_{ijx} = \begin{pmatrix} D_{ij1x}+D_{ij1x} & \cdots & D_{ij1x}+D_{ijkx} \\ D_{ij2x}+D_{ij1x} & \cdots & D_{ij2x}+D_{ijkx} \\ \vdots & \ddots & \vdots \\ D_{ijkx}+D_{ij1x} & \cdots & D_{ijkx}+D_{ijkx} \end{pmatrix}, \tag{5.29}$$

where D_{ijkx} stands for the x-th element of $\boldsymbol{D}_{ijk}$.

Now the second order derivate of f can be represented as follows.

$$\frac{\partial^2 f}{\partial^2 \boldsymbol{\sigma}_t} = \begin{pmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,NM} \\ h_{2,1} & h_{2,2} & \cdots & h_{2,NM} \\ \vdots & \vdots & \cdots & \vdots \\ h_{NM,1} & h_{NM,2} & \cdots & h_{NM,NM} \end{pmatrix}, \tag{5.30}$$

where each element is given by

$$\begin{aligned} h_{xy} = &- \sum_{i=1}^{N}\sum_{j=1}^{N} I_{ij} I_0 \ \mathrm{sum}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij} \odot \boldsymbol{\beta}_{ijx} \odot \boldsymbol{\beta}_{ijy}) + \\ &\sum_{i=1}^{N}\sum_{j=1}^{N} I_0^2 \ \mathrm{sum}[(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T \odot \boldsymbol{\gamma}_{ijx} \odot \boldsymbol{\gamma}_{ijy}]. \end{aligned} \tag{5.31}$$

5.2.2 The proposed new efficient formulation

The problem of the previous old formulations of Jacobian and Hessian shown above is the computation cost increasing as the number N_{ij} of paths increases. As discussed later, the computation cost is $O(N_{ij}^2)$ on average.

The idea of the proposed formulation is to explore the property of the exponential function and its derivative in the problem. As shown below, the computation cost can be reduced to $O(N_{ij})$ on average.

Jacobian: new formulation

First we rewrite the cost function as follows.

$$f = \sum_{i=1}^{N}\sum_{j=1}^{N} r_{ij}^2, \tag{5.32}$$

where r_{ij} is a residual,

$$r_{ij} = I_{ij} - I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}} = I_{ij} - I_0\, \boldsymbol{E}_{ij}^T \boldsymbol{H}_{ij}. \tag{5.33}$$

Now we use the chain rule of differentiation, we have

$$\frac{\partial f}{\partial \boldsymbol{\sigma}_t} = \sum_{i=1}^{N}\sum_{j=1}^{N} 2 r_{ij} \frac{\partial r_{ij}}{\partial \boldsymbol{\sigma}_t}, \tag{5.34}$$

where

$$\frac{\partial r_{ij}}{\partial \boldsymbol{\sigma}_t} = I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}} \boldsymbol{D}_{ijk} = I_0 D_{ij} (\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}). \tag{5.35}$$

Here we define [1]

$$D_{ij} = \left(\boldsymbol{D}_{ij1}, \boldsymbol{D}_{ij2}, \cdots, \boldsymbol{D}_{ijN_{ij}} \right), \tag{5.36}$$

which has N_{ij} vectors of dimension NM, and $(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})$ is a coefficient vector. Therefore,

$$\frac{\partial f}{\partial \boldsymbol{\sigma}_t} = \sum_{i=1}^{N}\sum_{j=1}^{N} 2 I_0 (I_{ij} - I_0\, \boldsymbol{E}_{ij}^T \boldsymbol{H}_{ij})\, D_{ij}\, (\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}). \tag{5.37}$$

Discussion

Suppose that the expectation of the number of paths is $\bar{N} = E[N_{ij}]$. Then the computations for the proposed new formulation of Jacobian above are:

[1] Note that this is not the same with the one defined in the previous approaches above, which is a structure used in MATLAB codes to store $\boldsymbol{D}_{ijk}$. Here D_{ij} is an $(NM) \times N_{ij}$ matrix.

- $O(\bar{N})$ multiplications for $\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}$,
- $O(\bar{N})$ additions for $\boldsymbol{E}_{ij}^T \boldsymbol{H}_{ij}$,
- $O(\bar{N}NM)$ multiplications and additions for $D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})$ because there are $O(\bar{N})$ vectors of dimension NM,

for each i and j. Totally, it takes $O(\bar{N}N^3M)$ operations to compute NM elements of the Jacobian, or $O(\bar{N}N^2)$ operations per element.

Contrary, for each i and j the previous old formulation Eq.(5.27) needs:

- $O(\bar{N})$ multiplications for $\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}$,
- $O(\bar{N}NM)$ multiplications for $(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}) \otimes D_{ij}$ because there are $O(\bar{N})$ vectors of dimension NM,
- $O(\bar{N}NM)$ additions for $\text{sum}[(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}) \otimes D_{ij}]$,

for the first term, and

- $O(\bar{N}^2)$ multiplications for $(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T$,
- $O(\bar{N}^2NM)$ additions for computing $\widetilde{D_{ij}}$ because there are $O(\bar{N}^2)$ vectors of dimension NM,
- $O(\bar{N}^2NM)$ multiplications for $((\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T) \otimes \widetilde{D_{ij}}$,
- $O(\bar{N}^2NM)$ additions for $\text{sum}[((\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T) \otimes \widetilde{D_{ij}}]$,

for the second term. Totally, it takes $O(\bar{N}^2N^3M)$ operations to compute NM elements of the Jacobian, or $O(\bar{N}^2N^2)$ operations per element. The difference is mainly caused by the second term of Eq.(5.27).

In summary, the proposed new formulation has the cost of $O(\bar{N}N^2)$ operations per element, while the previous old formulation has the cost of $O(\bar{N}^2N^2)$ operations per element. Table 5.1 summarizes the discussion above.

Hessian: new formulation

In the same manner, we can derive the Hessian as follows. From the Jacobian

$$\frac{\partial f}{\partial \boldsymbol{\sigma}_t} = \sum_{i=1}^{N} \sum_{j=1}^{N} 2 r_{ij} \frac{\partial r_{ij}}{\partial \boldsymbol{\sigma}_t}, \tag{5.38}$$

Table 5.1 Comparison of the new and old formulations for computing the Jacobian.

terms	new	old
$\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}$	$O(\bar{N})$	$O(\bar{N})$
$\boldsymbol{E}_{ij}^T \boldsymbol{H}_{ij}$	$O(\bar{N})$	
$D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})$	$O(\bar{N}NM)$	
$(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}) \odot D_{ij}$		$O(\bar{N}NM)$
$\mathrm{sum}[(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}) \odot D_{ij}]$		$O(\bar{N}NM)$
$(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T$		$O(\bar{N}^2)$
$\widetilde{D_{ij}}$		$O(\bar{N}^2NM)$
$((\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T) \odot \widetilde{D_{ij}}$		$O(\bar{N}^2NM)$
$\mathrm{sum}[((\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T) \odot \widetilde{D_{ij}}]$		$O(\bar{N}^2NM)$
per element	$O(\bar{N}N^2)$	$O(\bar{N}^2N^2)$

we obtain the Hessian by using the chain rule of differentiation;

$$\frac{\partial^2 f}{\partial \boldsymbol{\sigma}_t^2} = \sum_{i=1}^{N} \sum_{j=1}^{N} 2 \frac{\partial r_{ij}}{\partial \boldsymbol{\sigma}_t} \frac{\partial r_{ij}}{\partial \boldsymbol{\sigma}_t}^T + 2 r_{ij} \frac{\partial^2 r_{ij}}{\partial \boldsymbol{\sigma}_t^2}, \tag{5.39}$$

where

$$\frac{\partial^2 r_{ij}}{\partial \boldsymbol{\sigma}_t^2} = -I_0 \sum_{k=1}^{N_{ij}} H_{ijk} e^{-\boldsymbol{\sigma}_t^T \boldsymbol{D}_{ijk}} \boldsymbol{D}_{ijk} \boldsymbol{D}_{ijk}^T \tag{5.40}$$

$$= -I_0 D_{ij} \mathrm{diag}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}) D_{ij}^T. \tag{5.41}$$

Now the Hessian can be written as follows:

$$\begin{aligned} \frac{\partial^2 f}{\partial \boldsymbol{\sigma}_t^2} = & \sum_{i=1}^{N} \sum_{j=1}^{N} 2 I_0^2 D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}))^T \\ & - 2 I_0 \left(I_{ij} - I_0 \boldsymbol{E}_{ij}^T \boldsymbol{H}_{ij}\right) D_{ij} \mathrm{diag}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}) D_{ij}^T. \end{aligned} \tag{5.42}$$

Note that $D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}))^T$ should not be expanded like as $D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T D_{ij}^T$ because it involves a large matrix $(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T$ which is computationally intensive to compute.

Discussion

By reusing $\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}$ and $D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})$ computed for the Jacobian, the new formulation of Hessian needs

- $O(N^2M^2)$ multiplications for $D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}))^T$,

Table 5.2 Comparison of the new and old formulations for computing the Hessian.

terms	new	old
$D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(D_{ij}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij}))^T$	$O(N^2M^2)$	
$D_{ij}\mathrm{diag}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})$	$O(\bar{N}NM)$	
$D_{ij}\mathrm{diag}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})D_{ij}^T$	$O(\bar{N}N^2M^2)$	
$(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij} \odot \boldsymbol{\beta}_{ijx} \odot \boldsymbol{\beta}_{ijy})$		$O(\bar{N})$
$(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T$		$O(\bar{N}^2)$
$(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T \odot \boldsymbol{\gamma}_{ijx} \odot \boldsymbol{\gamma}_{ijy}$		$O(\bar{N}^2)$
sums of i, j		$O(N^2\bar{N}^2)$
per element	$O(\bar{N}N^2)$	$O(N^2\bar{N}^2)$

- $O(\bar{N}NM)$ multiplications for $D_{ij}\mathrm{diag}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})$,
- $O(\bar{N}N^2M^2)$ multiplications for $D_{ij}\mathrm{diag}(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})D_{ij}^T$,

for each i and j. Totally, it takes $O(\bar{N}N^4M^2)$ operations to compute N^2M^2 elements of the Hessian, or $O(\bar{N}N^2)$ operations per element.

Contrary, for each element the previous formulation (5.31) needs:

- $O(\bar{N}^2)$ multiplications for $\boldsymbol{\gamma}_{ijx}$,
- $O(\bar{N})$ multiplications and the sum for $(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij} \odot \boldsymbol{\beta}_{ijx} \odot \boldsymbol{\beta}_{ijy})$,
- $O(\bar{N}^2)$ multiplications for $(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T$,
- $O(\bar{N}^2)$ multiplications and the sum for $(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})(\boldsymbol{E}_{ij} \odot \boldsymbol{H}_{ij})^T \odot \boldsymbol{\gamma}_{ijx} \odot \boldsymbol{\gamma}_{ijy}$,
- $O(\bar{N}^2N^2)$ additions for the sums of i and j.

Totally, it takes $O(\bar{N}^2N^2)$ operations to compute a single element of the Hessian.

In summary, the proposed new formulation has the cost of $O(\bar{N}N^2)$ operations per element, while the previous old formulation has the cost of $O(N^2\bar{N}^2)$ operations per element. Table 5.2 summarizes the discussion above.

5.3 Numerical simulation

In this section, we report the results obtained by simulations using the proposed method by comparing primal-dual and log barrier interior point methods, as well as old and new formulations of Jacobian and Hessian.

Since the mathematical model we used to describe the light transport in the forward problem is exactly the same with the model in the previous work [71], we use the same setup as follows. For the 2D layered medium, the grid size was set to $N = M = 24$ with square voxels of size 1 [mm], i.e. the medium is 24 [mm] $\times$ 24 [mm]. The values of the extinction coefficients are set between 1.05 and 1.55 [mm^{-1}], and the lower and upper bounds (σ_{tl} and σ_{tu}) are set to be 1.0 and 2.0 [mm^{-1}], respectively. Values of the initial guess are 1.001 for all elements of $\boldsymbol{\sigma}_{t0}$, as well as $\boldsymbol{s}_0$ and $\boldsymbol{z}_0$. Parameters used in Alg.4 are set as $\sigma = 0.5$, $\eta = 0.01$, $\varepsilon_\mu = 1$, and $\varepsilon_{TOL} = 0.02$.

5.3.1 Estimation quality

The ground truth and the estimated results are shown in Figure 5.1. The matrix plots in the top row of the figure represent five different media (from a to e) used for the simulation, which were also used in the previous work [71]. Note that the medium e is the Shepp-Logan phantom [64]. Each voxel is shaded in gray according to the values of the extinction coefficients.

The following rows show the estimated results of different combinations of log-barrier (LB) or primal-dual (PD) methods, Old or New formulas, and Newton's or Quasi-Newton methods. The proposed method is PD-New-Newton/Quasi-Newton; the use of primal-dual method with Newton's or Quasi-Newton method by using the proposed new formulation. The row LB-Old-Quasi-Newton corresponds to the previous work [71] that uses the log-barrier method with Quasi-Newton method by using the old formulation, and the row LB-Old-Newton corresponds to another prior work [65] .

As we can see, the results of different combinations almost look the same for each of the five media. This observation is also validated by the root mean square error (RMSE) shown in Table 5.3. The RMSE values of all combinations are more or less the same, while some variations appears due to the different update rules between Newton's and Quasi-Newton methods, and different stopping conditions between log-barrier and primal-dual methods.

5.3.2 Estimation time

The main goal of this paper is to develop an efficient way to solve the inverse problem. Table 5.3 shows the computation cost of different combinations. All experiments were performed on a Linux workstation (two Intel Xeon E5-2630 2.4GHz CPUs, total 16 physical cores, with 256 GB memory). We implemented the method in MATLAB R2017a, and did not explicitly use the parallel computation toolbox of MATLB except the Hessian computation of

Fig. 5.1 Numerical simulation results for five media (a–e) in a grid of 24×24. Darker shades of gray represent larger values of extinction coefficients (more light is absorbed at the voxel). The bars on the side show extinction coefficient values in gray scale. The first row shows ground truth for five different types of media used for the simulation. The following rows show the estimated results for different combinations of log-barrier (LB) or primal-dual (PD) methods, Old or New formulas, and Newton's or Quasi-Newton methods.

LB/PD-Old-Newton due to its slow computation. Parallel matrix multiplications are however automatically performed on MATLAB. Table 5.3 show the computation time for different computation in seconds. We report the average and standard deviation of 10 trials, except the cases of LB/PD-Old-Newton which show the processing time of a single trial.

For any combination, our proposed new formulation is much faster than the old formulation. The uses of Newton's method greatly benefit from the efficient Hessian computation and the computation time reduces more than factor of 100. However the new formulation do not help to reduce the computation cost of Quasi-Newton methods and the reduction is factor of just 2 or 3. This is due to the fact that the Quasi-Newton method needs gradient vectors, and its computation is order of NM (the number of voxels) and it is not so large in terms of the total computation cost. In contrast, Hessian computation in the Newton's method is of the size $NM \times NM$, which is quite large compare to the gradient. Our new formulation is therefore better when the Newton' method is used.

With the Quasi-Newton method, the primal-dual approach seems to be comparable to the log-barrier method. By comparing rows LB/PD-New-QuasiNewton, log-barrier is faster than primal-dual for denser media (c, d, and e). This might be caused by the different way of approximations by the Quasi-Newton method. For the log-barrier method, the gradient is modified by the approximated Hessian. For the primal-dual method, however the approximated Hessian is used in the matrix to be solved, resulting in an update rule of $\boldsymbol{p}_\sigma$ regularized by diagonal elements of $\boldsymbol{w}$ in Eq.(5.16). Except the simplest medium (a), the fast combination is PD-New-Newton, which is proposed in this paper. This is due to the fast convergence of the Newton's method compared to the Quasi-Newton method, and also the fact that the primal-dual method needs fewer iterations than the log-barrier method. The qualities of results are almost the same as discussed above, then PD-New-Newton is the best when the working memory is enough for storing the Hessian. Otherwise, LB/PD-New-QuasiNewton are better to be used.

5.3.3 Comparison

We compare our methods to a standard DOT wit FEM implemented in the Electrical Impedance Tomography and Diffuse Optical Tomography Reconstruction Software (EIDORS) [3, 4]. in the same setting with the previous work [71]: $N = M = 24$ medium of size 24 (mm) $\times$ 24 (mm) with the five media (a to e). For solving DOT by EIDORS, we used $24 \times 24 \times 24 = 1152$ triangle elements. For boundary condition we placed 48 light sources and detectors at the same intervals around the medium. We used some different solvers and

Table 5.3 RMSEs and computation time for the numerical simulations for five different types of media (a–e) with grid size of 24×24, for different combinations of log-barrier (LB) or primal-dual (PD) methods, Old or New formulas, and Newton's or Quasi-Newton methods. Each computation time shows the mean and standard deviation of 10 trials, except the combinations of "Old-Newton". Note that RMSE values are exactly the same for 10 trials. Results of diffuse optical tomography (DOT) methods are shown for comparison.

		a	b	c	d	e
RMSE	LB-New-Newton	0.008422	0.012643	0.014594	0.021246	0.052511
	LB-New-QuasiNewton	0.008646	0.012478	0.014444	0.020375	0.049811
	LB-Old-Newton	0.008422	0.012643	0.014594	0.021246	0.052420
	LB-Old-QuasiNewton	0.008646	0.012478	0.014444	0.020375	0.049818
	PD-New-Newton	0.009776	0.013190	0.014490	0.021251	0.055912
	PD-New-QuasiNewton	0.009754	0.013184	0.014502	0.021201	0.056085
	PD-Old-Newton	0.009776	0.013190	0.014490	0.021251	0.055912
	PD-Old-QuasiNewton	0.009754	0.013184	0.014502	0.021201	0.056084
	DOT (GN, Laplace prior)	0.059339	0.062984	0.078100	0.065001	0.087094
	DOT (GN, NOSER prior)	0.052053	0.057515	0.075478	0.059156	0.086397
	DOT (GN, Tikhonov prior)	0.054729	0.056196	0.073146	0.059284	0.087659
	DOT (primal-dual, TV prior)	0.055047	0.059219	0.081811	0.070263	0.086107
Computation time [s]	LB-New-Newton	60.00 ± 4.60	57.63 ± 1.41	61.90 ± 2.88	62.32 ± 1.22	93.10 ± 2.46
	LB-New-QuasiNewton	18.64 ± 0.90	17.22 ± 1.03	25.32 ± 1.21	20.73 ± 1.10	32.57 ± 0.58
	LB-Old-Newton	126100	12848	13383	14037	21577
	LB-Old-QuasiNewton	44.86 ± 1.33	42.58 ± 1.19	63.75 ± 1.76	54.05 ± 2.04	91.17 ± 1.48
	PD-New-Newton	18.73 ± 2.18	16.52 ± 0.67	17.40 ± 0.93	18.28 ± 1.30	23.14 ± 1.32
	PD-New-QuasiNewton	14.44 ± 0.78	13.07 ± 0.61	40.25 ± 1.01	30.77 ± 0.86	48.26 ± 1.59
	PD-Old-Newton	5673	5418	5824	5663	7547
	PD-Old-QuasiNewton	75.67 ± 1.34	67.18 ± 1.38	203.42 ± 3.42	155.69 ± 2.67	248.05 ± 4.86
	DOT (GN, Laplace prior)	0.34 ± 0.04	0.41 ± 0.04	0.40 ± 0.03	0.40 ± 0.04	0.40 ± 0.03
	DOT (GN, NOSER prior)	0.52 ± 0.04	0.52 ± 0.05	0.53 ± 0.05	0.50 ± 0.01	0.52 ± 0.03
	DOT (GN, Tikhonov prior)	0.29 ± 0.01	0.29 ± 0.02	0.30 ± 0.03	0.29 ± 0.01	0.29 ± 0.02
	DOT (primal-dual, TV prior)	2.67 ± 0.07	2.68 ± 0.06	2.67 ± 0.07	2.64 ± 0.05	2.66 ± 0.07

priors; Gauss-Newton method [2] with Laplace, NOSER [19], and Tikhonov priors, and primal-dual method with total variation prior.

Due to the diffusion approximation of DOT, the results for DOT wit Gauss-Newton method are blurred, and those for DOT wit PD have a tendency of overestimating the hight coefficient value areas. In contrast, the results of PD-New-Newton (and other combinations in Figure 5.1) are more clear and sharp for all combinations. This observation is also validated by the root mean square error (RMSE) shown in Table 5.3. The RMSE values of PD-New-Newton are smaller than the values of DOT for all five media.

The obvious drawback of PD-New-Newton is its high computation cost. It is slower by factor of 10 compared to DOT with primal-dual method, and factor of 100 to DOT with Gauss-Newton method. A large part of the computation cost comes from the computation of Hessian and Jacobian, which depends on the number of paths N_{ij}. Further reduction of computation cost is left for our future work.

5.4 Conclusion

In this chapter, we have proposed a primal-dual approach to optical tomography with discretized path integral, and also efficient formulation for computing Hessian and Jacobian. Numerical simulation examples with 2D layered media are shown to demonstrate that the proposed method, called PD-New-Newton in the experiments, performed faster than the previous work (LB-Old-QuasiNewton) while the estimated extinction coefficients of the both method were comparable. Compared to DOT, the proposed method worked slower but produced better estimation results in terms of RMSE.

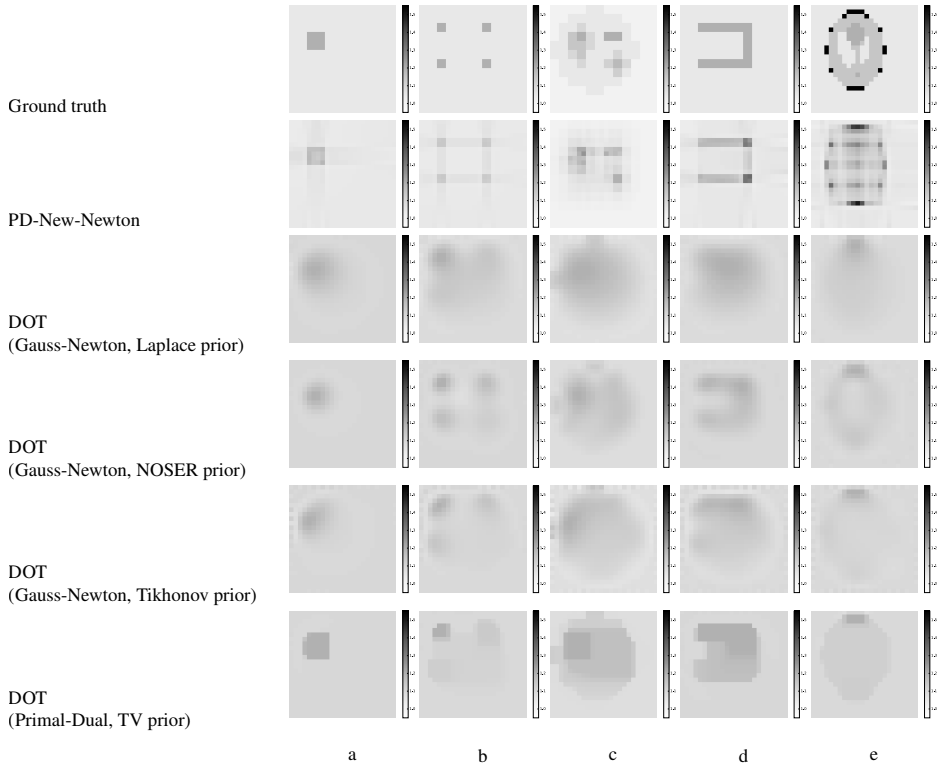


Fig. 5.2 Numerical simulation results for five media (a–e) in a grid of 24×24. Darker shades of gray represent larger values of extinction coefficients (more light is absorbed at the voxel). The bars on the side show extinction coefficient values in gray scale. The first row shows ground truth for five different types of media used for the simulation. The following rows show the estimated results for different combinations of log-barrier (LB) or primal-dual (PD) methods, Old or New formulas, and Newton's or Quasi-Newton methods. Results of the previous work [71] and diffuse optical tomography (DOT) are shown as baselines in the last three rows.

Chapter 6

Conclusion

In this thesis, we have developed an optical tomography method using discretized path integral as a forward model and solving a non-linear inverse problem with Log-barrier and Primal-dual interior point approaches. Optical tomography is urgently needed in medical area due to the fact that other tomography methods have their own disadvantages. For example, the radiation exposure in the X-ray Computed Tomography (CT) has always been a issue since the X-ray CT was invented; the Nuclear Magnetic Resonance Imaging (NMRI) has very limited working area.

In the forward problem , we use path integral to build a mathematical model for the light transport in the optical tomography. To simplify the model, as the first step, we discretizied path integral. The second step, we designed a Two-Dimesional layered material with specific scattering model. The third step, the phase function was approximated by a Gaussian model. After the above three steps, we obtained the forward model with a simplified expression.

Once we got the forward model, we formulated the inverse problem which is a inequality optimization problem with boxed constraints.

To solve the inverse problem, we implemented the Log-barrier interior point approach. The numerical results show our approach works well. The comparison results with DOT also point out our approach has advantage in accuracy.

In order to improve the performance, we introduced the Primal-dual interior point approach. We further form new efficient formulations for computing Jacobian and Hessian which consider as the most computational expensive partin the cost function of the constraint non-linear optimization problem. Numerical experiments show that Primal-dual approach works well with the new efficient formulations, and the computation time was reduced four times than the Log-barrier interior point approach while the quality of the estimation results maintain the same.

In our future work, the assumptions we made to the forward model should be released. And the scattering coefficient should also be estimated during the method. And to further reduce the computation cost, the platform should also be transfer to C++.

References

[1] Adams, M. L. and Larsen, E. W. (2002). Fast iterative methods for discrete-ordinates particle transport calculations. *Progress in Nuclear Energy*, 40(1):3 – 159.

[2] Adler, A. and Guardo, R. (1996). Electrical impedance tomography: regularized imaging and contrast detection. *IEEE Transactions on Medical Imaging*, 15(2):170–179.

[3] Adler, A. and Lionheart, W. R. (2005). Eidors: Towards a community-based extensible software base for eit. In *6th Conference on Biomedical Applications of Electrical Impedance Tomography, London, UK*, pages 1–4.

[4] Adler, A. and Lionheart, W. R. (2006). Uses and abuses of eidors: an extensible software base for eit. *Physiological measurement*, 27(5):S25.

[5] Antyufeev, V. S. (2000). *Monte Carlo Method for Solving Inverse Problems of Radiation Transfer*. Inverse and Ill-Posed Problems Series. VSP.

[6] Antyufeev, V. S. (2012). On the distribution of a random variable. *Sib. Zh. Ind. Mat.*, 15(2):3–10.

[7] Arridge, S. R. (1999). Optical tomography in medical imaging. *Inverse Problems*, 15(R41–93).

[8] Arridge, S. R. and Hebden, J. C. (1997). Optical imaging in medicine: II. modelling and reconstruction. *Physics in Medicine and Biology*, 42(5):841.

[9] Arridge, S. R. and Schotland, J. C. (2009). Optical tomography: forward and inverse problems. *Inverse Problems*, 25(12):123010.

[10] Arridge, S. R. and Schweiger, M. (1997). Image reconstruction in optical tomography. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 352(1354):717–726.

[11] Baba, M., Ishimaru, K., Hiura, S., Furukawa, R., Miyazaki, D., and Aoyama, M. (2014). Estimation of scattering properties of participating media using multiple-scattering renderer,. In *Proceedings of The Fourth IIEEJ International Workshop on Image Electronics and Visual Computing*.

[12] Bal, G. (2009). Inverse transport theory and applications. *Inverse Problems*, 25(5):053001.

[13] Barbour, R. L., Graber, H. L., Aronson, R., and Lubowsky, J. (1990). Model for 3-D optical imaging of tissue. In *Int. Geosci. and Remote Sensing Symp. (IGARSS)*, volume 2, pages 1395–1399.

[14] Blasi, P., Saec, B., and Schlick, C. (1993). A rendering algorithm for discrete volume density objects. In *Computer Graphics Forum*, volume 12, pages 201–210. Wiley Online Library.

[15] Boas, D. A., Brooks, D. H., Miller, E. L., DiMarzio, C. A., Kilmer, M., Gaudette, R. J., and Zhang, Q. (2001). Imaging the body with diffuse optical tomography. *Signal Processing Magazine, IEEE*, 18(6):57–75.

[16] Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambrige Univesity Press.

[17] Cerezo, E., Pérez, F., Pueyo, X., Seron, F. J., and Sillion, F. X. (2005). A survey on participating media rendering techniques. *The Visual Computer*, 21(5):303–328.

[18] Charette, A., Boulanger, J., and Kim, H. K. (2008). An overview on recent radiation transport algorithm development for optical tomography imaging. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 109(17–18):2743 – 2766.

[19] Cheney, M., Isaacson, D., Newell, J. C., Simske, S., and Goble, J. (1990). NOSER: An algorithm for solving the inverse conductivity problem. *International Journal of Imaging Systems and Technology*, 2(2):66–75.

[20] Cornette, W. M. and Shanks, J. G. (1992). Physically reasonable analytic expression for the single-scattering phase function. *Appl. Opt.*, 31(16):3152–3160.

[21] Dobashi, Y., Iwasaki, W., Ono, A., Yamamoto, T., Yue, Y., and Nishita, T. (2012). An inverse problem approach for automatically adjusting the parameters for rendering clouds using photographs. *ACM Transactions on Graphics*, 31(6).

[22] Dutre, P., Bala, K., Bekaert, P., and Shirley, P. (2003). *Advanced global illumination*. A. K. Peters, Ltd.

[23] Edström, P. (2005). A fast and stable solution method for the radiative transfer problem. *SIAM Review*, 47(3):pp. 447–468.

[24] Edström, P. (2008). A two-phase parameter estimation method for radiative transfer problems in paper industry applications. *Inverse Problems in Science and Engineering*, 16(7):927–951.

[25] Edström, P. (2010). Simulation and modeling of light scattering in paper and print applications. In Kokhanovsky, A. A., editor, *Light Scattering Reviews 5*, Springer Praxis Books, pages 451–475. Springer Berlin Heidelberg.

[26] Feynman, R. P. and Hibbs, A. R. (1965). *Quantum Mechanics and Path Integrals*. McGraw-Hill.

[27] Florescu, L., Markel, V. A., and Schotland, J. C. (2010). Single-scattering optical tomography: Simultaneous reconstruction of scattering and absorption. *Physical Review E*, 81:016602.

[28] Florescu, L., Schotland, J. C., and Markel, V. A. (2009). Single-scattering optical tomography. *Physical Review E*, 79:036607–1–10.

[29] Gibson, A. P., Hebden, J. C., and Arridge, S. R. (2005). Recent advances in diffuse optical imaging. *PHYSICS IN MEDICINE AND BIOLOGY*, 50:R1–R43.

[30] Gkioulekas, I., Zhao, S., Bala, K., Zickler, T., and Levin, A. (2013). Inverse volume rendering with material dictionaries. *ACM Trans. Graph.*, 32(6):162:1–162:13.

[31] Hayakawa, C. and Spanier, J. (2004). Perturbation monte carlo methods for the solution of inverse problems. In Niederreiter, H., editor, *Monte Carlo and Quasi-Monte Carlo Methods 2002*, pages 227–241. Springer Berlin Heidelberg.

[32] Hayakawa, C. K., Spanier, J., and Venugopalan, V. (2007). Coupled forward-adjoint monte carlo simulations of radiative transport for the study of optical probe design in heterogeneous tissues. *SIAM Journal on Applied Mathematics*, 68(1):pp. 253–270.

[33] Hebden, J. C., Arridge, S. R., and Delpy, D. T. (1997). Optical imaging in medicine: I. experimental techniques. *Physics in Medicine and Biology*, 42(5):825.

[34] Heino, J., Arridge, S., Sikora, J., and Somersalo, E. (2003). Anisotropic effects in highly scattering media. *Physical Review E*, 68:031908–1–8.

[35] Henyey, L. G. and Greenstein, J. L. (1941). Diffuse radiation in the galaxy. *Astrophysical Journal*, 93:70–83.

[36] Ishii, Y., Arai, T., Mukaigawa, Y., Tagawa, J., and Yagi, Y. (2013). Scattering tomography by monte carlo voting. In *IAPR International Conference on Machine Vision Applications*.

[37] Jacques, S. L. and Wang, X. (1997). Path integral description of light transport in tissues. In *Proc. SPIE, Optical Tomography and Spectroscopy of Tissue: Theory, Instrumentation, Model, and Human Studies II*, volume 2979, pages 488–499.

[38] Jarosz, W. (2008). *Efficient Monte Carlo Methods for Light Transport in Scattering Media*. PhD thesis, University of California, San Diego.

[39] Jensen, H. W. (2001). *Realistic image synthesis using photon mapping*. AK Peters, Ltd.

[40] Joshi, N., Donner, C., and Jensen, H. W. (2006). Noninvasive measurement of scattering anisotropy in turbid materials by nonnormal incident illumination. *Opt. Lett.*, 31(7):936–938.

[41] Kajiya, J. T. (1986). The rendering equation. *SIGGRAPH Comput. Graph.*, 20(4):143–150.

[42] Keijzer, M., Jacques, S. L., Prahl, S. A., and Welch, A. J. (1989). Light distributions in artery tissue: Monte carlo simulations for finite-diameter laser beams. *Lasers in surgery and medicine*, 9(2):148–154.

[43] Kokhanovsky, A. A. (2004). *Light Scattering Media Optics: Problems and Solutions*. Springer, third edition.

[44] Kristensen, A. W. (2011). *Efficient Unbiased Rendering using Enlightened Local Path Sampling*. PhD thesis, Technical University of Denmark.

[45] Kurachi, N. (2011). *The Magic of Computer Graphics*. A. K. Peters, Ltd., Natick, MA, USA.

[46] Křivánek, J., Georgiev, I., Hachisuka, T., Vévoda, P., Šik, M., Nowrouzezahrai, D., and Jarosz, W. (2014). Unifying points, beams, and paths in volumetric light transport simulation. *ACM Trans. Graph.*, 33(4):103:1–103:13.

[47] Lafortune, E. P. and Willems, Y. D. (1996). Rendering participating media with bidirectional path tracing. In *Proceedings of the Eurographics Workshop on Rendering Techniques '96*, pages 91–100, London, UK, UK. Springer-Verlag.

[48] Marchuk, G. I., Mikhailov, G. A., Nazaraliev, M. A., Darbinjan, R. A., Kargin, B. A., and Elepov, B. S. (1980). Monte carlo methods for solving direct and inverse problems of the theory of radiative transfer in a spherical atmosphere. In *The Monte Carlo Methods in Atmospheric Optics*, volume 12 of *Springer Series in Optical Sciences*, pages 54–146. Springer Berlin Heidelberg.

[49] McCormíck, N. J. (1984). Recent developments in inverse scattering transport meteods. *Transport Theory and Statistical Physics*, 13(1-2):15–28.

[50] Nishita, T., Sirai, T., Tadamura, K., and Nakamae, E. (1993). Display of the earth taking into account atmospheric scattering. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '93, pages 175–182, New York, NY, USA. ACM.

[51] Nocedal, J. and Wright, S. J. (2006). *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, second edition.

[52] Pauly, M., Kollig, T., and Keller, A. (2000). Metropolis light transport for participating media. In *Eurographics Workshop on Rendering Techniques*, pages 11–22.

[53] Pharr, M. and Humphreys, G. (2010). *Physically based rendering: From theory to implementation*. Morgan Kaufmann.

[54] Polydorides, N. (2002). *Image reconstruction algorithms for soft-field tomography*. PhD thesis, University of Manchester: UMIST.

[55] Polydorides, N. and Lionheart, W. R. (2002). A matlab toolkit for three-dimensional electrical impedance tomography: a contribution to the electrical impedance and diffuse optical reconstruction software project. *Measurement Science and Technology*, 13(12):1871.

[56] Premože, S., Ashikhmin, M., and Shirley, P. (2003). Path integration for light transport in volumes. In *Proceedings of the 14th Eurographics workshop on Rendering*, pages 52–63. Eurographics Association.

[57] Raab, M., Seibert, D., and Keller, A. (2008). Unbiased global illumination with participating media. In *Monte Carlo and Quasi-Monte Carlo Methods 2006*, pages 591–605.

[58] Ren, K. (2010). Recent developments in numerical techniques for transport-based medical imaging methods. *Communications in Computational Physics*, 8:1–50.

[59] Ren, K., Bal, G., and Hielscher, A. H. (2006). Frequency domain optical tomography based on the equation of radiative transfer. *SIAM J. Sci. Comput.*, 28(4):1463–1489.

[60] Schotland, J. C. (2000). Tomography with diffusing photons: a feynman path integral perspective. In *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, volume 6, pages 3791–3794 vol.6.

[61] Schotland, J. C. (2011). Path integrals and optical tomography. *Contemporary Mathematics*, 548(77–84).

[62] Schweiger, M., Gibson, A., and Arridge, S. R. (2003). Computational aspects of diffuse optical tomography. *Computing in Science & Engineering*, 5(6):33–41.

[63] Segovia, B. (2007). *Interactive Light Transport with Virtual Point Lights*. Thèse de doctorat en informatique, Université Lyon 1.

[64] Shepp, L. A. and Logan, B. F. (1974). The fourier reconstruction of a head section. *IEEE Transactions on Nuclear Science*, 21(3):21–43.

[65] Tamaki, T., Yuan, B., Raytchev, B., Kaneda, K., and Mukaigawa, Y. (2013). Multiple-Scattering Optical Tomography with Layered Material. In *2013 International Conference on Signal-Image Technology & Internet-Based Systems*, pages 93–99. IEEE.

[66] Tessendorf, J. (1987). Radiative transfer as a sum over paths. *Physical review A*, 35(2):872–878.

[67] Veach, E. (1997). *Robust Monte Carlo methods for light transport simulation*. PhD thesis, Stanford University.

[68] Veach, E. and Guibas, L. J. (1997). Metropolis light transport. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, pages 65–76, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.

[69] Wilson, M. J. and Wang, R. K. (2001). A path-integral model of light scattered by turbid media. *Journal of Physics B: Atomic, Molecular and Optical Physics*, 34(8):1453.

[70] Yuan, B., Tamaki, T., Kurita, T., Raytchev, B., and Kaneda, K. (2016). Optical tomography with discretized path integrals: a comparison with log-barrier and primal-dual methods. In *The Korea-Japan joint workshop on Frontiers of Computer Vision (FCV2016)*, pages 378–382.

[71] Yuan, B., Tamaki, T., Kushida, T., Mukaigawa, Y., Kubo, H., Raytchev, B., and Kaneda, K. (2015a). Optical tomography with discretized path integral. *Journal of Medical Imaging*, 2(3):033501.

[72] Yuan, B., Tamaki, T., Kushida, T., Raytchev, B., Kaneda, K., Mukaigawa, Y., and Kubo, H. (2015b). Layered optical tomography of multiple scattering media with combined constraint optimization. In *2015 21st Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV)*, pages 1–6. IEEE.

[73] Yuan, B., Tamaki, T., Raytchev, B., and Kaneda, K. (2017). Primal-dual approach to optical tomography with discretized path integral with efficient formulations. *Journal of Medical Imaging*, 4(3):033501.