Approaches to Motion-Corrected PET Image Reconstruction from Respiratory Gated Projection Data

by

Matthew W. Jacobson

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in The University of Michigan
2006

Doctoral Committee:

Professor Jeffrey A. Fessler, Chair
Professor Alfred O. Hero III
Professor Charles R. Meyer
Professor Romesh Saigal

In memory of my grandfather, Isadore Shore. May he rest in peace.

ACKNOWLEDGEMENTS

I would like to express sincerest thanks to my advisor, Prof. Jeff Fessler, for his guidance and support, as well as for many enjoyable and thought-provoking discussions. Working for him has been a true privilege. I would also like to thank those who have served on my committee, Profs. Alfred Hero, Charles Meyer, Valen Johnson, and Romesh Saigal, for their efforts and contributions to the fruition of my dissertation.

I further wish to acknowledge the many colleagues I have had throughout the years, including my labmates at The University of Michigan: Web Stayman, Idris Elbakri, Sangtae Ahn, Jeongtae Kim, Nan Sotthiviriat, Anastasia Yendiki, Yingying Zhang, Rongping Zeng, Dan Ruan, Somesh Srivastava, Kim Khalsa, Hugo Shi, and Eran Bashan. Special mention also goes to Michael Zibulevsky, Kris Thielemans, and Pavel Kisilev for many years of inspiring collaboration.

Finally, I would like to thank my parents and my brother Jonah for their love, patience, and encouragement.

This work was supported in part by NIH/NCI grant 1P01 CA87634 and in part also by an EECS Departmental fellowship.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

CHAPTER 1

Introduction

In past years, the conventional practice in Positron Emission Tomography (PET) has been to reconstruct images while ignoring the effects of anatomical motion in the patient. When motion is ignored, a PET scanner acts something like a conventional optical camera with the shutter open: one obtains the superposition of images of the object as it appears in various phases of motion. Thus, the image exhibits a degree of blur that is related to the motion magnitude. Until recently, the drawbacks of ignoring object motion has been ignored due to limitations in the spatial resolution of early PET scanners. The magnitude of anatomical motion was not always substantial enough, in comparison to the scanner resolution, for motion correction to promise any benefit. However, with the improvement of scanner resolution over time, motion-related blur is becoming a limiting factor in the resolution achievable in PET reconstruction.

This problem has been compounded by changing trends in the application of PET. In early years, PET was applied mainly in the research of the brain, where the relatively small movements of the head made motion effects still easier to dismiss. Conversely, recent clinical PET practice has emphasized cancer treatment, and therefore thorax scans have become more prevalent. Thus, not only has scanner resolution increased, but interest has shifted over the years to the scan of anatomy where motion is much larger. As a result

of these factors, the recent tomographic imaging literature has seen much interest (e.g., [51, 62, 37, 66, 95, 80, 61, 96, 79]) in motion correction techniques.

In this dissertation, our work is motivated by the problem of reconstructing motion-corrected images of the thorax from respiratory gated (histogram mode) data and its potential benefits to lung cancer treatment. In acquiring respiratory gated data, one essentially makes a separate scan of the object for every different position of the lungs. In PET scans of the thorax, image intensity in the region of a lung lesion can be used as an indicator of malignancy and also, if it is malignant, of how well it is responding to treatment. Quantifying this intensity accurately is therefore desirable. However, blur associated with the motion of the lungs can degrade quantification accuracy, unless effective motion-correction measures are employed. The desire for an effective reconstruction method naturally leads to questions about which method, among various candidates, performs best. For any given method, questions then arise as to how that method can be efficiently implemented. This dissertation considers these various questions in three main parts.

The first part, dealing with the comparison of various reconstruction methods, is the subject of Chapter 4, where we look at various motion-correction approaches, that might be applied to the lung lesion quantification task. One of these methods is based on an original model, proposed by us, for the statistics of respiratory gated measurements. This model is derived by representing the PET image, as it appears in various phases of its motion, via image transformations of the kind widely considered in the image registration literature. A method that we call Joint Estimation with Deformation Modeling (JEDM) is used to derive both motion parameter and image intensity estimates simultaneously based on the full set of gated projection data. The other methods are based on the more adhoc strategy of first reconstructing separate images from each of the measurement gates and then deriving estimates of their motion in the image domain. These kinds of strategies are

intuitive extensions of conventional reconstruction methods and have been considered by various investigators (e.g., [51, 97, 96, 79]).

We compare these methods in terms of how accurately, on average, they quantify a lung lesion in a simulated torso scan. Pains were taken, in these simulations, to imitate realistic torso anatomy and motion. Accordingly, ground truth image intensity and motion parameter values were derived from gated CT images of an actual human patient. We also propose a strategy by which regularizing roughness penalties can be judiciously designed, with the aid of side information from a breath hold CT so as to avoid blurring the lesion.

The remainder of the dissertation addresses the cost function minimizations required by the methods of Chapter 4, and likely to be required by most other approaches as well.

In Chapter 5, we look at the computational cost of certain interpolation operations common to the different motion-corrected reconstruction methods in Chapter 4. We furthermore propose pre-approximation/pre-computation techniques to ease some of the computational burden of these operations. The impact of these acceleration measures on some of the minimization sub-problems required by the methods of Chapter 4 are tested.

Chapter 6 presents a body of theoretical analysis that we have done on the behavior of so-called Majorize-Minimize (MM) algorithms. This analysis is a stepping stone in ongoing efforts toward finding effective implementations of the methods in Chapter 4. Majorize-Minimize algorithms are a family of optimization algorithms in which a given cost function is montonically reduced by minimizing a sequence of majorizing approximations to that cost function. The design of these majorizing functions (which we call tangent majorants) is not highly systematic, and often requires problem-specific insights on the part of the algorithm designer. We initially had several ideas for the design of tangent majorants, applicable to the cost function minimizations involved in the motion correction methods of Chapter 4. These designs involved the coordinate block alternation

technique proposed in [32, 33]. However, the analyses in those works was not entirely general enough to include the form of the tangent majorants in our candidate algorithms. Moreover, in the motion correction application, it is natural to consider constrained parameter spaces of a form also not formally addressed in these works. This led us to embark on an extensive theoretical analysis that expands the range of block alternating MM algorithms that one could consider.

In the course of this work, we came to many other generalizations and insights as well, which we also present. Among them, we give what we believe to be the first generic analysis of the local region of convergence for MM algorithms employing connected tangent majorants.

The main contributions of this dissertation can be summarized as follows.

- An original statistical model for gated PET data and, associated with this, a statistically principled approach, namely the JEDM method, to simultaneous motion and activity image estimation (Sections 4.2 and 4.3).

- An empirical investigation of the average lesion quantification accuracies of different motion correction schemes (Section 4.5.2) using realistic simulated anatomy and motion. The comparison includes the JEDM method and some gate-wise reconstruction approaches that are natural to consider and also similar to those considered by other investigators.

- An approach to using breath hold CT side information to avoid over-blurring of cancerous lesions (Sections 4.4.5 and Section 4.5.4).

- Table-lookup methods to accelerate the interpolation operations in various motion estimation cost functions (Chapter 5).

- A generalization of previous convergence conditions for MM algorithms (Chapter 6).

The generalizations include,

  - With the technique of block alternation, the feasible sets can be closed, non-polyhedral, and the objective function need not be convex.
  - Minimization of the tangent majorants may be carried out over strict subsets of the feasible set.
  - Only once-differentiability is assumed on the cost function and the tangent majorants.
  - continuity conditions on the tangent majorants, required in previous analyses (e.g., [25, 102, 57, 33]) can be relaxed in favor of local curvature bounds. This permits more freedom in the way tangent majorants can vary from between iterations.

- An analysis of the region of local convergence of MM algorithms that use connected tangent majorants (Section 6.5). This property has some negative implications for MM, as it is popularly employed, but also positive possibilities for how it can be used to implement non-convex minimization strategies.

The above remarks describe the core material in this dissertation. Relevant background to this material is discussed in Chapter 2. Also, the reader will encounter optional sections throughout the dissertation whose titles are marked with an *. These sections delve into topics somewhat tangential to the main theme of the dissertation, and can be skipped without loss of continuity.

CHAPTER 2

Background and Preliminaries

2.1 Miscellaneous Mathematical Conventions

We list here some miscellaneous conventions to be used throughout this dissertation.

1. *Matrices vs. non-matrices.* Symbols for vectors and matrices will be in boldfaced type. Other quantities shall be in plain type. Thus for example $\boldsymbol{M}$ is a matrix, but M_{ij} is its ij-th entry.

2. *Euclidean Spaces.* The notation $\mathbb{R}^d$ shall denote the Euclidean space of real, length d column vectors.

3. *Vertical concatenation.* Ordered n-tuples shall be thought throughout as the vertical concatenation of those elements. Thus, given column vectors $\boldsymbol{x}$, $\boldsymbol{y}$, and scalar a, the ordered triple $(\boldsymbol{x}, a, \boldsymbol{y})$ will denote the column vector obtained by stacking $\boldsymbol{x}$ on top of a on top of $\boldsymbol{y}$. Similary, in the expression $f(\boldsymbol{x}, a, \boldsymbol{y})$, it is to be understood that f is a function of the column vector $(\boldsymbol{x}, a, \boldsymbol{y})$.

4. *Diagonal concatenation.* The operation $\mathrm{diag}\,\{A_i\}$ shall denote the diagonal concatenation of $\{A_i\}_{i=1}^n$, an ordered set of not necessarily square matrices, i.e.,

$$\mathrm{diag}\,\{A_i\} = \begin{bmatrix} A_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & A_n \end{bmatrix}.$$

In addition, given a vector $\boldsymbol{v}$ with components v_i

$$\mathcal{D}\,\{\boldsymbol{v}\} = \mathrm{diag}\,\{v_i\}$$

shall denote the diagonal matrix with $\boldsymbol{v}$ along the diagonal.

5. *Multiple interpretation of arrays.* In various situations, we will introduce vectors $\boldsymbol{v}$ whose components v_j are associated with the nodes of a multi-dimensional grid. In such cases, we shall freely and interchangeably index the components of this vector either using a single linear subscript, as in v_j, or using multiple subscripts, as in v_{mnp}, where the multiple subscripts denote the associated node coordinates. We shall also refer to these $\boldsymbol{v}$ at times as discrete images, projections, etc…or some other kind of discrete scalar field, even though we shall be manipulating $\boldsymbol{v}$ mathematically as a vector.

6. *Identity Matrices.* They will be denoted $\boldsymbol{I}$. When the dimensions are not self-evident $\boldsymbol{I}_n$ shall denote the $n \times n$ identity matrix.

7. *Kronecker Products.* The Kronecker tensor product of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ shall be denoted by

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} A_{11}\boldsymbol{B} & A_{12}\boldsymbol{B} \\ A_{21}\boldsymbol{B} & \ddots \end{bmatrix}.$$

8. *Generalized Summation and the* KL *Distance.* The generalized summation operator $\sum_i^*$ will denote the sum over i when i is a discrete index and the integral over

i when it is a continuous index. The following definition of the Kullback-Leibler distance KL distance, therefore,

$$\mathrm{KL}(a,b) \stackrel{\triangle}{=} \sum_i{}^{*}[b_i - a_i \log b_i] - \sum_i{}^{*}[a_i - a_i \log a_i] \tag{2.1}$$

will apply to both continuous or discrete functions a_i and b_i of i. Note that we use the convention $0 \log 0 = 0$.

9. *Gradient and Jacobian Matrices.* Given a length J column vector-valued function $\boldsymbol{f}(x_1, x_2, \ldots, x_K)$ with components $f_j(x_1, x_2, \ldots, x_K)$, the $K \times J$ gradient (matrix) of $\boldsymbol{f}$, denoted $\nabla \boldsymbol{f}$, has kj-th entry $[\nabla \boldsymbol{f}]_{kj} = \frac{\partial f_j}{\partial x_k}$. When $\boldsymbol{f}$ is scalar-valued, this is just the usual column gradient of $\boldsymbol{f}$. The operator denoted ∇^T will produce the transpose of the gradient matrix,

$$\nabla^T \boldsymbol{f} \stackrel{\triangle}{=} (\nabla \boldsymbol{f})^T$$

and is precisely the Jacobian matrix of $\boldsymbol{f}$.

10. *Set Cardinality.* Given a finite set S, the notation $|S|$ will denote its cardinality, i.e., the number of elements in S.

2.2 Background on Tomography

Tomography is a scanning method used in nuclear medicine in which photons are radiated through or emitted from an object (which, in practical settings, is a hospital patient) and detected by a scanner. The scanner consists of an array of detector cells that surround the object, and whose function is to measure the number of photons y_i passing through the object along certain rays $i = 1, 2, \ldots, N_{\mathrm{rays}}$. It is common to model these y_i as statistically independent Poisson random variables. We denote by $\boldsymbol{y}$, the vector with components y_i. Since $\boldsymbol{y}$ is a Poisson random vector, its distribution is characterized by its mean $\bar{\boldsymbol{y}}(\boldsymbol{\theta})$,

where $\boldsymbol{\theta}$ is an unknown parameter vector. The form of $\bar{\boldsymbol{y}}(\boldsymbol{\theta})$ depends on the specifics of the scanner and data acquistion physics. Typically, the components of $\boldsymbol{\theta}$ parametrize certain spatially (and possibly also temporally) varying functions that are informative about the inner workings and structure of the object. The aim in tomography is to deduce $\boldsymbol{\theta}$ from the y_i and thus to reconstruct images of these spatially varying quantities.

One example of a spatially varying quantity that clinicians wish to reconstruct is the *attenuation* image. The attenuation image $\mu(\vec{\boldsymbol{r}})$ measures the tendency of the object to absorb or deflect photons at location $\vec{\boldsymbol{r}}$ in the object. Attenuation is related to material composition, and so $\mu(\vec{\boldsymbol{r}})$ can be used to discern the location of different tissue types in a patient. Another example is the *activity* image $\lambda(\vec{\boldsymbol{r}})$, which is pertinent to emission tomography (see Section 2.2.2). Emission tomography is an imaging modality in which the source of the photons is a radioactive tracer substance that is pre-injected into the patient. The activity $\lambda(\vec{\boldsymbol{r}})$ measures the concentration of tracer at location $\vec{\boldsymbol{r}}$, and gives information about the amount of metabolic processing occuring at $\vec{\boldsymbol{r}}$.

These concepts are illustrated in Figure 2.1, which depicts a 2D cross-section of a cylindrical detector array with a hypothetical object. This detector geometry is common in Positron Emission Tomography (PET). In PET, photons passing through the object result from the decay of a radio-tracer that is pre-injected into the patient. Tracer decay events lead to the emission of photon pairs that travel approximately colinearly in opposite directions and are detected by pairs of opposing detectors centered around each projection ray. In emission tomography modalities, such as PET, the mean measurements $\bar{y}_i$ are influenced by both the attenuation and activity image. It is common to approximate the

attenuation and activity images by nearest neighbour interpolation

$$\mu(\vec{\boldsymbol{r}}) \approx \sum_j \mu_j v_j(\vec{\boldsymbol{r}})$$

$$\lambda(\vec{\boldsymbol{r}}) \approx \sum_j \lambda_j v_j(\vec{\boldsymbol{r}})$$

Here, each interpolator $v_j(\vec{\boldsymbol{r}})$ is the indicator function of a *voxel* (i.e., a small 3D box-shaped region) at sample location $\vec{\boldsymbol{r}}_j$. Thus μ is described by a parameter vector $\boldsymbol{\mu}$ with components $\mu_j = \mu(\vec{\boldsymbol{r}}_j)$ and similarly $\boldsymbol{\lambda}$ is a parameter vector describing λ. As indicated in Figure 2.1, this leads to a parametrization $\bar{y}_i(\boldsymbol{\theta})$ where $\boldsymbol{\theta}$ is a concatenation of $\boldsymbol{\mu}$, $\boldsymbol{\lambda}$ (see also Section 2.2.2), and possibly other parameters as well.

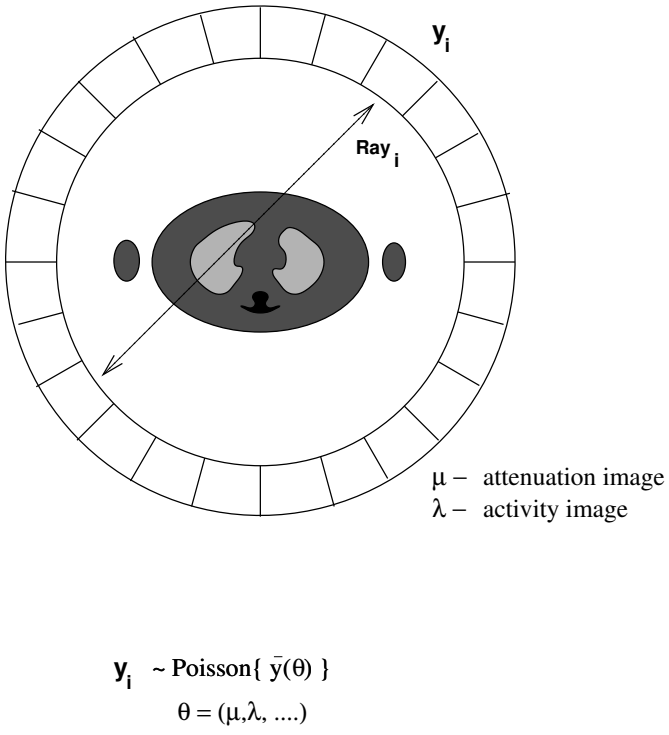

Figure 2.1: A 2D cross-section of a hypothetical object in a cylindrical tomographic system.

In the remainder of this section, we discuss the form of $\bar{y}_i(\boldsymbol{\theta})$ in a bit more detail for some common tomographic systems. We also describe some issues related to the reconstruction of images based on these models.

2.2.1 Transmission Tomography Basics

A transmission source is a source that is *external* to the object and transmits photons through it. Transmission tomography is a scanning protocol in which only a transmission source is used. The aim is to determine the *attenuation* $\mu(\vec{\boldsymbol{r}})$ at various sample locations $\vec{\boldsymbol{r}}_j$ in the object. The attenuation $\mu(\vec{\boldsymbol{r}})$ measures the tendency of an object to absorb photons at location $\vec{\boldsymbol{r}}$ and is related to the material composition of the object at that location. (It is also related to the energy of the source, although this energy level is usually known a priori).

As mentioned, one commonly approximates the continuous-space attenuation function $\mu(\vec{\boldsymbol{r}})$ using nearest-neighbour interpolation,

$$\mu(\vec{\boldsymbol{r}}) \approx \sum_j \mu_j v_j(\vec{\boldsymbol{r}}), \tag{2.2}$$

among a finite set of desired samples $\mu_j = \mu(\vec{\boldsymbol{r}}_j)$. Here, each interpolator $v_j(\vec{\boldsymbol{r}})$ is the indicator function of a voxel at sample location $\vec{\boldsymbol{r}}_j$. We shall refer to the column vector $\boldsymbol{\mu}$ with components μ_j as the (discrete) *attenuation image*.

Based on Beer's law and the approximation (2.2), the survival probability, i.e., the probability that a photon transmitted along ray i through the object (see Figure 2.1) passes *unabsorbed* to the surrounding detector array, is often modeled as $\exp(-[\boldsymbol{A}\boldsymbol{\mu}]_i)$. Here $\boldsymbol{A}$ is a matrix whose ij-th entry gives the length of intersection of ray i with voxel j, so that $[\boldsymbol{A}\boldsymbol{\mu}]_i$ is an approximate line integral through $\mu(\vec{\boldsymbol{r}})$ along i.

From the above considerations, the following, now classical model for the Poisson mean $\bar{y}_i(\boldsymbol{\mu})$ of transmission tomography measurements is obtained:

$$\bar{y}_i(\boldsymbol{\mu}) = b_i \exp(-[\boldsymbol{A}\boldsymbol{\mu}]_i) + r_i. \tag{2.3}$$

In (2.3), the $b_i \geq 0$ are so called *blank scan* data. They are calibration co-efficients obtained by running a transmission scan with no object present. The quantities $r_i \geq 0$ are the

mean of stray background photons that do not originate from the source. The r_i are often pre-determined by making detector measurements with no source present. The problem is now to use (2.3) to determine the unknown vector $\boldsymbol{\theta} = \boldsymbol{\mu}$.

2.2.2 Emission Tomography Basics

An emission source is one which emits photons from within the object, typically due to the decay of a radio-isotope (alternatively called a *radio-tracer*, or simply *tracer*) that has been introduced into it. Emission tomography is a scanning protocol in which only an emission source is used. In practical settings, the object is a living patient who has ingested or been injected with the radio-isotope. In non-practical settings, the scan may be conducted on a test object (or *phantom*) containing tracer-filled compartments. In emission tomography, the aim is to determine the radio-tracer concentration or *activity* $\lambda(\vec{\boldsymbol{r}})$ at various sample locations $\vec{\boldsymbol{r}}_j$. In living patients, $\lambda(\vec{\boldsymbol{r}})$ reflects metabolic activity occuring at $\vec{\boldsymbol{r}}$, and so is useful for measuring functional activity inside the body of the patient.

As mentioned, one commonly approximates the continuous-space activity function $\lambda(\vec{\boldsymbol{r}})$ using nearest-neighbour interpolation,

$$\lambda(\vec{\boldsymbol{r}}) \approx \sum_j \lambda_j v_j(\vec{\boldsymbol{r}}). \tag{2.4}$$

We refer to the column vector $\boldsymbol{\lambda}$ with components λ_j as the (discrete) *activity image*. A Poisson model for the mean of emission tomography measurements is

$$\bar{y}_i(\boldsymbol{\lambda}, \mu) = [\boldsymbol{P}(\boldsymbol{\mu})\boldsymbol{\lambda}]_i + r_i. \tag{2.5}$$

As in (2.3), r_i is the mean of the background counts detected along ray i. Also, $\boldsymbol{P}(\boldsymbol{\mu})$ is a matrix, depending on the attenuation image $\boldsymbol{\mu}$, whose ij-th element $P_{ij}(\boldsymbol{\mu})$ is the probability that a photon emission event in voxel j is detected along ray i.

The form of $P_{ij}(\boldsymbol{\mu})$ depends on the physics of both the emission and the acquisition processes. For example, in Positron Emission Tomography (PET), tracer decay results

in the emission of photon pairs that travel approximately colinearly in opposite directions and are detected by pairs of opposing detectors centered around each projection ray. Beer's law (see also Subsection 2.2.1) then leads to a model of the form,

$$P_{ij}(\boldsymbol{\mu}) = \exp(-[\boldsymbol{A}\boldsymbol{\mu}]_i)\varepsilon_i G_{ij} \tag{2.6}$$

where G_{ij} is the geometry-based probability that an emission event in voxel j propogates within the space seen by detector pair i and ε_i are *efficiency factors* for the detectors associated with ray i. Different models/approximations for G_{ij} have been considered (e.g., [91, 74]). Another factorized model (see e.g., [94]) is,

$$P_{ij}(\boldsymbol{\mu}) = \exp(-[\boldsymbol{A}\boldsymbol{\mu}]_i)n_i A_{ij} \tag{2.7}$$

where n_i are the *normalization factors* routinely measured in PET by scanning a uniform test object.

The problem is now to use (2.5) to determine the unknown vector $\boldsymbol{\lambda}$. Here, $\boldsymbol{\mu}$ is a nuissance parameter vector which must be also be determined. A common practice in emission tomography is to pre-determine $\boldsymbol{\mu}$ from a separate transmission scan of the patient, prior to the injection of radio-tracer. One then views $\boldsymbol{\mu}$ as known, whereupon (2.5) reduces to the now classical model

$$\bar{y}_i(\boldsymbol{\lambda}) = [\boldsymbol{P}\boldsymbol{\lambda}]_i + r_i. \tag{2.8}$$

and the task reduces to the determination of $\boldsymbol{\lambda}$. This decoupling leads to relatively simple reconstruction algorithms. Clearly though, a more statistically principled approach would be to aggregate transmission scan and emission scan measurements into a single data set and estimate $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ simultaneously based on (2.3) and (2.5) (e.g., [16, 31]).

2.2.3 Further Tomographic Acquisition Models*

Some more elaborate tomographic acquisition models than those discussed in Section 2.2.1 and Section 2.2.2 have recently been considered. For example, [104] addressed

the following generalization of (2.3),

$$\bar{y}_i(\boldsymbol{\mu}) = \sum_m [b_{im} \exp(-[\boldsymbol{A}_m \boldsymbol{\mu}]_i)] + r_i. \tag{2.9}$$

This equation models the case where the transmission scan measurements in ray i result from separate, overlapping transmission source beams, indexed by m. The summation over m in (2.9) superposes the contributions of all the sources. This model is appropriate for certain Single Photon Emission Computed Tomography (SPECT) system designs.

As another example, [28] and [46] looked at a model for when a transmission and emission scan are both done *after* the patient receives the radio-tracer injection. This results in an emission measurement vector $\boldsymbol{y}^E$ and a transmission measurement vector $\boldsymbol{y}^T$ with Poisson means given by

$$\begin{aligned} \bar{y}_i^E(\boldsymbol{\lambda}, \boldsymbol{\mu}) &= b_i^E \exp(-[\boldsymbol{A}\boldsymbol{\mu}]_i)[\boldsymbol{G}\boldsymbol{\lambda}]_i + r_i^E \\ \bar{y}_i^T(\boldsymbol{\lambda}, \mu) &= b_i^T \exp(-[\boldsymbol{A}\boldsymbol{\mu}]_i) + r_i^T + k_i \exp(-[\boldsymbol{A}\boldsymbol{\mu}]_i)[\boldsymbol{G}\boldsymbol{\lambda}]_i. \end{aligned}$$

Because tracer is present in the patient during the transmission scan, y_i^T gets contribution from both transmission and emission sources. Thus, the above expression for $\bar{y}_i^T$ is a superposition of the form (2.3) and (2.5), with $P_{ij}(\boldsymbol{\mu})$ approximated as in (2.6).

The above show how Poisson models for tomographic systems can become progressively sophisticated. The challenge of designing reconstruction algorithms for such models progresses in a corresponding way.

2.2.4 Radon-Based Reconstruction

When the unknown vector $\boldsymbol{\theta}$ is an image, the Poisson mean vector $\bar{\boldsymbol{y}}_i(\boldsymbol{\theta})$ often has the approximate form

$$\bar{\boldsymbol{y}}(\boldsymbol{\theta}) = f(\boldsymbol{A}\boldsymbol{\theta}) \tag{2.10}$$

where f is an invertible function and, as before, $\boldsymbol{A}$ is a matrix whose ij-th element is the length of intersection of voxel j with ray i. One can see that $\bar{\boldsymbol{y}}(\boldsymbol{\mu})$, as given in, (2.3), has the form (2.10) with $\boldsymbol{\theta} = \boldsymbol{\mu}$, when all the b_i are strictly positive. In a similar manner, one obtains an emission tomography model $\bar{\boldsymbol{y}}(\boldsymbol{\lambda})$ of the form (2.10), with $\boldsymbol{\theta} = \boldsymbol{\lambda}$, by combining (2.7) with (2.5) and pre-computing $\boldsymbol{\mu}$ as described in Section 2.2.2.

One can rewrite (2.10) to obtain

$$\boldsymbol{\theta} = \boldsymbol{A}^{\dagger} f^{-1}(\bar{\boldsymbol{y}}). \tag{2.11}$$

where $\boldsymbol{A}^{\dagger}$ is a pseudo-inverse of $\boldsymbol{A}$. This motivates a simple first order Method of Moments (MOM) type estimator,

$$\hat{\boldsymbol{\theta}}(\boldsymbol{y}) = \boldsymbol{A}^{\dagger} f^{-1}(\boldsymbol{y}). \tag{2.12}$$

Here, $\boldsymbol{y}$ is used as the empirical first order moment approximating $\bar{\boldsymbol{y}}$.

Since $\boldsymbol{A}$ is a matrix of voxel-ray intersections, $\boldsymbol{A\theta}$ performs a discrete approximation of the Radon transform on the unknown image $\boldsymbol{\theta}$. Discrete versions of Radon transform pseudo-inverses can be used to approximate $\boldsymbol{A}^{\dagger}$. Furthermore, efficient algorithms are known by which these pseudo-inverses can be computed. For 2D parallel projection geometries, a popular choice is the Filtered Back Projection (FBP) algorithm, in which $\boldsymbol{A}^{\dagger}$ takes the form,

$$\boldsymbol{A}^{\dagger} = \boldsymbol{A}^{T}\mathcal{F} \tag{2.13}$$

Here, $\mathcal{F}$ is a high-pass filtering operator. A variant of this for fully 3D projections is the 3DRP algorithm [50]. Combining the above with (2.12) gives

$$\hat{\boldsymbol{\theta}}(\boldsymbol{y}) = \boldsymbol{A}^{T}\mathcal{F} f^{-1}(\boldsymbol{y}). \tag{2.14}$$

Reconstruction methods such as these, that are based on the radon transform and its inverse, have been used the most widely by tomographic imaging practitioners for many years. We shall refer to such reconstruction methods as *Radon-based*.

Radon-based methods lead to fair quality reconstructed images of at a minimum of computational expense. Apart from the evaluation of $\boldsymbol{A}^T$, the computations in inversion formulae like (2.14) tend to be minor (see also Section 2.2.6). This quality/computation trade-off was regarded as the optimal one by medical practitioners for many years, and therefore, prior to the late 1990s, Radon-based methods were the most popular and widespread. However, Radon-based methods have at least two shortcomings that limit the quality of the images that they can produce. Firstly, they exploit only the form of the mean $\bar{\boldsymbol{y}}(\boldsymbol{\theta})$ and not on how $\boldsymbol{y}$ is distributed about that mean. Secondly, to obtain the inversion formula in (2.12), it is necessary to approximate $\bar{\boldsymbol{y}}(\boldsymbol{\theta})$ in terms of $\boldsymbol{A}$. This in turn can place restrictions on how one models the acquisition process. For example, in emission tomography, putting $\bar{\boldsymbol{y}}(\boldsymbol{\lambda})$ in the form (2.10) required, as discussed above, that one model $\boldsymbol{P}(\boldsymbol{\mu})$ as in (2.7). In the next section, we discuss penalized likelihood (PL) reconstruction which, while considerably more demanding computationally, suffer from neither of these shortcomings.

2.2.5 Penalized Likelihood Estimation

In past years, much literature (e.g., [39, 41, 40, 55, 23, 33, 22, 4]) has emerged on the use of *penalized likelihood* (PL) estimation algorithms for reconstructing $\boldsymbol{\theta}$ (in particular, the λ or μ components of $\boldsymbol{\theta}$) from $\boldsymbol{y}$. A negative loglikelihood function for Poisson measurement vector $\boldsymbol{y}$ is

$$\begin{aligned} L(\boldsymbol{\theta}) &\stackrel{\triangle}{=} \mathrm{KL}(\boldsymbol{y}, \bar{\boldsymbol{y}}(\boldsymbol{\theta})) \\ &= \sum_{i=1} [\bar{y}_i(\boldsymbol{\theta}) - y_i \log \bar{y}_i(\theta)] - \sum_{i=1} [y_i - y_i \log y_i] \end{aligned} \tag{2.15}$$

By adding to this a penalty function $R(\boldsymbol{\theta})$, one obtain a penalized likelihood cost function

$$\Phi_{\mathrm{PL}}(\boldsymbol{\theta}) = L(\boldsymbol{\theta}) + R(\boldsymbol{\theta}). \tag{2.16}$$

A penalized likelihood estimate $\hat{\boldsymbol{\theta}}$ is obtained according to,

$$\hat{\boldsymbol{\theta}} \in \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}} \{\Phi_{\mathrm{PL}}(\boldsymbol{\theta})\} \tag{2.17}$$

where Θ is an allowable space of parameters. When $R(\boldsymbol{\theta}) \equiv 0$, then (2.17) reduces to the formula for the maximum likelihood estimate (MLE) of $\boldsymbol{\theta}$.

The addition of a penalty improves the conditioning the problem, which in turn reduces the statistical variance of $\hat{\boldsymbol{\theta}}$. Furthermore, it allows one to suppress non-physical behavior that could otherwise manifest in the MLE. For example, in emission tomography, with $\boldsymbol{\theta} = \boldsymbol{\lambda}$, one expects the activity image $\boldsymbol{\lambda}$ to be smooth in most places. Therefore, it is common to choose the penalty $R(\boldsymbol{\lambda})$ as to increase with image roughness. Such a penalty function is often constructed to be of the form,

$$R(\boldsymbol{\lambda}) = \sum_k \omega_k \psi_k([\boldsymbol{C}\boldsymbol{\lambda}]_k). \tag{2.18}$$

Here, the 1D functions $\psi_k(x)$ are so-called potential functions with weights $\omega_k \geq 0$ and the matrix $\boldsymbol{C}$ is such that the components $[\boldsymbol{C}\boldsymbol{\lambda}]_k$ are measurements of differences between neighbhouring voxel intensities in $\boldsymbol{\lambda}$. The $\psi_k(x)$ are usually chosen symmetric, convex, and strictly convex on $|x| < \delta$ for some fixed parameter $0 < \delta \leq \infty$. A difficulty with roughness penalties is that they smooth over desired sharp features in the images like tissue boundaries. In some cases, however, one has access to side information that can be used to discriminate between image regions where smoothness is desirable and undesirable [30, 18]. In such situations, one can set the ω_k weights in (2.18) to low values in regions where sharpness is desired.

It has been found (e.g., [78, 15]) that penalized likelihood estimation algorithms produce images that outperform Radon-based methods at certain medical tasks. A difficulty with penalized likelihood estimation is that the minimization in (2.17) typically has no closed form solution and must be carried out with iterative optimization algorithms. The

iterative nature of these algorithms results in lengthy computation, a drawback that has historically impeded the acceptance of penalized likelihood algorithms in clinical environments. Parallel computing and the development of incremental gradient methods for algorithm acceleration (e.g., [44, 12, 14, 24, 6, 1, 2]) have progressively lessened this problem in recent years. Furthermore, clinicians have lately come to desire better performance from tomographic imaging methods, and therefore seem more prepared to accept the computational demands of penalized likelihood algorithms.

2.2.6 The Role of Forward and Back Projection

In the previous sections, we saw Poisson models $\bar{\boldsymbol{y}}(\boldsymbol{\mu})$ and $\bar{\boldsymbol{y}}(\boldsymbol{\lambda})$ that depend on the unknown images $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ through expressions like $\boldsymbol{A\mu}$ and $\boldsymbol{G\lambda}$. We refer to matrices like $\boldsymbol{A}$ and $\boldsymbol{G}$ – which map image vectors into projection space vectors – as projection matrices. Multiplication with such matrices are called discrete *forward projections* of the images. Conversely, multiplication with their transpose are called discrete *back projection* operations.

Such operations are a fundamental and inevitable in tomographic reconstruction algorithms because they describe the physics relating the unknown parameters and the measurements, which of course must be accounted for in any reasonable reconstruction algorithm. For example, in Radon-based methods of the form (2.14), we see that a backprojection step involving $\boldsymbol{A}^T$ is required. Also, evaluating the derivative of the loglikelihood term $L(\boldsymbol{\theta})$ in the penalized likelihood cost function (2.16) involves one forward and backprojection step. This is because $L(\boldsymbol{\theta})$ depends on $\boldsymbol{\theta}$ through $\bar{\boldsymbol{y}}(\boldsymbol{\theta})$ which in turn depends on $\boldsymbol{\theta}$ through some projection matrix. As an example, in (2.8), $\bar{\boldsymbol{y}}(\boldsymbol{\lambda})$ depends on $\boldsymbol{\lambda}$ through $\boldsymbol{G\lambda}$. Thus, in this case, the gradient of the penalized likelihood takes the form

$$\nabla \Phi_{\text{PL}}(\boldsymbol{\lambda}) = \boldsymbol{G}^T \nabla_{\boldsymbol{G\lambda}} L(\boldsymbol{G\lambda}) + \nabla R(\boldsymbol{\lambda}).$$

Here, we have expressedly denoted the dependence of L on $\boldsymbol{G\lambda}$. One sees that gradient calculations require both a forward projection $\boldsymbol{G\lambda}$, to evaluate $\nabla_{\boldsymbol{G\lambda}} L(\boldsymbol{G\lambda})$ and a back-projection $\boldsymbol{G}^T$. Higher order derivatives require even more forward and back projection matrix manipulations.

Projection matrices tend to be quite large and have historically been the computational bottleneck in tomographic reconstructiong algorithms. Much research effort has been devoted to their fast computation [89, 26, 94, 20, 5]). A common ingredient to the acceleration of these operations is to take advantage of the sparsity of projection matrices. Generally, projection matrix elements A_{ij} or G_{ij} are nonzero only if voxel j lies near ray i. Another ingredient to accelerated projections is to use some simplified approximation to the values of the matrix elements. Such approximations admit faster computation, but compromise the accuracy of the physical model. The impact of this kind of trade-off has been studied, for example, in [103].

2.3 Background on Image Registration

Image registration is a field of image processing dealing with the alignment of geometrically similar images. Formally, one is given a continuous-space source image $f^{\mathrm{src}}(\vec{\boldsymbol{r}})$ and a target image $f^{\mathrm{targ}}(\vec{\boldsymbol{r}})$. The aim is to find a transformation $\mathcal{T}$ such that $\mathcal{T} f^{\mathrm{src}}(\vec{\boldsymbol{r}})$ is similar to $f^{\mathrm{targ}}(\vec{\boldsymbol{r}})$. Naturally, there are restrictions on the form $\mathcal{T}$ can take, so it must be drawn from some agreed upon set S_{trans} of transformation functions. A wide range of registration methods are posed as minimization problems of the generic form,

$$\text{min.} \quad \mathrm{discr}(\mathcal{T} f^{\mathrm{src}}, f^{\mathrm{targ}}) + R(\mathcal{T}) \quad \text{s.t.} \quad \mathcal{T} \in S_{\mathrm{trans}} \tag{2.19}$$

where discr is a measure of the discrepancy between $\mathcal{T} f^{\mathrm{src}}(\vec{\boldsymbol{r}})$ and $f^{\mathrm{targ}}(\vec{\boldsymbol{r}})$ and $R(\mathcal{T})$ is a regularizing penalty function meant to discourage undesired $\mathcal{T}$.

In this dissertation, we shall be mainly concerned with image registration as it applies to intra-modality motion correction. That is, f^{src} and f^{targ} represent images of the same subject acquired by the same physical device, except that the subject has moved between the two acquisitions. The aim is to find a $\mathcal{T}$ that describes the differences between f^{src} and f^{targ} due to this motion. However, it is important to note that the field of image registration goes well beyond this. In the wider spectrum of image registration applications, the images need not have been derived from the same imaging device (inter-modality registration), nor even from the same subject (inter-subject registration). In medical applications, inter-subject registration becomes necessary when doing statistical comparisons of images of the same anatomy across populations of subjects. Inter-modality registration becomes necessary when multiple medical scanning methods are used to obtain different information about the same anatomy. Even if the subject is the same for both, the fact that the subject needs to visit different scanners means the images obtained from these scanners must be aligned for comparison purposes. A more detailed survey of these matters can be found, for example, in [19].

The image registration literature considers a wide variety of possibilities for discr, S_{trans}, and $R(\mathcal{T})$, some more suited to certain registration tasks than others. We overview some of these in the next several subsections. The discussion will be in the context of 3D image registration (thus, $\vec{\boldsymbol{r}} = (x, y, z)$).

2.3.1 Discrepancy Measures

In intra-modality registration (our focus here), the intensity values of both images can be meaningfully compared via the L_2-norm difference,

$$\mathrm{discr}(\mathcal{T} f^{\mathrm{src}}, f^{\mathrm{targ}}) = \frac{1}{2} ||\mathcal{T} f^{\mathrm{src}} - f^{\mathrm{targ}}||_{L_2}^2. \tag{2.20}$$

This criterion has been considered, for example, in [99, 92, 53, 51, 54, 73, 97, 106]. A discrepancy measure that has become popular for inter-modality registration is the mutual information criterion [100, 17, 65, 93, 68, 77],

$$\mathrm{discr}(\mathcal{T} f^{\mathrm{src}}, f^{\mathrm{targ}}) = \mathrm{KL}(\rho(\mathcal{T} f^{\mathrm{src}}, f^{\mathrm{targ}}), \rho(\mathcal{T} f^{\mathrm{src}})\rho(f^{\mathrm{targ}})).$$

where $\rho(f_1, f_2, \dots)$ are the joint intensity distributions of $(f_1, f_2, \dots)$. Other comparison criteria in the image registration literature include those based on landmarks (e.g., [9, 84]). The method known as principal axes transformations (e.g., [3]) uses the image intensities of f^{src} and f^{targ} to define distributions on the spatial coordinates $\vec{\boldsymbol{r}}$ and compares the images in terms of the associated moments. Finally, correlation-based registration compares images in terms of the correlation coefficient of their intensities (see e.g., [38, p. 583]).

2.3.2 Image Transformations

A widely considered class of transformations S_{trans} are pure geometric deformations parametrized by a deformation parameter vector $\boldsymbol{\alpha}$,

$$\mathcal{T}(\boldsymbol{\alpha}) f^{\mathrm{src}}(x, y, z) = f^{\mathrm{src}}(\boldsymbol{d}(x, y, z, \boldsymbol{\alpha})). \tag{2.21}$$

Furthermore, a widely considered form for $\boldsymbol{d}(x, y, z, \boldsymbol{\alpha})$ is where $\boldsymbol{\alpha}$ is partitioned into sub-vectors $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_X, \boldsymbol{\alpha}_Y, \boldsymbol{\alpha}_Z)$ of basis coefficients and

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \overset{\boldsymbol{d}}{\longmapsto} \begin{bmatrix} x + \sum_{k=1}^{K_X} \alpha_{Xk} b_{Xk}(x, y, z) \\ y + \sum_{k=1}^{K_Y} \alpha_{Yk} b_{Yk}(x, y, z) \\ z + \sum_{k=1}^{K_Z} \alpha_{Zk} b_{Zk}(x, y, z) \end{bmatrix}. \tag{2.22}$$

Here $\{b_{Ck}(x, y, z)\}_{C \in \{X,Y,Z\}}$ are deformation basis functions for the various Cartesian directions. When $\{b_{Ck}(x, y, z)\}_{k=1}^{K_C} = \{x, y, z, 1\}$ for all $C = \{X, Y, Z\}$, we obtain the family of 12 degree of freedom affine[1] deformations. Usually, one is interested in affine

[1]The use of the term *affine* here refers to the form of $\boldsymbol{d}$ as a function of (x, y, z) and *not* as a function of $\boldsymbol{\alpha}$. Clearly, all $\boldsymbol{d}$ in the more general family (2.22) are affine with respect to $\boldsymbol{\alpha}$. This terminology is simply a convention of the image

deformations restricted to the set

$$S_{\text{trans}} = \left\{ \boldsymbol{\alpha} \,:\, \det \nabla_{\vec{\boldsymbol{r}}}^{T} \boldsymbol{d}(\vec{\boldsymbol{r}}, \boldsymbol{\alpha}) > 0 \right\}. \tag{2.23}$$

Within the family of affine deformations, there is the family of 6 degree of freedom *rigid* deformations, wherein f^{src} can only rotate/translate like a solid object. The family of rigid deformations can be obtained from the affine family by restricting S_{trans}, so that $\boldsymbol{\alpha}$ satisfies,

$$S_{\text{trans}} = \left\{ \boldsymbol{\alpha} \,:\, \nabla_{\vec{\boldsymbol{r}}}^{T} \boldsymbol{d}(\vec{\boldsymbol{r}}, \boldsymbol{\alpha}) \nabla_{\vec{\boldsymbol{r}}} \boldsymbol{d}(\vec{\boldsymbol{r}}, \boldsymbol{\alpha}) = \boldsymbol{I} \right\}. \tag{2.24}$$

Rigid deformations arise in motion correction applications when the anatomy in question is well-modeled as a rigid body. This includes head motion correction (e.g., [45]). Both rigid and affine deformation models have been applied to cardiac motion (e.g., [51, 61]).

Situations are encountered when higher degree of freedom deformations are desired. A typical case is in modeling the motion of non-rigid, elastic anatomy, such as the thorax and lungs. Another case is in inter-subject registration. For such applications, a popular choice of deformation basis functions [52, 68, 86, 81, 59, 73] are those based on B-splines. Assume, for convenience, that f^{src} and f^{targ} are supported on continuous space region of integer dimensions $N_X \times N_Y \times N_Z$ and that these dimensions are evenly divisible by $\{\Delta_X, \Delta_Y, \Delta_Z\}$. Then the standard construction is, for each $C \in \{X, Y, Z\}$, to cover this support region with basis functions,

$$b_{Ck}(x, y, z) = \beta_q \left(\frac{x}{\Delta_X} - m_k \right) \beta_q \left(\frac{y}{\Delta_Y} - n_k \right) \beta_q \left(\frac{z}{\Delta_Z} - p_k \right) \tag{2.25}$$

where β_q is the 1D B-spline of order q and (m_k, n_k, p_k), $k = 1, 2, \ldots$ are the node coordinates of a *control point* grid of dimension $(N_X/\Delta_X) \times (N_Y/\Delta_Y) \times (N_Z/\Delta_Z)$ nodes.

---

registration literature.

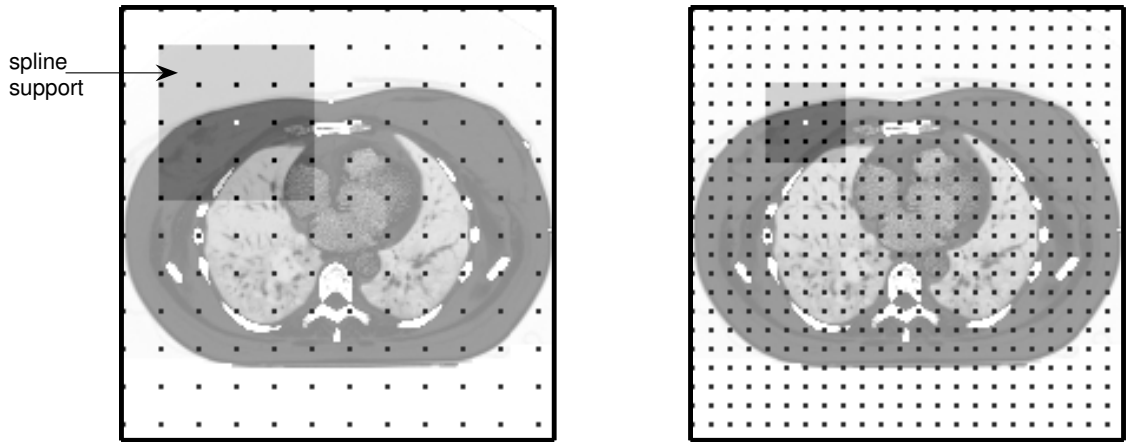

Figure 2.2: Depicts a 2D control point grid, superimposed on a thorax phantom, at two levels of fineness. The region of influence of a cubic B-spline basis functions is also shown for each case.

Most often, cubic B-splines are used

$$\beta_3(t) = \begin{cases} \frac{|t|^3}{2} - |t|^2 + \frac{2}{3}, & 0 \leq |t| \leq 1 \\ \frac{-(|t|-2)^3}{6}, & 1 \leq |t| \leq 2 \\ 0, & |t| \geq 2 \end{cases} \tag{2.26}$$

A principal advantage of B-spline models is related to the localized support of β_q. This allows spatial deformation to be controlled locally via the corresponding α_{Ck} in that region. One can also tune the fineness or coarseness of the deformation model by adjusting the node spacings $\{\Delta_C\}_{C\in\{X,Y,Z\}}$. Figure 2.2 depicts a 2D control point grid at two levels of fineness, along with the region of influence of a cubic B-spline basis functions in both cases.

A difficulty with B-spline deformations is that they can exhibit non-physical traits such as uninvertibility (with respect to $\vec{\boldsymbol{r}}$) or an absence of topology-preserving behavior. One can combat this by restricting $\boldsymbol{\alpha}$ to certain constrained regions, as in [73, 49]. In [71], an

alternative to B-spline bases is considered, that is invertible, but shares, to a degree, the localization properties of B-spline models. Another difficulty with B-splines arises when modeling the deformation of objects containing some rigid, but mostly elastic substances (e.g., the bone and soft tissue present in the thorax). Transformations based on B-splines may induce rigid media to deform elastically, contrary to their physical nature. Efforts to combat this include post-filtering steps [90] and the design of regularizing terms $R(\mathcal{T})$, as in [85], to encourage locally rigid behavior, where appropriate. In [67], an alternative to B-splines, involving weighted combinations of rigid transformations, was proposed.

2.3.3 Regularization in Image Registration

Here we discuss briefly, different choices of the regularizing term $R(\mathcal{T})$ in (2.19). For the parametric $\mathcal{T}$ given by (2.21) and (2.25), we have already mentioned the use of penalties in [85] to encourage different degrees of rigidity at different spatial locations. In [106], neighbourhood roughness penalties, similar to those discussed for images in Section 2.2.5, were applied to the α_{Ck} in the control point grid. The idea here was that smooth transitions between neighbouring α_{Ck} would induce smooth deformations. In [83], the performance of a penalty on bending energy, as quantified by the sum of the entries $(\nabla^T_{\vec{\boldsymbol{r}}}\boldsymbol{d}(\vec{\boldsymbol{r}},\boldsymbol{\alpha}))(\nabla_{\vec{\boldsymbol{r}}}\boldsymbol{d}(\vec{\boldsymbol{r}},\boldsymbol{\alpha}))$, is compared to a compressibility penalty, that penalizes deviations of $\det \nabla^T_{\vec{\boldsymbol{r}}}\boldsymbol{d}(\vec{\boldsymbol{r}},\boldsymbol{\alpha})$ from unity. The use of penalties to incorporate side information is seen, for example, in [54]. There, a weighted quadratic penalty is constructed based on landmark side information that penalizes deviations of deformed points from the landmarks.

A variety of regularizers also arise in non-parametric registration problems based on the parameter-free analogue of(2.21),

$$\mathcal{T} f^{\text{src}}(x,y,z) = f^{\text{src}}(\boldsymbol{d}(x,y,z)).$$

Some of these regularizers are motivated by physical equations for the behavior of matter. Examples include *elastic registration* [11] and *fluid registration* [70, p. 121]. Others are constructed to encourage spatial smoothness of $\boldsymbol{d}$ based on its derivatives as in *diffusion registration* [34] and *curvature registration* [35].

2.4 Background on Majorize-Minimize (MM) Algorithms

One can see, for example from (2.17) and (2.19), that tomographic imaging and motion correction applications often involve the minimization of cost functions. One approach to such minimizations is the Majorize-Minimize (MM) optimization technique.[2] The MM technique will be the focus of Chapter 6. To aid subsequent discussion, however, we give some preliminary background here. Given a minimization problem

$$\text{min.} \quad \Phi(\boldsymbol{\theta}) \quad \text{s.t.} \quad \boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p, \tag{2.27}$$

an MM algorithm is one that reduces Φ monotonically by minimizing a succession of approximations to Φ, each of which majorizes Φ in a certain sense. An MM algorithm uses what we call a *majorant generator* $\phi(\cdot;\cdot)$ to associate a given *expansion point* $\boldsymbol{\theta}^i$ with what we call a *tangent majorant* $\phi(\cdot;\boldsymbol{\theta}^i)$. In the simplest case (illustrated for a 1D cost function in Figure 2.3), a tangent majorant satisfies $\Phi(\boldsymbol{\theta}) \leq \phi(\boldsymbol{\theta};\boldsymbol{\theta}^i)$ for all $\boldsymbol{\theta} \in \Theta$ and $\Phi(\boldsymbol{\theta}^i) = \phi(\boldsymbol{\theta}^i;\boldsymbol{\theta}^i)$. That is, $\phi(\cdot;\boldsymbol{\theta}^i)$ majorizes Φ with equality at $\boldsymbol{\theta}^i$. Because of this majorization condition, any $\boldsymbol{\theta} \in \Theta$ that reduces $\phi(\cdot;\boldsymbol{\theta}^i)$ from its value at $\boldsymbol{\theta}^i$, likewise reduces Φ,

$$\phi(\boldsymbol{\theta};\boldsymbol{\theta}^i) < \phi(\boldsymbol{\theta}^i;\boldsymbol{\theta}^i) \implies \Phi(\boldsymbol{\theta}) < \Phi(\boldsymbol{\theta}^i)$$

In particular, the constrained minimizer $\boldsymbol{\theta}^{i+1} \in \Theta$ of $\phi(\cdot;\boldsymbol{\theta}^i)$ satisfies $\Phi(\boldsymbol{\theta}^{i+1}) < \Phi(\boldsymbol{\theta}^i)$ whenever strict reduction of $\phi(\cdot;\boldsymbol{\theta}^i)$ is possible. Repeating these steps iteratively, one

[2] The technique has gone by various other names as well, such as iterative majorization or optimization transfer.

obtains a sequence of feasible vectors $\{\boldsymbol{\theta}^i\}$ such that $\{\Phi(\boldsymbol{\theta}^i)\}$ is monotone non-increasing.

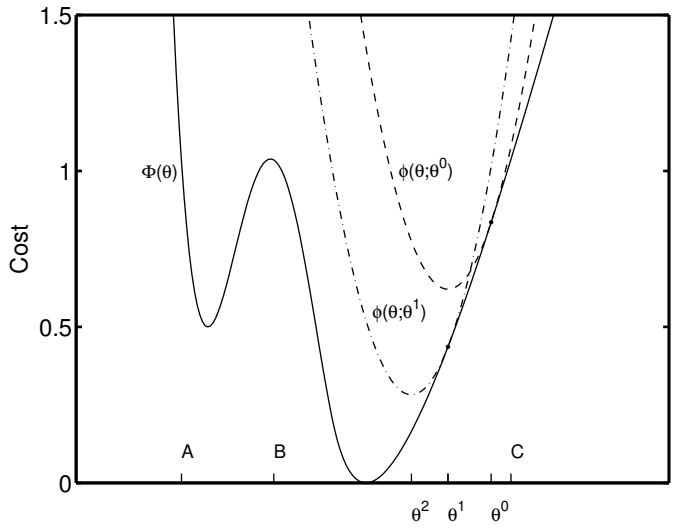

Figure 2.3: One-dimensional illustration of an MM algorithm.

More elaborate forms of MM have been considered [32, 33] that allow an iteration-dependent sequence $\{\phi^i(\cdot;\cdot)\}$ of majorant generators to be used, rather than just a single $\phi(\cdot;\cdot)$. This generalization allows one to choose the form of the majorant generator at a given iteration based on the observed progress of the algorithm. In addition, one can allow the tangent majorants $\{\phi^i(\cdot;\boldsymbol{\theta}^i)\}$ to depend on an i-dependent subset of the components of $\boldsymbol{\theta}$. Thus, one can take iterative steps that, similar to coordinate descent, reduce $\Phi(\boldsymbol{\theta})$ as a function of subsets of the optimization variables. This technique, which we call *block alternation*, can facilitate MM algorithm design for several reasons. Firstly, it makes it easier to satisfy the majorization requirement, because $\Phi(\boldsymbol{\theta}) \leq \phi^i(\boldsymbol{\theta};\boldsymbol{\theta}^i)$ now need only be satisfied for certain components of $\boldsymbol{\theta}$. Secondly, since the majorization requirement is weaker, it also becomes easier for $\phi^i(\cdot;\boldsymbol{\theta}^i)$ to approximate Φ with higher accuracy, which can lead to faster convergence [32]. Thirdly, each resulting $\phi^i(\cdot;\boldsymbol{\theta}^i)$ can be easier to minimize since it is a function of fewer than p variables.

The subject of MM has appeared in a range of statistics literature over the years (e.g., [42, 43, 56, 58]). A prominent example is the Expectation Maximization (EM) methodol-

ogy (commonly attributed to [25]) which is an application of MM to maximum likelihood estimation. In the field of statistical tomographic imaging, maximum likelihood estimators, and EM algorithms for implementing them, achieved early popularity in [88] and [57]. Since then, MM has become an algorithm design technique of increasing prevalence in this field, as is apparent in such works as [21, 22, 23, 104, 107]. The popularity of MM has also reached the commercial medical imaging world. In particular, several manufacturers now package their emission tomography scanners with software for an incremental gradient version (due to [44]) of an MM algorithm.

An MM algorithm is derived by choosing majorant generators of a particular form, and the form selected is based on the algorithm designer's insights into the shape of the cost function in question. One design goal is to choose majorant generators $\{\phi^i(\cdot;\cdot)\}$ so that the corresponding tangent majorants $\{\phi^i(\cdot;\boldsymbol{\theta}^i)\}$ approximate Φ as accurately as possible over as large a region of Θ as possible, all the while respecting the majorization requirement. One expects greater approximation accuracy to result in more rapid descent of $\{\Phi(\boldsymbol{\theta}^i)\}$. At the same time, one tries to choose the majorant generators so that the resulting tangent majorants $\{\phi^i(\cdot;\boldsymbol{\theta}^i)\}$ can be constructed and minimized in a computationally efficient manner. Typically, therefore, one generates tangent majorants that are convex (although we will not make this a global assumption in our analysis in Chapter 6) as in Figure 2.3. Naturally, these design aims can conflict, and this has led to abundant literature that examines alternative designs for particular applications (e.g., [27, 28, 32, 33]).

Since the majorant generators, and hence the descent mechanism of the algorithm, are custom designed to suit the particular cost function at hand, MM algorithms have the potential to outperform more generic algorithms that do not take the specifics of Φ into account. The latter would include textbook-variety derivative based methods (e.g., gradient projection and conditional gradient). These considerations are particularly important if Φ

is of a general type that must be minimized routinely. If a cost function of a specific form is minimized on a repeat basis then, logically, algorithms like MM that are custom-designed to suit this form can be a worthwhile investment. This notion seems to account for the popularity of MM in statistical tomographic imaging. There, one routinely minimizes instances of loglikelihood or other cost functions of a common form, but corresponding to different sets of measured data.

CHAPTER 3

A Matrix-Vector Framework for Interpolation-Based Image Deformations

In this chapter, we develop a discrete domain description of the transformation $\mathcal{T}(\boldsymbol{\alpha})$ defined by (2.21) and (2.22) when f^{src} has the parametric form

$$f^{\text{src}}(x,y,z) = \sum_{k=1}^{|\mathcal{G}|} u_k w_k(x,y,z). \tag{3.1}$$

Here, $\mathcal{G}$ is the following set of integer grid coordinates

$$\mathcal{G} \stackrel{\triangle}{=} \left\{(x_k,y_k,z_k) \in \mathbb{Z}^3 \,:\, \mathbf{0} \le (x_k,y_k,z_k) \le (N_X,N_Y,N_Z)\right\},$$

the u_k are components of a parameter vector $\boldsymbol{u} \in \mathbb{R}^{|\mathcal{G}|}$, and

$$w_k(x,y,z) = w\left(x-x_k\right) w\left(y-y_k\right) w\left(z-z_k\right), \qquad (x_k,y_k,z_k) \in \mathcal{G}. \tag{3.2}$$

are interpolating basis functions formed from tensor product of the 1D interpolator w. The integer coordinates $(x_k,y_k,z_k) \in \mathcal{G}$ range with k over $\mathcal{G}$. This framework will be used throughout this dissertation.

3.1 Deformation Space Notation

We first consider the array obtained from the deformed sample points

$$\boldsymbol{d}_j(\boldsymbol{\alpha}) \stackrel{\triangle}{=} \boldsymbol{d}(x_j,y_j,z_j,\boldsymbol{\alpha}), \qquad (x_j,y_j,z_j) \in \mathcal{G}. \tag{3.3}$$

Define $\boldsymbol{\nu}_X$, $\boldsymbol{\nu}_Y$, and $\boldsymbol{\nu}_Z$, all in $\mathbb{R}^{|\mathcal{G}|}$, as the vectors whose j-th components are x_j, y_j, and z_j respectively, and

$$\boldsymbol{\nu} \stackrel{\triangle}{=} (\boldsymbol{\nu}_X, \boldsymbol{\nu}_Y, \boldsymbol{\nu}_Z).$$

Also, define the matrices $\boldsymbol{B}_X$, $\boldsymbol{B}_Y$, and $\boldsymbol{B}_Z$ to have entries

$$[\boldsymbol{B}_C]_{jk} \stackrel{\triangle}{=} b_{Ck}[x_j, y_j, z_j], \quad C \in \{X, Y, Z\}.$$

Then (3.3) can be re-expressed

$$\boldsymbol{d}_j(\boldsymbol{\alpha}) = (d_{jX}(\boldsymbol{\alpha}_X), d_{jY}(\boldsymbol{\alpha}_Y), d_{jZ}(\boldsymbol{\alpha}_Z)) \tag{3.4}$$

$$\stackrel{\triangle}{=} ([\boldsymbol{\nu}_X + \boldsymbol{B}_X\boldsymbol{\alpha}_X]_j, [\boldsymbol{\nu}_Y + \boldsymbol{B}_Y\boldsymbol{\alpha}_Y]_j, [\boldsymbol{\nu}_Z + \boldsymbol{B}_Z\boldsymbol{\alpha}_Z]_j). \tag{3.5}$$

The vector quantity $\boldsymbol{B}_C\boldsymbol{\alpha}_C$ is the discrete *deformation map*, for the direction of deformation $C \in \{X, Y, Z\}$.

We shall also define

$$\boldsymbol{B} \stackrel{\triangle}{=} \operatorname{diag}\{\boldsymbol{B}_X, \boldsymbol{B}_Y, \boldsymbol{B}_Z\}.$$

and refer to $\boldsymbol{B}\boldsymbol{\alpha}$ as the overall *deformation map vector*. Note that

$$\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{\alpha} = (\boldsymbol{\nu}_X + \boldsymbol{B}_X\boldsymbol{\alpha}_X, \boldsymbol{\nu}_Y + \boldsymbol{B}_Y\boldsymbol{\alpha}_Y, \boldsymbol{\nu}_Z + \boldsymbol{B}_Z\boldsymbol{\alpha}_Z) \tag{3.6}$$

in light of our vector concatenation convention (see Section 2.1, Item 3).

3.2 Intensity Transformation Matrices

Let us define the operator $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})$ as the one that maps $\boldsymbol{f}^{\text{src}}$, the vector whose j-th component is $f_j^{\text{src}} \stackrel{\triangle}{=} f^{\text{src}}(x_j, y_j, z_j)$ to the vector whose j-th component is $\mathcal{T}(\boldsymbol{\alpha})f^{\text{src}}(x_j, y_j, z_j)$.

Then combining (3.1) with (2.21)

$$f_j^{\text{src}} = \sum_k w_k(x_j, y_j, z_j) u_k \tag{3.7}$$

$$\begin{aligned}[\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})\boldsymbol{f}^{\text{src}}]_j &= f^{\text{src}}(\boldsymbol{d}(x_j, y_j, z_j, \boldsymbol{\alpha})) \\ &= \sum_k w_k(\boldsymbol{d}_j(\boldsymbol{\alpha})) u_k. \end{aligned} \tag{3.8}$$

Now, we define the matrix $\boldsymbol{W}(\boldsymbol{\alpha})$ to have entries

$$[\boldsymbol{W}(\boldsymbol{\alpha})]_{jk} \stackrel{\triangle}{=} w_k(\boldsymbol{d}_j(\boldsymbol{\alpha})). \tag{3.9}$$

From (3.5), we have that $\boldsymbol{d}_j(\boldsymbol{0}) = (x_j, y_j, z_j)$ and therefore

$$[\boldsymbol{W}(\boldsymbol{0})]_{jk} = w_k(x_j, y_j, z_j). \tag{3.10}$$

Thus, (3.7) and (3.8) can be re-expressed in the matrix-vector form

$$\boldsymbol{f}^{\text{src}} = \boldsymbol{W}(\boldsymbol{0})\boldsymbol{u} \tag{3.11}$$

$$\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})\boldsymbol{f}^{\text{src}} = \boldsymbol{W}(\boldsymbol{\alpha})\boldsymbol{u}, \tag{3.12}$$

Finally, we assume that the interpolators w_k are chosen so that in (3.7), the u_k are uniquely defined by the samples f_j^{src}, or equivalently that $\boldsymbol{W}(\boldsymbol{0})$ is invertible in (3.11). Combining (3.11) and (3.12) we obtain,

$$\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})\boldsymbol{f}^{\text{src}} = \boldsymbol{W}(\boldsymbol{\alpha})(\boldsymbol{W}(\boldsymbol{0}))^{-1}\boldsymbol{f}^{\text{src}}, \tag{3.13}$$

and since this holds for any $\boldsymbol{f}^{\text{src}}$,

$$\boxed{\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha}) = \boldsymbol{W}(\boldsymbol{\alpha})\ (\boldsymbol{W}(\boldsymbol{0}))^{-1},} \tag{3.14}$$

which is the explicit representation of $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})$ as a matrix.

3.3 Discrete Formulations of Registration Problems

In practical image registration, it is necessary to operate on discrete representations of the source and target images f^{src} and f^{targ}. Accordingly, one must also pose discrete versions of the discrepancy measures discussed in Section 2.3.1. In this dissertation, we will be interested in the comparison of $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})\boldsymbol{f}^{\mathrm{src}}$ to $\boldsymbol{f}^{\mathrm{targ}}$ (defined analogously to $\boldsymbol{f}^{\mathrm{src}}$) via the Sum of Squared Differences (SSD) criterion

$$\mathrm{discr}_{\mathrm{SSD}}(\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})\boldsymbol{f}^{\mathrm{src}}, \boldsymbol{f}^{\mathrm{targ}}) \stackrel{\triangle}{=} \frac{1}{2}||\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})\boldsymbol{f}^{\mathrm{src}} - \boldsymbol{f}^{\mathrm{targ}}||_{\ell_2}^2 \tag{3.15}$$

which is the widely considered discrete version (e.g., [99, 92, 53, 51, 54, 73, 97, 106]) of (2.20). For convenience, we shall make the abbreviated definition

$$\Phi_{\mathrm{SSD}}(\boldsymbol{\alpha}) \stackrel{\triangle}{=} \mathrm{discr}_{\mathrm{SSD}}(\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})\boldsymbol{f}^{\mathrm{src}}, \boldsymbol{f}^{\mathrm{targ}}).$$

When the w in (3.2) are B-splines, it is better, for the purpose of minimizing $\Phi_{\mathrm{SSD}}(\boldsymbol{\alpha})$ (or similar cost functions) to incorporate (3.12) and re-express (3.15) as follows

$$\Phi_{\mathrm{SSD}}(\boldsymbol{\alpha}) \stackrel{\triangle}{=} \frac{1}{2}||\boldsymbol{W}(\boldsymbol{\alpha})\boldsymbol{u} - \boldsymbol{f}^{\mathrm{targ}}||_{\ell_2}^2. \tag{3.16}$$

This is because $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})$, as given by (3.14) is much less sparse[1] a matrix than $\boldsymbol{W}(\boldsymbol{\alpha})$. Iterative minimization of the RHS of (3.16) tends to be much more efficient, in terms of gradient evaluations and so forth, than the RHS of (3.15). For these same reasons, we will also have occasion to re-express other cost functions $\Phi(\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})\boldsymbol{f}^{\mathrm{src}})$ as $\Phi(\boldsymbol{W}(\boldsymbol{\alpha})\boldsymbol{u})$, thereby bringing the problem into the domain of u_k coefficients.

---

[1] In fact, if one considers (3.11) for the case when the w are B-splines, it becomes clear that $(\boldsymbol{W}(\boldsymbol{0}))^{-1}$, the mapping from the vector of B-spline coefficients $\boldsymbol{u}$ to the vector image samples $\boldsymbol{f}^{\mathrm{src}}$, is simply interpolation pre-filter and $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})$ is the operator that re-interpolates the samples f_j^{src} via the corresponding cardinal splines. (cf. [98, pp. 25-26]). Naturally, direct interpolation via cardinal splines is a much less sparse operation than when done in the B-spline domain.

CHAPTER 4

A Comparison of Motion-Corrected PET Reconstruction Methods

4.1 Introduction

In this chapter, we present the first branch of work of which this dissertation is composed. Namely, we compare different methods for carrying out respiratory motion-corrected PET reconstruction of the thorax. In making this comparison, the aim is to assess which of various methods gives best results in quantifying the amount of radio-tracer in hot lesions at the boundary of the lung (of the kind seen in FDG oncology studies).

In past years, the conventional practice in Positron Emission Tomography (PET) has been to reconstruct images while ignoring the effects of anatomical motion in the patient. When motion is ignored, a PET scanner acts something like a conventional optical camera with the shutter open: one obtains the superposition of images of the object as it appears in various phases of motion. Thus, the image exhibits a degree of blur that is related to the motion magnitude. Until recently, the drawbacks of ignoring object motion has been ignored due to limitations in the spatial resolution of early PET scanners. The magnitude of anatomical motion was not always substantial enough, in comparison to the scanner resolution, for motion correction to promise any benefit. However, with the improvement of scanner resolution over time, motion-related blur is becoming a limiting factor in the resolution achievable in PET reconstruction.

This problem has been compounded by changing trends in the application of PET. In early years, PET was applied mainly in the research of the brain, where the relatively small movements of the head made motion effects still easier to dismiss. Conversely, recent clinical PET practice has emphasized cancer treatment, and therefore thorax scans have become more prevalent. Thus, not only has scanner resolution increased, but interest has shifted over the years to the scan of anatomy where motion is much larger. As a result of these factors, the recent tomographic imaging literature has seen much interest (e.g., [51, 62, 37, 66, 95, 80, 61, 96, 79]) in motion correction techniques.

A prominent class of motion correction techniques that one sees in the literature revolve around applying motion-correcting pre-transformations to the photon detection information, e.g., [62, 95, 80, 61]. These techniques have been applied mainly to brain and cardiac imaging. They assume an affine model for the anatomical motion, and in some cases also on the ability to measure the motion directly. Both assumptions are highly non-ideal when dealing with motion of the lungs, as is our focus here.

Another prominently considered class that one sees (e.g., [51, 97, 96, 79]), and one that does not have the above limitations, is based on frame-wise reconstruction from gated projection data. Gated data consists of a separate measured projection vectors $\{\boldsymbol{y}_t\}_{t=0}^{N_{\text{gates}}-1}$ each constituting a scan of the object in one of N_{gates} phases of its motion. Frame-wise methods proceed by first reconstructing an image from each projection gate (or frame) $\boldsymbol{y}_t$, then aligning these images by image registration methods, and finally fusing them together in some sort of consolidation step (e.g., by averaging them). In these types of approaches, the registration step is done post-reconstruction and hence, typically, does not make full use of the projection measurement statistics. Moreover, each image registration step occurs between very noisy images reconstructed from low-count, single-gate data, and can be expected to result in poor alignment.

This motivated us in earlier work [47] to pose a parametric Poisson model for the gated measurements involving a single unknown activity image and a set of deformation variables describing how that image deforms from gate to gate due to patient motion. By maximizing a loglikelihood for this model, a method we call Joint Estimation by Deformation Modeling (JEDM), one determines both image and deformation parameter estimates jointly from the full set of projection measurements. We previously compared JEDM to frame-wise approaches and to ungated reconstruction for a highly simplified simulated thorax scan involving a single-slice image, a 2-gate acquisition, and a 1-parameter deformation model. Preliminary results showed JEDM to outperform the other methods in terms of both lung lesion tracer uptake recovery and motion estimation.

In this chapter, we continue this work, but look at much more realistic simulations. Reconstruction is done with a 3D B-spline motion model based on 2300 parameters per gate. Moreover, the simulated motion and activity distribution are derived from CT scans of actual thorax anatomy. Thirdly, whereas before we tested only pure maximum likelihood estimation, here we add a roughness penalty term to accomplish regularization. In designing the penalty, we discuss how side information can be used – assumed to be available from a breath-hold CT scan – to avoid smoothing over hot lesions in a region of interest (see Section 4.4.5).

This chapter is organized as follows. In Section 4.2, we describe the Poisson model for gated PET data on which JEDM is based. The various reconstruction methods to be compared are outlined in Section 4.3. Some details on how the methods will be implemented are given in Section 4.4. Finally, in Section 4.5, we present experiments comparing the methods in terms of their lesion quantification performance. Experiments are also presented that test our lesion preserving penalty strategy.

4.2 Parametric Statistical Model for Gated PET Data

In acquiring gated data, one obtains a set of measured projection vectors $\{\boldsymbol{y}_t\}_{t=0}^{N_{\text{gates}}-1}$, each constituting a scan of the object in one of N_{gates} phases (which we shall interchangeably refer to as gates or frames) of its motion. The total measured projection vector can be represented as the concatenation $\boldsymbol{y} = (\boldsymbol{y}_0, \ldots, \boldsymbol{y}_{N_{\text{gates}}-1})$. The natural extension of the classical statistical model (2.8) is to model the mean of each $\boldsymbol{y}_t$ as

$$\bar{\boldsymbol{y}}_t(\boldsymbol{\lambda}_t) = \tau_t(\boldsymbol{P}\boldsymbol{\lambda}_t + \boldsymbol{r}_t) \qquad 0 \le t \le N_{\text{gates}} - 1 \tag{4.1}$$

where $\boldsymbol{\lambda}_t$ denotes the activity sample vector of the object, as it appears in gate t and τ_t is the gate duration.

A limitation of this model is that it does not exploit the fact that the $\{\boldsymbol{\lambda}_t\}$ are related, i.e., that they derive from motion-deformed versions of the same object. Our approach, based on work begun in [47], is to relate the $\{\boldsymbol{\lambda}_t\}$ using the discrete image transformation model described in Chapter 3. Accordingly, we introduce an unknown deformation parameter vector $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_{N_{\text{gates}}})$ and model the relationship between the $\{\boldsymbol{\lambda}_t\}$ as follows,

$$\boldsymbol{\lambda}_0 \stackrel{\triangle}{=} \boldsymbol{\lambda} \tag{4.2}$$

$$\boldsymbol{\lambda}_t = \boldsymbol{\mathcal{T}}(\boldsymbol{\alpha}_t)\boldsymbol{\lambda}, \qquad 1 \le t \le N_{\text{gates}} - 1. \tag{4.3}$$

Combining this with (4.1) leads to

$$\bar{\boldsymbol{y}}_0(\boldsymbol{\lambda}) = \tau_0(\boldsymbol{P}\boldsymbol{\lambda} + \boldsymbol{r}_0) \tag{4.4}$$

$$\bar{\boldsymbol{y}}_t(\boldsymbol{\lambda}, \boldsymbol{\alpha}_t) = \tau_t(\boldsymbol{P}\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha}_t)\boldsymbol{\lambda} + \boldsymbol{r}_t), \quad 1 \le t \le N_{\text{gates}} - 1. \tag{4.5}$$

Thus, the mean of the projection data in gate 0 are now based on projections of $\boldsymbol{\lambda}$, the activity image samples in that gate. In each subsequent gate, however, the mean is based on projections of $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha}_t)\boldsymbol{\lambda}$, a transformation of $\boldsymbol{\lambda}$ that accounts for motion. Note that

the number of unknowns in $(\boldsymbol{\lambda}, \boldsymbol{\alpha})$ would normally be much smaller than the number in $\{\boldsymbol{\lambda}_t\}$. The deformation parameter vector $\boldsymbol{\alpha}$ would consist of a relatively small number of components, reflecting the number of degrees of freedom in the anatomical motion.

Taking into account (3.11) and (3.12) with $\boldsymbol{\lambda} = \boldsymbol{f}^{\mathrm{src}}$, one can see that alternative parametrization of (4.4) and (4.5) is

$$\bar{\boldsymbol{y}}_0(\boldsymbol{u}) = \tau_0(\boldsymbol{P}\boldsymbol{W}(\boldsymbol{0})\boldsymbol{u} + \boldsymbol{r}_0) \tag{4.6}$$

$$\bar{\boldsymbol{y}}_t(\boldsymbol{u}, \boldsymbol{\alpha}_t) = \tau_t(\boldsymbol{P}\boldsymbol{W}(\boldsymbol{\alpha}_t)\boldsymbol{u} + \boldsymbol{r}_t), \quad 1 \le t \le N_{\mathrm{gates}} - 1. \tag{4.7}$$

where

$$\boldsymbol{u} = (\boldsymbol{W}(\boldsymbol{0}))^{-1}\boldsymbol{\lambda}. \tag{4.8}$$

The gate-wise models (4.6) and (4.7), can be expressed more compactly as

$$\bar{\boldsymbol{y}}(\boldsymbol{u}, \boldsymbol{\alpha}) = \boldsymbol{\mathcal{P}}(\boldsymbol{\alpha})\boldsymbol{u} + \boldsymbol{r} \tag{4.9}$$

where (denoting here the Kronecker product by $\otimes$ and the $\mathbb{R}^d$ identity by $\boldsymbol{I}_d$),

$$\begin{aligned} \boldsymbol{\mathcal{P}}(\boldsymbol{\alpha}) &\stackrel{\triangle}{=} \boldsymbol{I}_{N_{\mathrm{gates}}} \otimes (\tau_0\boldsymbol{W}(\boldsymbol{0}), \tau_1\boldsymbol{W}(\boldsymbol{\alpha}_1), \ldots, \tau_{N_{\mathrm{gates}}-1}\boldsymbol{W}(\boldsymbol{\alpha}_{N_{\mathrm{gates}}-1})) \\ \boldsymbol{r} &\stackrel{\triangle}{=} (\tau_0\boldsymbol{r}_0, \ldots, \tau_{N_{\mathrm{gates}}-1}\boldsymbol{r}_{N_{\mathrm{gates}}-1}). \end{aligned} \tag{4.10}$$

When the matrices $\{\boldsymbol{W}(\boldsymbol{\alpha}_t)\}$ have non-negative entries, one can interpret $\boldsymbol{\mathcal{P}}(\boldsymbol{\alpha})$ as a motion-corrected projection matrix. This is the case when, in (3.2), the 1D interpolators w are non-negative.

4.3 Proposed Reconstruction Methods

In this chapter, we shall compare three methods, described below, for reconstructing $\boldsymbol{\lambda}$. Two of these (JEDM and FWPR-PA) were considered in our previous work [47].

1. Joint Estimation with Deformation Modeling (JEDM). In the JEDM method, we consider the joint penalized loglikelihood function (see also Section 2.2.5) based on (4.4) and (4.5)

$$\Phi_{\text{JEDM}}(\boldsymbol{\lambda}, \boldsymbol{\alpha}) = \text{KL}(\boldsymbol{y}_0, \bar{\boldsymbol{y}}_0(\boldsymbol{\lambda})) + \sum_{t=1}^{N_{\text{gates}}-1} \text{KL}(\boldsymbol{y}_t, \bar{\boldsymbol{y}}_t(\boldsymbol{\lambda}, \boldsymbol{\alpha}_t)) + \beta R_{\text{act}}(\boldsymbol{\lambda}). \tag{4.11}$$

Here

$$R_{\text{act}}(\boldsymbol{\lambda}) = \sum_k \omega_k \psi([\boldsymbol{C}\boldsymbol{\lambda}]_k). \tag{4.12}$$

is an activity roughness penalty function of the form (2.18), and $\beta \geq 0$ is a regularization parameter controlling the influence of R_{act}. We then reconstruct an image $\hat{\boldsymbol{\lambda}}$ according to

$$\hat{\boldsymbol{\lambda}} = \underset{\boldsymbol{\lambda}}{\text{argmin}} \left\{ \min_{\boldsymbol{\alpha}} \Phi_{\text{JEDM}}(\hat{\boldsymbol{\lambda}}, \boldsymbol{\alpha}) \right\} \tag{4.13}$$

This is illustrated in Figure 4.1. We introduced this approach in [47], but made very preliminary tests there and only for the unregularized case (i.e., $\beta = 0$), which corresponds to pure maximum likelihood estimation.

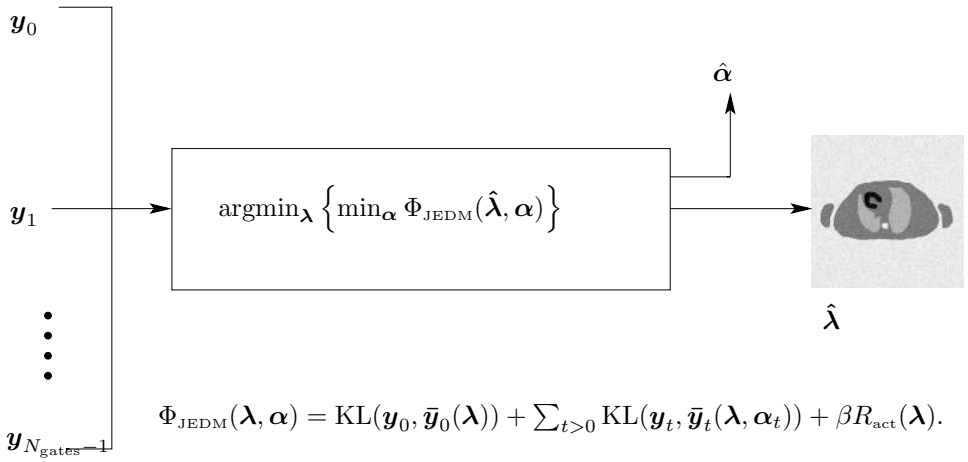

Figure 4.1: Flow diagram of the JEDM method.

2. Frame-Wise reconstruction with Post-Registration (FWPR). An FWPR method is our generic terminology for a method in which images $\{\hat{\boldsymbol{\lambda}}_t\}$, each reconstructed separately from the corresponding projection gate $\boldsymbol{y}_t$, are post-registered and consolidated in some way to produce a final image $\hat{\boldsymbol{\lambda}}$. There are two varieties that we consider here.

(a) **FWPR with Post-Averaging (FWPR-PA).** In the post-registration step, each $\hat{\boldsymbol{\lambda}}_t$, $t > 0$, is registered to the common target image $\hat{\boldsymbol{\lambda}}_0$. This yields deformation parameter estimates $\{\hat{\boldsymbol{\alpha}}_t\}$ satisfying, to some degree of accuracy

$$\boldsymbol{\mathcal{T}}(\hat{\boldsymbol{\alpha}}_t)\hat{\boldsymbol{\lambda}}_t \approx \hat{\boldsymbol{\lambda}}_0, \qquad 1 \leq t \leq N_{\text{gates}} - 1. \tag{4.14}$$

Consolidation is then accomplished by taking the weighted average (according to gate duration) of $\hat{\boldsymbol{\lambda}}_0$ and the $\{\boldsymbol{\mathcal{T}}(\hat{\boldsymbol{\alpha}}_t)\hat{\boldsymbol{\lambda}}_t\}$,

$$\hat{\boldsymbol{\lambda}} = \frac{\tau_0\hat{\boldsymbol{\lambda}}_0 + \sum_{t=1}^{N_{\text{gates}}-1} \tau_t \boldsymbol{\mathcal{T}}(\hat{\boldsymbol{\alpha}}_t)\hat{\boldsymbol{\lambda}}_t}{\sum_{t=0}^{N_{\text{gates}}-1} \tau_t}.$$

This approach, illustrated in Figure 4.2, is an intuitive and natural one, and seems to have occured independently to various investigators (e.g., [51, 96, 97, 47]). However, unlike JEDM, neither the determination of $\{\hat{\boldsymbol{\alpha}}_t\}$, nor the final consolidation of the $\{\hat{\boldsymbol{\lambda}}_t\}$ are based on a model for the measurement statistics.

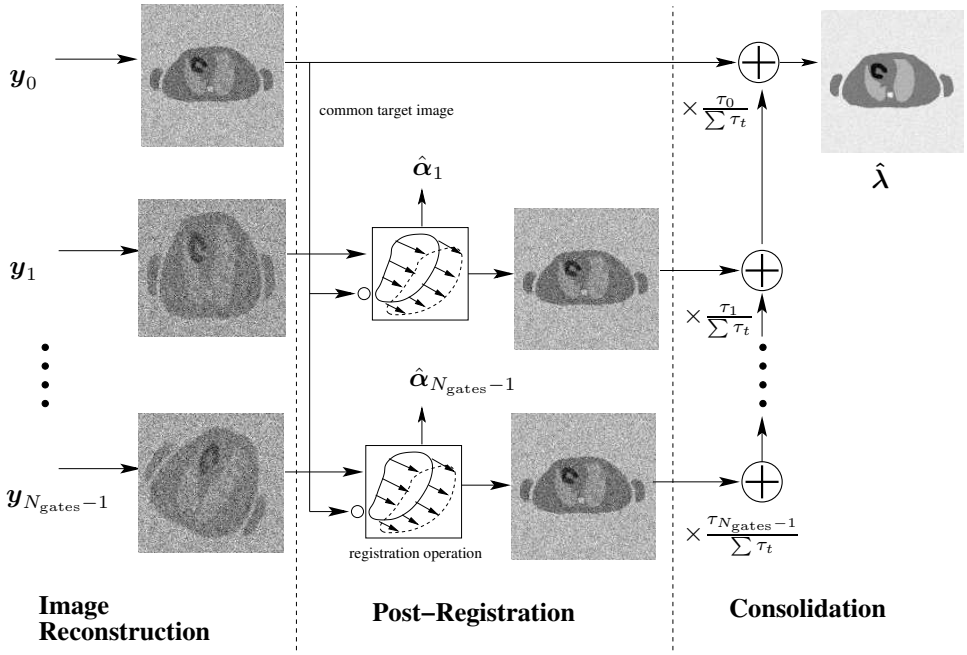

Figure 4.2: Flow diagram for the FWPR-PA method.

(b) **FWPR with Penalized-Likelihood Consolidation (FWPR-PLC).** In the post-registration step, each $\hat{\boldsymbol{\lambda}}_t$, $t > 0$, serves this time as a target image to which $\hat{\boldsymbol{\lambda}}_0$ is registered. This

yields deformation parameter vector estimate $\hat{\boldsymbol{\alpha}} = \{\hat{\boldsymbol{\alpha}}_t\}$ satisfying, to some degree of accuracy

$$\boldsymbol{\mathcal{T}}(\hat{\boldsymbol{\alpha}}_t)\hat{\boldsymbol{\lambda}}_0 \approx \hat{\boldsymbol{\lambda}}_t, \qquad 1 \leq t \leq N_{\text{gates}} - 1. \tag{4.15}$$

Consolidation is then accomplished by substituting these $\{\hat{\boldsymbol{\alpha}}_t\}$ into (4.11) as knowns and carrying out penalized likelihood minimization with respect to $\boldsymbol{\lambda}$,

$$\hat{\boldsymbol{\lambda}} = \underset{\boldsymbol{\lambda}}{\operatorname{argmin}} \left\{\Phi_{\text{JEDM}}(\boldsymbol{\lambda}, \hat{\boldsymbol{\alpha}})\right\}.$$

These steps are illustrated in Figure 4.3.

This variation is slightly more statistically principled than FWPR-PA, in that the consolidation step exploits the loglikelihood of the measurements. A similar approach has been considered recently in [79], but where the $\{\hat{\boldsymbol{\alpha}}_t\}$ are derived from the registration of gated CT scans. Here, we assume that at most a breath-hold CT scan is available, and so we must derive the deformation parameter estimates from the gated emission data.

Apart from the methods listed above, there are other notable variants that we will not consider, except perhaps in future work. These include an approach, proposed in [36], for reconstructing based on a motion-parametrized model for *ungated* projection data. The mean of the measurement vector $\boldsymbol{y}^{\text{ungated}}$ is essentially obtained by summing over (4.4) and (4.5),

$$\bar{\boldsymbol{y}}^{\text{ungated}}[\boldsymbol{\lambda}, \boldsymbol{\alpha}] = \boldsymbol{P}\left[\tau_0 \boldsymbol{I} + \sum_{t=1}^{N_{\text{gates}}-1} \tau_t \boldsymbol{\mathcal{T}}(\boldsymbol{\alpha}_t)\right]\boldsymbol{\lambda} + \boldsymbol{r} \tag{4.16}$$

Another approach, proposed in [37, 66] is to do penalized likelihood reconstruction based on (4.1), but with a penalty function involving motion parameters. The penalty term (and only the penalty term) encourages the solution $\{\boldsymbol{\lambda}_t\}$ to be related through a certain motion model. This is in contrast to JEDM in which the motion is described entirely in the loglikelihood $\text{KL}(\boldsymbol{y}, \bar{\boldsymbol{y}}(\boldsymbol{u}, \boldsymbol{\alpha}))$.

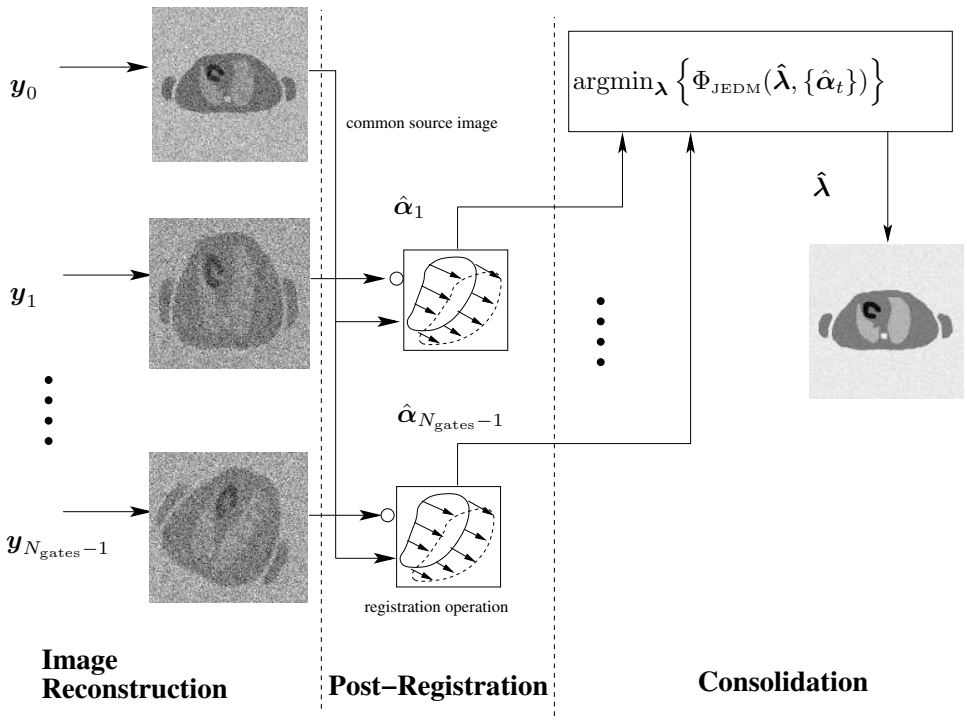

Figure 4.3: Flow diagram for the FWPR-PLC method.

4.4 Structure of Algorithm Implementations

In this section, we describe the skeleton of how we implement the reconstruction methods proposed in Section 4.3. We adhere to this implementation format in our experiments in Section 4.5. However, different implementation choices from those described below can be contemplated, and may be looked at in future work.

4.4.1 Transformation Model Parameters

The discrete deformation operators $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})$ and $\boldsymbol{W}(\boldsymbol{\alpha})$ will be based on deformation basis functions $\{b_{Ck}(x, y, z)\}_{C=\{X,Y,Z\}}$ of the form (2.25). Cubic B-splines β_3 shall be used (see (2.26)). As mentioned, this choice is a popular one for modeling the motion of non-rigid structures in the thorax.

The image intensity basis functions w_k in (3.2), shall also be cubic B-spline tensor products, i.e., $w(x) = \beta_3(x)$. Equivalently, we transform the voxel values $\boldsymbol{\lambda}$ using cubic

cardinal spline interpolation (see also Footnote 1). This intensity model has been used in [99, 92, 53, 54], although many people still use linear interpolation (e.g., [52, 101]). A principal attraction of the former is that cubic cardinal spline interpolation is an accurate approximation of sinc interpolation and, when implemented in the B-spline domain, is much cheaper computationally due to the small support of cubic B-splines. Sinc interpolation obviously has its appeal due to its roots in Shannon sampling theory. It has also been reported [82], that sinc-like interpolators reduce the presence of inflection points and/or sub-optimal local minima where the optimization algorithms can get stuck. This is probably linked to the higher order differentiability of cubic splines, as opposed to linear interpolation which is not even once differentiable.

Despite the potential advantages of $w(x) = \beta_3(x)$, it also leads to much less sparse $\boldsymbol{W}(\boldsymbol{\alpha})$ and so comes at a higher computational cost than linear interpolation. We shall discuss this in greater detail in Chapter 5.

4.4.2 Implementation of JEDM

Our current approach is to minimize Φ_{JEDM} over unconstrained $\boldsymbol{\alpha}$ and over $\boldsymbol{\lambda}$ constrained to the set[1]

$$\Theta_{\boldsymbol{\lambda}} \stackrel{\triangle}{=} \left\{ \boldsymbol{\lambda} \,:\, \boldsymbol{u} = (\boldsymbol{W}(\boldsymbol{0}))^{-1}\boldsymbol{\lambda} \geq \boldsymbol{0} \right\}. \tag{4.17}$$

By combining (4.9) and (4.17), one can reformulate (4.13) as follows

$$\Phi_{\text{JEDM}}(\boldsymbol{u}, \boldsymbol{\alpha}) = \text{KL}(\boldsymbol{y}, \boldsymbol{\mathcal{P}}(\boldsymbol{\alpha})\boldsymbol{u} + \boldsymbol{r}) + \beta R_{\text{act}}(\boldsymbol{W}(\boldsymbol{0})\boldsymbol{u}) \tag{4.18}$$

$$\hat{\boldsymbol{u}} = \underset{\boldsymbol{u}\geq\boldsymbol{0}}{\text{argmin}} \min_{\boldsymbol{\alpha}} \Phi_{\text{JEDM}}(\boldsymbol{u}, \boldsymbol{\alpha}) \tag{4.19}$$

$$\hat{\boldsymbol{\lambda}} = \boldsymbol{W}(\boldsymbol{0})\hat{\boldsymbol{u}}. \tag{4.20}$$

---

[1]This is an inevitable complication of using $w(x) = \beta_3(x)$. Ideally, we would like the minimization constrained to have non-negative activity samples in every gate, $\{(\boldsymbol{\lambda}, \boldsymbol{\alpha}) \,:\, \boldsymbol{\lambda} \geq \boldsymbol{0}, \boldsymbol{\mathcal{T}}(\boldsymbol{\alpha}_t)\boldsymbol{\lambda} \geq \boldsymbol{0}\}$, but this is a complicated, non-convex set. In light of (3.14), the set $\Theta_{\boldsymbol{\lambda}}$, defined in (4.17) is a convex subset of this, and one we believe is sufficiently large to find useful $\boldsymbol{\lambda}$ estimates.

The minimization of (4.19) is approached using the general coordinate block alternating minimization below.

Algorithm 4.4.1 (Block alternating JEDM implementation)

For $n = 1, \ldots, N_{\text{outer}}$

For $m = 1, \ldots, M_{\boldsymbol{u}}(n)$

Fix all variables but $\boldsymbol{u}$ and reduce $\Phi_{\text{JEDM}}(\boldsymbol{u}, \boldsymbol{\alpha})$ as a function of $\boldsymbol{u}$,

$$\Phi_{\text{JEDM}}(\boldsymbol{u}^{\text{old}}, \boldsymbol{\alpha}^{\text{old}}) > \Phi_{\text{JEDM}}(\boldsymbol{u}^{\text{new}}, \boldsymbol{\alpha}^{\text{old}}). \tag{4.21}$$

according to an appropriate sufficient decrease rule.

end

For $t = 1, \ldots, N_{\text{gates}} - 1$

For $m = 1, \ldots, M_{\boldsymbol{\alpha}}(n)$,

Fix all variables but $\boldsymbol{\alpha}_t$ and reduce $\Phi_{\text{JEDM}}(\boldsymbol{u}, \boldsymbol{\alpha})$ as a function of $\boldsymbol{\alpha}_t$,

$$\Phi_{\text{JEDM}}(\boldsymbol{u}^{\text{old}}, \boldsymbol{\alpha}^{\text{old}}) > \Phi_{\text{JEDM}}(\boldsymbol{u}^{\text{old}}, \ldots, \boldsymbol{\alpha}_t^{\text{new}}, \ldots). \tag{4.22}$$

according to an appropriate sufficient decrease rule.

end

end

end

The reader will observe in (4.18) that, for fixed $\boldsymbol{\alpha}$, Φ_{JEDM} as a function of $\boldsymbol{u}$ has the form of a standard penalized likelihood cost function, based on (2.8), but with projection matrix $\boldsymbol{\mathcal{P}}(\boldsymbol{\alpha})$. This means that the variety of specialized algorithms that have been devised over the years for PL estimation in PET can be used to accomplish (4.21). Apart from that, by reformulating the minimization in terms of $\boldsymbol{u}$, rather than $\boldsymbol{\lambda}$, manipulations of $\boldsymbol{\mathcal{P}}(\boldsymbol{\alpha})$

involve (see (4.10)) the sparse deformation matrices $\boldsymbol{W}(\boldsymbol{\alpha})$ instead of the denser $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})$. This parallels remarks made about (3.16) in Chapter 3.

We would like to ensure that Algorithm 4.4.1, even if it may not converge to a global minimum, at least has no non-stationary limit points. For this, the updates (4.21) and (4.22) must obey a sufficient decrease criterion. For monotonic algorithms, sufficient decrease is generally ensured (cf. [105, p. 91]) when the update operation yields, over the neighborhood any given non-stationary point, a uniform minimum decrease. It is straightforward to show this to be the case in Algorithm 4.4.1, when (4.21) are iterations of the the Separable Paraboloidal Surrogates (SPS) algorithm [29] and when (4.22) are iterations of steepest descent with the Armijo line search rule [7, p. 29]. This is the combination that we shall use.

With the Armijo rule, one pre-selects scalars $0 < \sigma, \gamma < 1$ and $s > 0$. When updating a cost function $f(\boldsymbol{\theta})$ along update direction $\Delta\boldsymbol{\theta}$, a step $(\gamma^n s)\Delta\boldsymbol{\theta}$ is made where n is the smallest integer satisfying,

$$f(\boldsymbol{\theta}) - f(\boldsymbol{\theta} + (\gamma^n s)\Delta\boldsymbol{\theta}) \geq -\sigma\gamma^n s \nabla^T f(\boldsymbol{\theta})\Delta\boldsymbol{\theta}. \tag{4.23}$$

The SPS algorithm is an example of an MM algorithm with quadratic tangent majorants. The quadratics are characterized by curvature parameters for which there are various choices. We use the minimum possible curvature parameter values (see [28, p. 62]) in all cases. This choice is expected to yield the fastest rate of descent.

In our experience, the deformation parameters $\boldsymbol{\alpha}$ converge fairly quickly, as compared to $\boldsymbol{u}$, presumably because the motion parameters are fewer in number and much more over-determined. We have therefore found it efficient to have $M_{\boldsymbol{\alpha}}(n)$ in Algorithm 4.4.1

be of the form

$$M_{\boldsymbol{\alpha}}(n) = \begin{cases} M^*_{\boldsymbol{\alpha}}, & n \leq N^{\max}_{\boldsymbol{\alpha}} \\ 0, & \text{otherwise} \end{cases}, \tag{4.24}$$

that is, to stop updating $\boldsymbol{\alpha}$ after a certain number of outer iterations $n = N^{\max}_{\boldsymbol{\alpha}}$.

4.4.3 Implementation of Motion-Free Penalized Likelihood

In the experiments to follow in Chapter 4.5, we will have occasion to do various Penalized Likelihood (PL) reconstructions that do not involve unknown motion parameters. To make these reconstructions comparable to JEDM, we implement them in terms of $\boldsymbol{u}$ as in (4.18) – (4.20), but with changes described as follows:

1. **Frame-Wise Reconstructions.** All frame-wise reconstructions in FWPR-PA and FWPR-PLC will be done with (4.18) replaced by

   $$\Phi_{\text{PL}}(\boldsymbol{u}) = \text{KL}(\boldsymbol{y}_t, \boldsymbol{P}\boldsymbol{W}(\boldsymbol{0})\boldsymbol{u} + \boldsymbol{r}_t) + \frac{\tau_t}{\sum_{\tilde{t}} \tau_{\tilde{t}}} \beta R_{\text{act}}(\boldsymbol{W}(\boldsymbol{0})\boldsymbol{u})$$

   Note that the regularization parameter β is scaled according to gate duration. This is so that the weight of the penalty term relative to the negative loglikelihood term is approximately the same in all gates t. In turn, this ensures that all frames undergo approximately the same degree of smoothing prior to being consolidated, and that this smoothing is comparable to that in ordinary JEDM.

2. **Penalized Likelihood with Known Motion.** Comparisons will be made with the case when gated data is PL reconstructed with known deformation parameter vector $\boldsymbol{\alpha}^{\text{true}}$. In this case, we will use

   $$\Phi_{\text{PL}}(\boldsymbol{u}) = \text{KL}(\boldsymbol{y}, \boldsymbol{\mathcal{P}}(\boldsymbol{\alpha}^{\text{true}})\boldsymbol{u} + \boldsymbol{r}) + \beta R_{\text{act}}(\boldsymbol{W}(\boldsymbol{0})\boldsymbol{u}).$$

   in place of (4.18). This case provides an upper performance bound against which methods that involve motion estimation can be measured.

3. **Ungated Penalized Likelihood.** Comparisons will be made with the conventional practice of reconstructing based on ungated projections. Accordingly, we carry out PL reconstructed based on $\sum_t \boldsymbol{y}_t$, using the cost function

$$\Phi_{\text{PL}}(\boldsymbol{u}) = \text{KL}(\sum_t \boldsymbol{y}_t, \boldsymbol{P}\sum_t \tau_t \boldsymbol{W}(\boldsymbol{0})\boldsymbol{u} + \sum_t \tau_t \boldsymbol{r}_t) + \beta R_{\text{act}}(\boldsymbol{W}(\boldsymbol{0})\boldsymbol{u}).$$

in place of (4.18).

As with JEDM, all PL iterations shall be done via SPS with optimum curvature parameters.

4.4.4 Implementation of Image Domain Registration

The frame-wise algorithms FWPR-PA and FWPR-PLC require image domain registration steps (4.14) and (4.15) respectively. Currently, we implement these registrations using the Sum of Squared Differences (SSD) criterion (see (3.16)). For these minimizations, we apply steepest descent with the Armijo step rule (4.23), just as with the cost function reductions (4.22).

4.4.5 Lesion Preserving Penalty Design Strategy

As has already been discussed, our principle aim in considering motion-corrected reconstruction methods is to combat the blur of small hot lesions that ignoring motion would otherwise induce. A natural concern, therefore, is whether the roughness penalty $R_{\text{act}}(\boldsymbol{\lambda})$, used in all of these methods, could introduce blur comparable to what motion correction removes. If so, this would detract from the effectiveness of the motion correction and could obscure the comparison of the different methods in Section 4.3.

As discussed in Section 2.2.5, one can set the ω_k weights in (4.12) to low values in regions where sharpness is desired. This is reasonable provided that one has *a priori* information about where these sharp features are located. In the case of the tumour quan-

tification task considered here, it is reasonable – in light of the growing preponderance of PET-CT scanners – to assume that this information is available from a breath-hold CT scan. If a breath-hold CT scan is available, we know the approximate location of the lesion in the PET image near full inspiration. In addition, we know the approximate amplitude of typical respiratory motion. These two pieces of information can be combined to localize the lesion to some generously sized ROI, whose dimensions are on the order of a few centimeters.

Once this ROI is identified, one can set appropriate ω_k to zero so that R_{act} will ignore voxels there. However, since many voxels outside of this ROI remain subject to the roughness penalty, we can still hope that the regularization will greatly reduce under-determined behavior and statistical variance in the reconstruction. We apply this practice throughout our experiments in Section 4.5. Some anecdotal support for it is given in Section 4.5.4.

4.5 Experiments

In this section, we conduct experiments using the various reconstruction methods, implemented as described in Section 4.4. The experiments were based on a simulated thorax acquisition which we describe next.

4.5.1 Simulated Data and Reconstruction Parameters

We simulated a 10 gate thorax acquisition as follows. We first obtained real thorax CT images at 5 levels of respiration. The CT images were downsampled to voxels of size $4 \times 4 \times 2$ millimeters and cropped to a grid size of $81 \times 105 \times 17$. These 5 images were co-registered using an $11 \times 14 \times 5 \times 3$ control point grid of cubic B-spline deformation basis functions using the Sum of Squared Differences (SSD) criterion. These grid dimensions along with Section 4.4.1 characterize the deformation model that we use throughout these experiments. Let $\{\boldsymbol{\alpha}_t\}_{t=1}^{4}$ denote the resulting motion parameter vector sequence

in order of increasing inspiration level. Using this sequence, we formed the following concatenation

$$\boldsymbol{\alpha}^{\text{true}} = (\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \boldsymbol{\alpha}_3, \boldsymbol{\alpha}_4, \boldsymbol{\alpha}_4, \boldsymbol{\alpha}_3, \boldsymbol{\alpha}_2, \boldsymbol{\alpha}_1, \mathbf{0})$$

to model a symmetric 10 gate breathing cycle.[2] This $\boldsymbol{\alpha}^{\text{true}}$ is what we used as the ground truth motion parameter vector in the simulation described here.

A synthetic ellipsoidal lung lesion of axial radius 2 mm and transaxial radius 4 mm was inserted into the gate 0 image. This image's intensity values were mapped to obtain a realistic PET activity distribution and also so that it lay within the set $\Theta_{\boldsymbol{\lambda}}$ (see (4.17)). We call this image $\boldsymbol{\lambda}^{\text{true}}$. A slice of $\boldsymbol{\lambda}^{\text{true}}$ containing the lung lesion is shown in Figure 4.4(a). A sense of the lesion blur that would result from an ungated reconstruction is given in Figure 4.4(b) where the superposition of the image from all gates is shown.

A mean Poisson projection vector $\bar{\boldsymbol{y}}((\boldsymbol{W}(\mathbf{0}))^{-1}\boldsymbol{\lambda}^{\text{true}}, \boldsymbol{\alpha}^{\text{true}})$, as described by (4.9) was simulated. The forward projection operator $\boldsymbol{P}$ was 2D and based on a line integral model (see, e.g., [89]) discretized into a 128 angular by 105 radial system of bins in every image slice. For the present study, equal gate durations τ_t were used. This choice deviates from some proposed sinusoidal models for patient breathing patterns (e.g., [63, 87]). However, in our private communication with clinicians, we have seen no consensus on whether a lung cancer patient's breathing would follow such a regular pattern, considering that the lung tumor might interfer with it. We therefore postpone the problem of realistically simulating the relative τ_t to future work. The τ_t were chosen, however, so as to obtain 80K mean total counts per axial millimeter of the image, 30% of which were accounted for in the background term $\boldsymbol{r}$. These are typical count and background levels in thorax acquisitions on the ECAT HR+ scanner.

---

[2]However, our subsequent reconstruction experiments will use *no* knowledge of the breathing cycle's symmetry! Each motion correction method will estimate 9, and not 4, separate $\boldsymbol{\alpha}_t$ vectors.

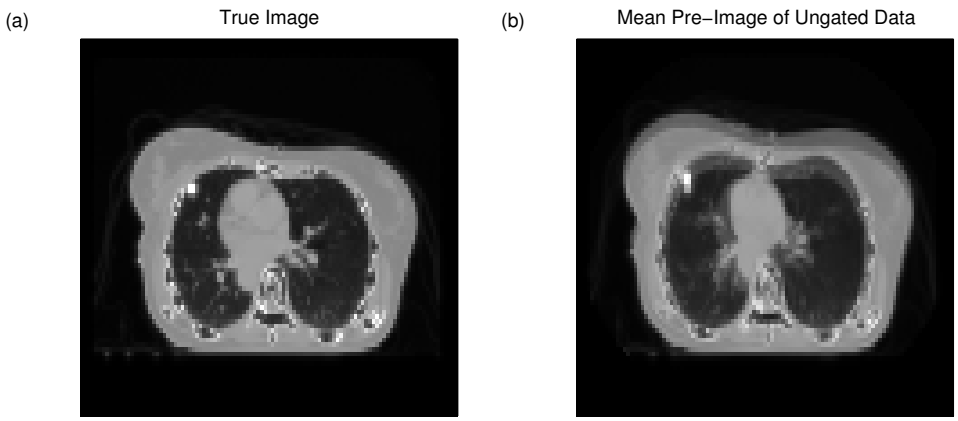

Figure 4.4: (a) The true activity (slice #8) in gate 0. (b) The superposition of the true activity (slice #8) from all gates.

We added 20 realizations of Poisson noise to the mean measurement $\bar{\boldsymbol{y}}$. On each data realization, the three reconstruction methods of Section 4.3 were performed. In addition, an ungated PL reconstruction and a gated PL reconstruction with known motion were performed. The implementation of all of these are as described in Section 4.4. Our aim here is to assess, as far as possible, the performance of the various methods when optimally implemented. This is complicated by the possibility of the various minimization operations involved getting trapped at sub-optimal local minima. In an effort to reduce the chance of this, we have initialized all minimization sub-operations in all the reconstruction methods at the true value of the parameter that they are trying to estimate (that is, $\boldsymbol{\lambda}^{\mathrm{true}}$, $\boldsymbol{\alpha}^{\mathrm{true}}$, $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha}_t^{\mathrm{true}})\,\boldsymbol{\lambda}^{\mathrm{true}}$ etc…).

Iteration parameters for the various minimization sub-problems were chosen based on some preliminary empirical tests. For every SSD registration sub-problem (see Section 4.4.4), 300 iterations were used. For every, PL sub-problem (see Section 4.4.3), 1500 iterations were used. For the JEDM reconstructions, the iteration parameters of Algorithm 4.4.1 were $N_{\mathrm{outer}} = 45$ and $M_{\boldsymbol{u}}(n) = 50$ for all n. We chose $M_{\boldsymbol{\alpha}}(n)$ as in (4.24)

with $M_{\boldsymbol{\alpha}}^{*} = 30$ and $N_{\boldsymbol{\alpha}}^{\max} = 15$.[3] We chose the parameters $\gamma = \sigma = 0.5$ in all Armijo line search operations (recall (4.23)). The parameter s was chosen adaptively so that the initial step magnitude $||s\Delta\boldsymbol{\alpha}||_{\infty}$ in the very first search step would equal 10 voxel lengths.

Roughness penalties were applied with $\beta = 3.125 \times 10^{-7}$. An approximately $3.2 \times 3.2 \times 1.2$ cm ROI around the lung lesion (see Figure 4.5) was excluded from R_{act}, as described in Section 4.4.5. The lesion, throughout its true motion, remains within this ROI.

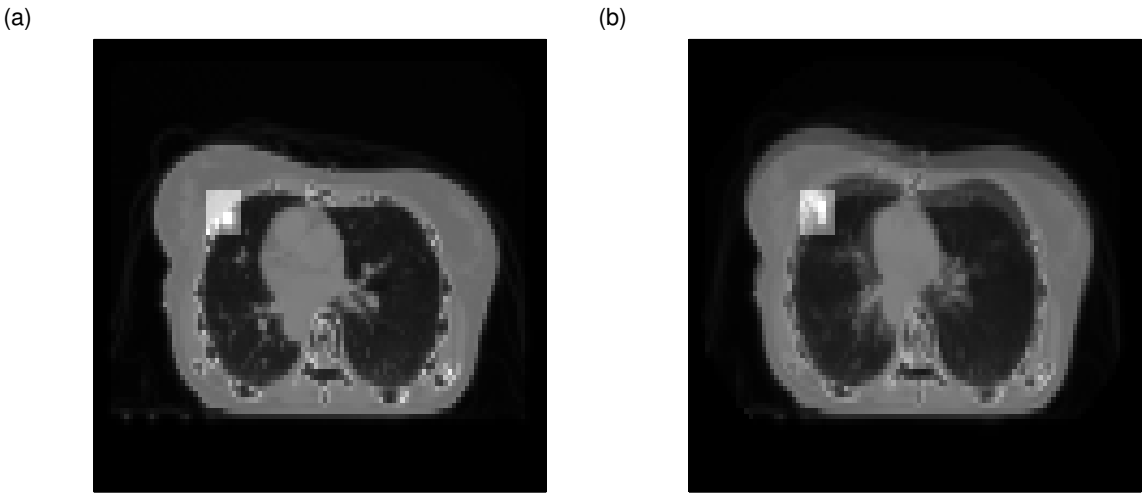

Figure 4.5: The same as Figure 4.4, except the ROI around the lesion excluded from the penalty R_{act} is shown superimposed. Part (b) is to show this ROI encompasses the motion of the lesion.

4.5.2 Activity Estimation and Lesion Recovery Performance

Denoting U as the lesion uptake in the reconstruction and U^{true} as that in $\boldsymbol{\lambda}^{\text{true}}$, the performance of the various reconstructions were quantified in terms of the absolute perecent uptake error,

$$\Delta U = \frac{|U - U^{\text{true}}|}{U^{\text{true}}} \times 100\%.$$

[3]Note that our proposed method, JEDM, gets more total passes over $\boldsymbol{u}$ and $\boldsymbol{\alpha}$ than the other methods. Because all minimizations are initialized at the ideal, true parameter values however, this is, if anything, a handicap for JEDM, not an advantage. More passes over the variables gives the algorithm iterates additional opportunity to retreat from the ideal initial values.

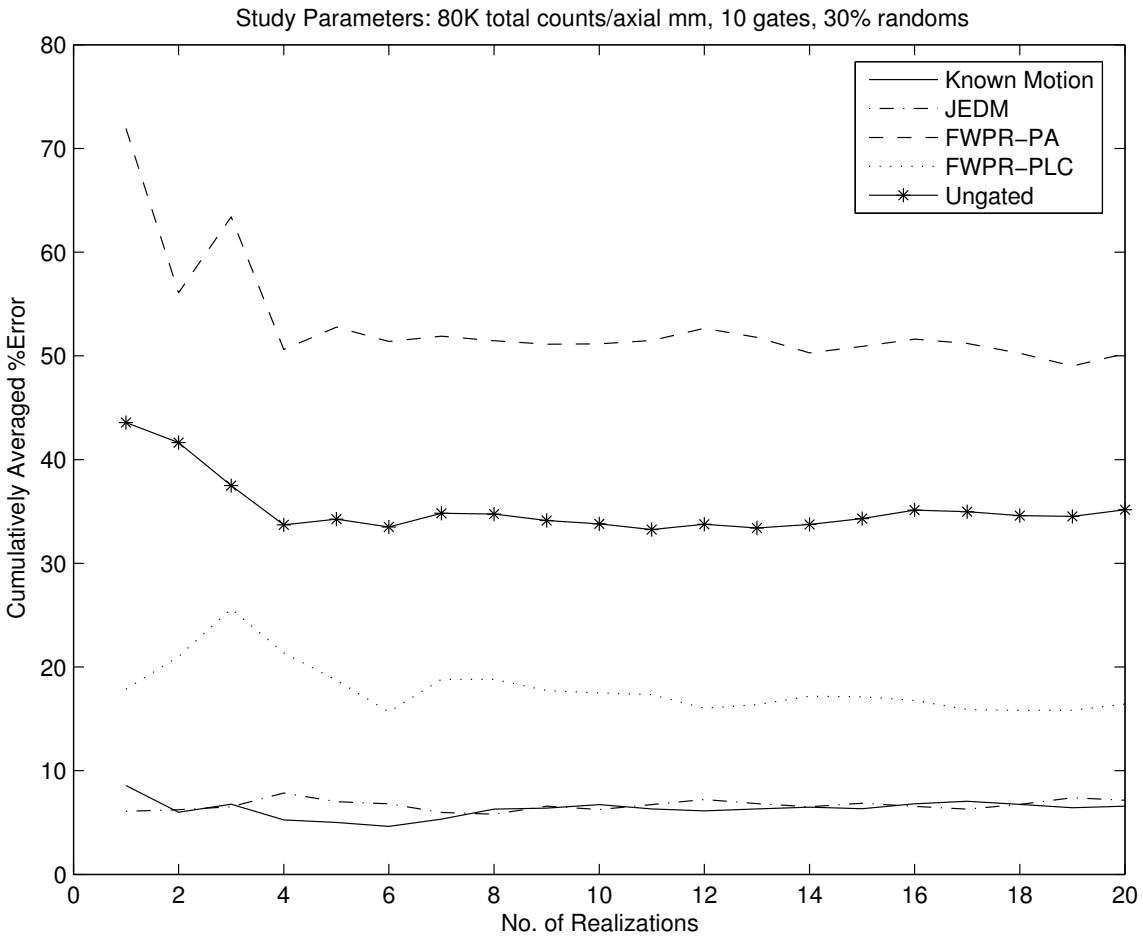

Figure 4.6: Cumulative mean percent error of the lesion tracer uptake for the various reconstruction methods.

The cumulative means of ΔU, as a function of number of realizations, are plotted for the different reconstruction methods in Figure 4.6.

Sample reconstructions based on one of the 20 realizations are displayed in Figure 4.7. A common slice from each reconstruction method and from the true activity image is shown. The mean of these images is shown in Figure 4.8.

In these preliminary results, our proposed method JEDM demonstrated the best performance in terms of average lesion quantification accuracy (see Figure 4.6). The accuracy was close to that of the idealized case when the motion vector $\boldsymbol{\alpha}^{\text{true}}$ was known. (This was in spite of the handicap mentioned in Footnote 3). The second best performance was that

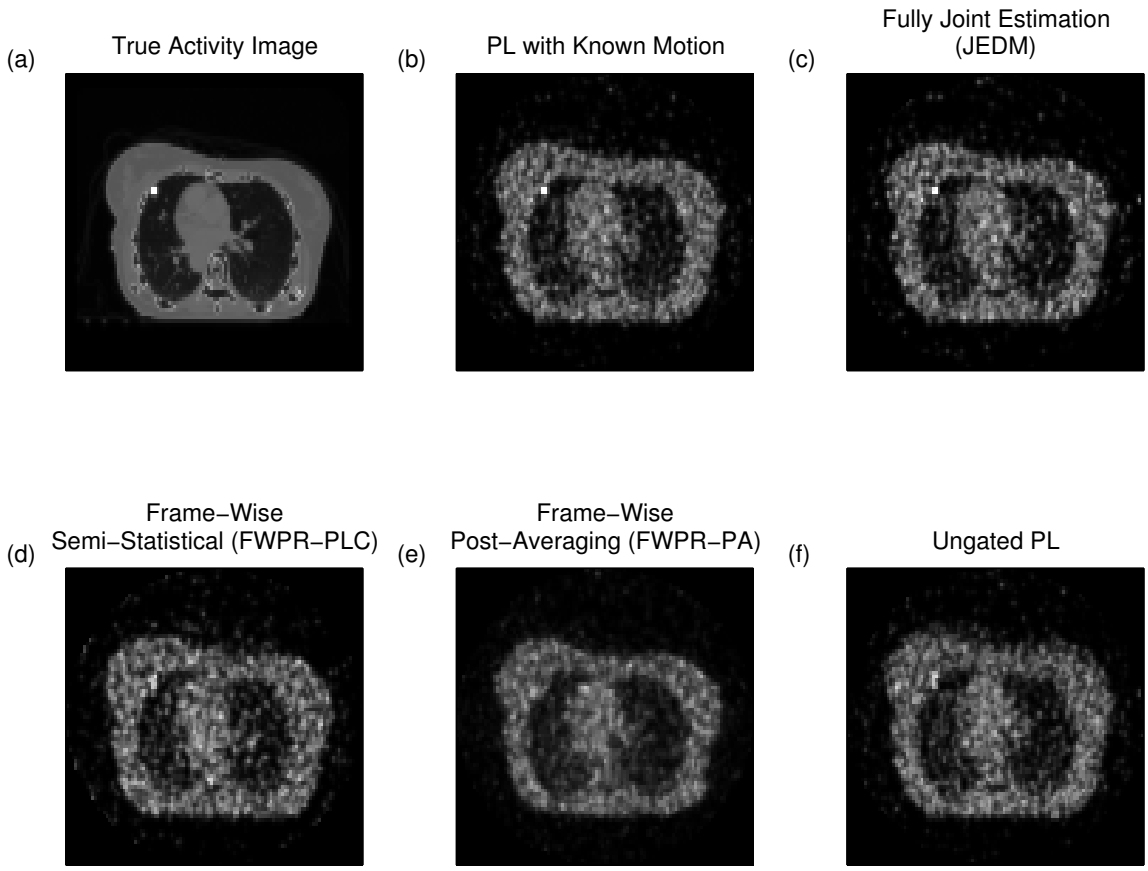

Figure 4.7: Slice #8 of the reconstructed image, as reconstructed by various methods, for one realization of the data.

of FWPR-PLC which, in terms of absolute quantification performance, did not lag dramatically far behind JEDM, and certainly outperformed the established custom of simply ignoring motion and reconstructing from ungated data.

The FWPR-PA method exhibited the worst lesion quantification, worse even than the ungated reconstructions. This may have been due, in part, to the fact that (4.14) essentially tries to find a $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha})$ that will unwarp $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha}_t^{\text{true}})\boldsymbol{\lambda}^{\text{true}}$. However, small lesion-like features in $\boldsymbol{\mathcal{T}}(\boldsymbol{\alpha}_t^{\text{true}})\boldsymbol{\lambda}^{\text{true}}$ would be sensitive to voxel interpolation error. That would make them difficult to unwarp, and could have independently contributed to the misquantification of the lesion. Additional experiments will be needed to determine how much misquantification

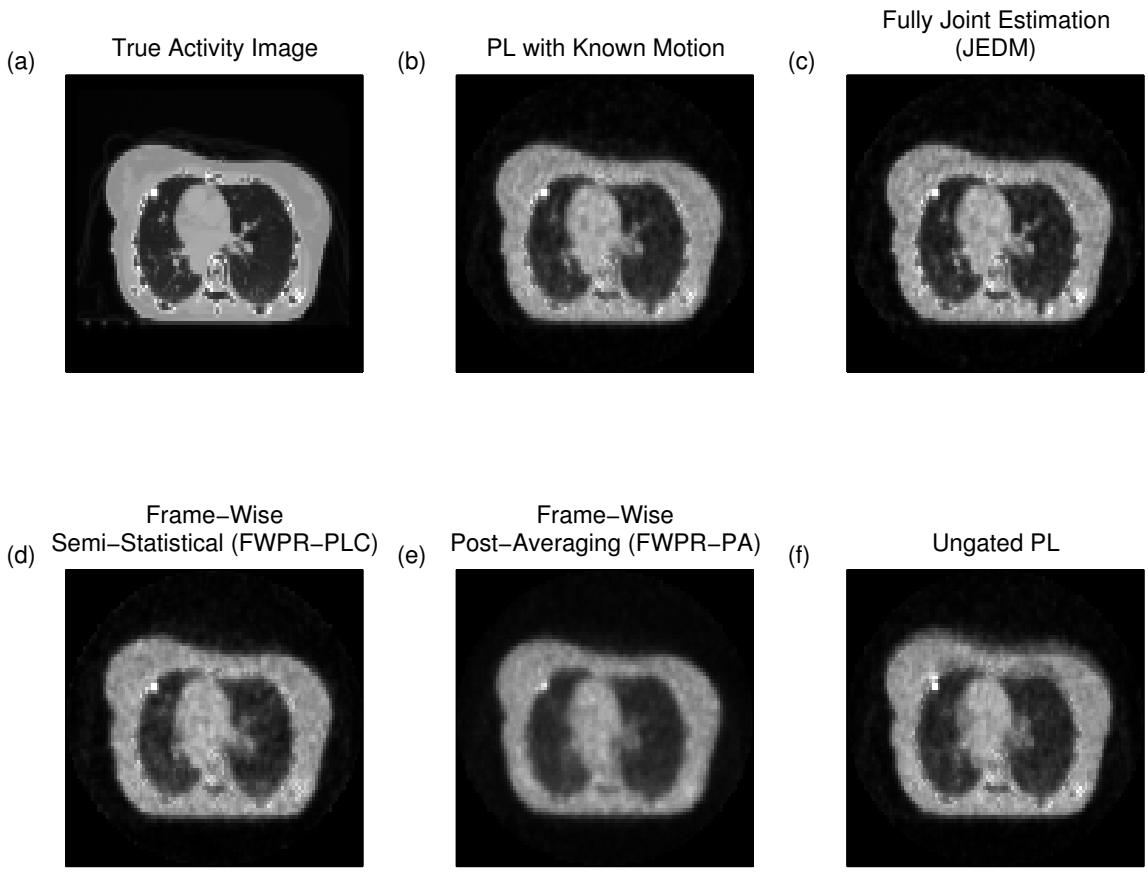

Figure 4.8: Slice #8 of the mean image, as reconstructed by various methods, and averaged over 20 realizations.

in FWPR-PA is due to interpolation error and how much is due to misregistration.

Based on Figure 4.8, however, both frame-wise methods seem to have suffered, as expected, from similarly high variance in registration accuracy. This is evident by the blur in Figures 4.8(d) and (e) not only in the region of the lesion, but throughout the images as a whole. Because of this, there seems little reason, at this point, to expect that FWPR-PA would outperform FWPR-PLC, let alone compete with JEDM.

4.5.3 Motion Estimation Performance

Using the estimated deformation parameters $\{\hat{\boldsymbol{\alpha}}_t\}$ from JEDM and FWPR-PLC, we calculated the position of the center of the lesion in each gate. These estimated trajectories

are compared to the true trajectory in Figure 4.9. The three panels in the figure show the position in each Cartesian direction of motion $C \in \{X, Y, Z\}$. In the present context, the X-direction of motion was the front-to-back motion of the patient, Y was left-to-right motion, and Z was axial motion. In the plots, the error bar radius is one standard error. The corresponding standard deviations, but averaged across gates, are tabulated in Table 4.1.

The trends observed in this performance data are consistent with observations in Section 4.5.2. The JEDM motion estimation accuracy is greater, on the whole, than that of the frame-wise method FWPR-PLC. Interestingly, the discrepancy is most pronounced (see Figure 4.9) in the left-right direction of motion. Moreover, the variance of FWPR-PLC is higher. Again, this is expected considering that JEDM estimates motion based on the full set of projection data whereas FWPR-PLC does not.

Table 4.1: The standard deviations, averaged across gates, of the estimated lesion trajectories for each Cartesian direction of motion. Values are in millimeters.

	Front-Back	Left-Right	Axial
FWPR-PLC	2.53	2.73	1.07
JEDM	1.63	1.67	0.82

4.5.4 Anecdotal Test of Lesion Preserving Penalization Strategy

Here, we look at an anecdotal test of the lesion preserving penalty design strategy discussed in Section 4.4.5. Reconstructions were made, shown for one slice in Figure 4.10, to test various relevant combinations of cases. The percent tracer recovery in the lesion for each case is also shown. The reconstructions were made using half as many gates and three times as many total counts as the data simulation described in Section 4.5.1, but is essentially the same otherwise.

Figures 4.10(a) and (b) look at converse cases. In (a), a moderate roughness penalty is applied uniformly and indiscrimantly throughout the image, but motion is accounted for (using JEDM) in the reconstruction. Conversely, in (b), roughness penalties are completely

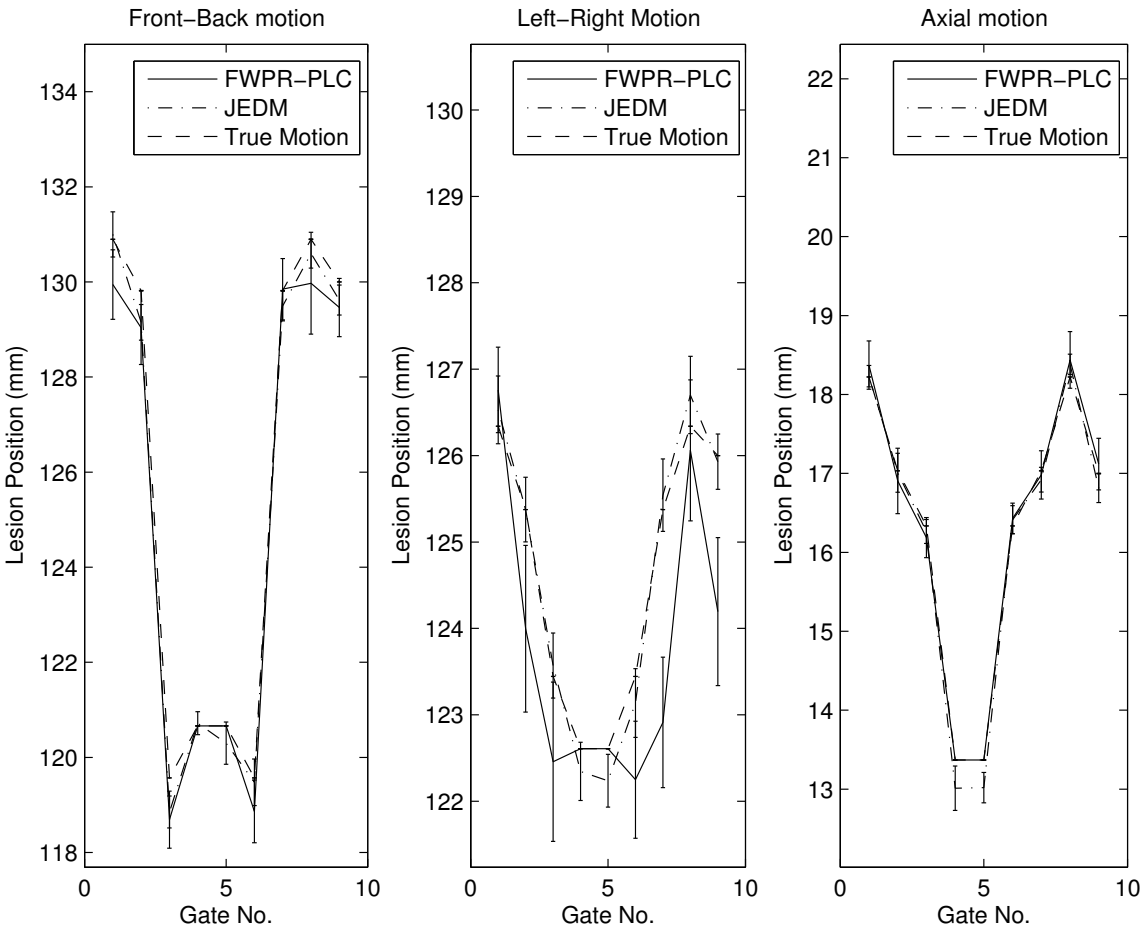

Figure 4.9: The trajectory of the center of the lesion, along with the mean trajectories as estimated by JEDM and FWPR-PLC, plotted for each Cartesian direction of motion. The radius of the error bars are a single standard error, taken over the 20 realizations.

omitted, but the data is ungated and motion effects are completely ignored. We see that the percent lesion tracer recoveries for these are not dramatically different. This seems to support our earlier conjecture, namely, that indiscriminant regularization might blur a lesion to a degree similar to that of ignoring motion. Of course, (a) is an improvement over (b) in terms of SNR, because of the reduced background variance effected by the penalty.

A comparison of Figure 4.10(c) with (a) and (d), however, suggests that the potential benefits of the motion correction is more fully realized when the ROI exclusion technique is used. In (d), we observe the case where blur is contributed neither by a roughness penalty

(since the reconstruction was purely ML) nor by unknown motion (since the reconstruction incorporated $\boldsymbol{\alpha}^{\text{true}}$). Accordingly, the maximum lesion recovery of the 4 cases is observed in (d). In (a), conversely, one observes the minimum background variance of the 4 cases, due to the uniform penalty. Case (c) exhibits the best of both worlds: the background noise is similar to (a), yet the lesion recovery is very close to that of (d).

These initial tests suggest, therefore, that motion correction is most effective when the penalty is judiciously designed, so as to incorporate side information.

4.6 Conclusion

We conducted reconstruction experiments to compare the ability of various algorithms to quantify an FDG-like hot lesion in the lung. In the simulated conditions of these experiments, the lung lesion was subject to significant inter-gate motion and the total count levels imitated those encountered in common medical practice. Moreover, the simulated anatomy and motion were both derived from gate-wise CT reconstructions of an actual human patient.

In these experiments, we compared a statistically principled method, JEDM, to less statistically principled ones based on frame-wise reconstruction approaches. The JEDM method was expected to perform better in part because all motion parameters are estimated based on the full set of gated projection data. This stands in contrast to the frame-wise methods in which the motion parameter vector at each gate is derived only from two gates of low count projection data. These expectations seem to be supported by our experiments. Not only did the JEDM method outperform the frame-wise methods in terms of lesion quantification, but significantly more blur was observed throughout the mean reconstructed image of the frame-wise methods. This suggests that the frame-wise methods had difficulty in the registration step as a whole. A comparison of motion estimation accuracy

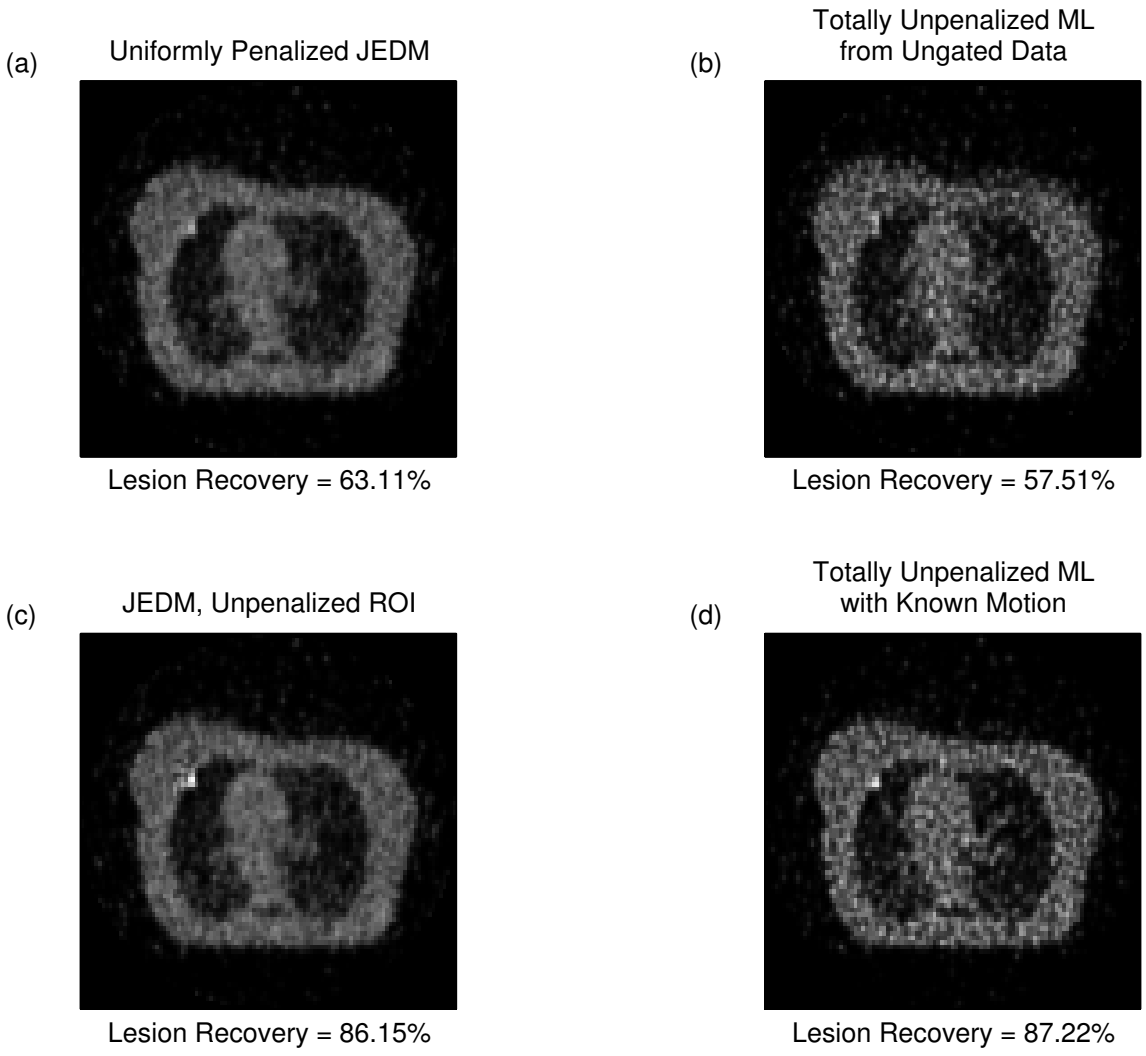

Figure 4.10: Anecdotal comparison of the effect, on tracer recovery, of (a) indiscrimant roughness penalization but accounting for motion, (b) penalization totally absent but ignoring motion, (c) selective penalization and accounting for motion, and (d) penalization totally absent and ideal motion correction.

of JEDM and FWPR-PLC was also made in Section 4.5.3, and lent further support to these conclusions.

For its part, the JEDM exhibited promising performance. Its average lesion misquantification came very close to the case where motion was fully known. In addition, the absolute levels of lesion misquantification were quite small $\sim 10\%$. It is likely that this was enabled by our lesion preserving penalization technique. In Section 4.5.4, we found that this technique improves SNR relative to the practice of indiscrimantly penalizing throughout the image.

In our future work, there are two principal issues that need to be explored further. One issue is the sensitivity of the various reconstruction methods to local minima. An effort was made to circumvent this issue by initializing all reconstructions at truth. Since this is possible only in simulation, we will need to see if known strategies for avoiding local minima can be used to recover the high performance of JEDM when ground truth is unknown. These strategies would include multi-resolution pyramid techniques from the image registration literature (e.g., [99, 92]).

Apart from this, there are still more realistic degrees of simulation to be tried. For example, our simulations so far have assumed no modeling error. Both the reconstructions and the simulations were based on identical models of motion and count acquisition. Finally, of course, we would like to compare these methods on real projection data.

CHAPTER 5

Accelerating Interpolation Operations Using Pre-Computation Strategies

5.1 Introduction

This chapter contains the second branch of our work, and is based on [48]. We examine here several pre-computation techniques that can be used to reduce the computational effort in voxel intensity interpolations. These interpolation operations are common across registration tasks based on the image transformation framework of Chapter 3. In such registration tasks, one faces the need to iteratively reduce cost functions

$$\Phi(\boldsymbol{\alpha}) = \boldsymbol{\xi}(\boldsymbol{W}(\boldsymbol{\alpha})\boldsymbol{u}) \tag{5.1}$$

An example is the SSD cost function (3.16). Also, for fixed $\boldsymbol{u}$, the iterative reductions in (4.22) are of a cost function of the form (5.1). Other techniques for accelerating registration include randomized downsampling [52] and multiresolution approaches (e.g., [99, 92]). The techniques that we discuss here can easily be combined with these to obtain additional speed-up factors.

It will be convenient, in what follows, to express $\boldsymbol{W}(\boldsymbol{\alpha})\boldsymbol{u}$ alternatively as,

$$\begin{aligned}\boldsymbol{W}(\boldsymbol{\alpha})\boldsymbol{u} &= \mathcal{W}^{\boldsymbol{u}}_{www}(\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{\alpha}) \\ &= \mathcal{W}^{\boldsymbol{u}}_{www}(\boldsymbol{\nu}_X + \boldsymbol{B}_X\boldsymbol{\alpha}_X, \boldsymbol{\nu}_Y + \boldsymbol{B}_Y\boldsymbol{\alpha}_Y, \boldsymbol{\nu}_Z + \boldsymbol{B}_Z\boldsymbol{\alpha}_Z)\end{aligned}$$

where we have introduced $\mathcal{W}^{\boldsymbol{u}}_{fgh}(\cdot)$ to denote the interpolations of $\boldsymbol{u}$ performed at the various of deformed grid points determined by $\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{\alpha}$,

$$\begin{aligned}[\mathcal{W}^{\boldsymbol{u}}_{fgh}(\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{\alpha})]_j \stackrel{\triangle}{=} \sum_{mnp} u_{mnp} f\left([\boldsymbol{\nu}_X + \boldsymbol{B}_X\boldsymbol{\alpha}_X]_j - m\right) \times \\ g\left([\boldsymbol{\nu}_Y + \boldsymbol{B}_Y\boldsymbol{\alpha}_Y]_j - n\right) \times \\ h\left([\boldsymbol{\nu}_Z + \boldsymbol{B}_Z\boldsymbol{\alpha}_Z]_j - p\right)\end{aligned} \tag{5.2}$$

for all $1 \leq j \leq |\mathcal{G}|$ and using the tensor product of 1D interpolators f, g, and h. With this, we can re-express (5.1) as

$$\Phi(\boldsymbol{\alpha}) = \boldsymbol{\xi}(\mathcal{W}^{\boldsymbol{u}}_{www}(\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{\alpha})). \tag{5.3}$$

The computational cost of evaluating Φ is influenced by the respective computational costs of $\boldsymbol{B}$, $\mathcal{W}^{\boldsymbol{u}}_{www}()$, and $\boldsymbol{\xi}$. Analogous remarks hold for evaluations of the derivatives of Φ. Taking the gradient in (5.3) with respect to $\boldsymbol{\alpha}_X$, for example, leads to,

$$\nabla_{\boldsymbol{\alpha}_X}\Phi = \boldsymbol{B}_X^T\mathcal{W}^{\boldsymbol{u}}_{\dot{w}ww}(\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{\alpha})\nabla\boldsymbol{\xi} \tag{5.4}$$

Note here that the gradients of $\mathcal{W}^{\boldsymbol{u}}_{www}(\cdot)$, that being $\mathcal{W}^{\boldsymbol{u}}_{\dot{w}ww}(\cdot)$, are again interpolation operations of a form and complexity comparable to $\mathcal{W}^{\boldsymbol{u}}_{www}(\cdot)$.

In this chapter, we will examine methods for accelerating the evaluations of the interpolation steps $\mathcal{W}^{\boldsymbol{u}}_{www}(\cdot)$ (which will apply equally well to the interpolation steps $\mathcal{W}^{\boldsymbol{u}}_{\dot{w}ww}(\cdot)$, etc… in the derivatives). We will restrict attention throughout to the case where the w in

(3.2) are cubic B-splines,

$$w(t) = \beta_3(t) = \begin{cases} \frac{|t|^3}{2} - |t|^2 + \frac{2}{3}, & 0 \leq |t| \leq 1 \\ \frac{-(|t|-2)^3}{6}, & 1 \leq |t| \leq 2 \cdot \\ 0, & |t| \geq 2 \end{cases}$$

Then, taking note of (3.5), we have

$$[\mathcal{W}^{\boldsymbol{u}}_{www}(\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{\alpha})]_j = \sum_{mnp} u_{mnp} \beta_3 \left(d_{jX}(\boldsymbol{\alpha}_X) - m\right) \beta_3 \left(d_{jY}(\boldsymbol{\alpha}_Y) - n\right) \beta_3 \left(d_{jZ}(\boldsymbol{\alpha}_Z) - p\right) \tag{5.5}$$

The ideas in this section readily generalize to other w. Focusing on cubic B-splines is in part to simplify presentation, in part due to the popularity of this choice among certain researchers (e.g., [99, 92, 53, 54]), and in part for their relevance to our work in Chapter 4. There, we used cubic B-splines for reasons discussed in Section 4.4.1. In addition, cubic B-splines are a practically relevant case because the benefits of using them, as compared to the more traditional choice of linear interpolation (i.e., first order B-splines), are bought at significantly greater computational expense. In 3D linear interpolation, one must interpolate a series of $2 \times 2 \times 2$ blocks of image values, whereas the wider cubic B-splines involve blocks of size $4 \times 4 \times 4$. Thus, 8 times more computation is required.

In focusing on cubic B-splines, we are is in part to simplify presentation and for relevance to the specifics of our experiments in Chapter 4. it is because there is a certain popularity to the choice $w = \beta_3$ among other researchers . As discussed in Section 4.4.1, the more traditional use of higher order B-splines over the more common choice of

The impact of accelerating $\mathcal{W}^{\boldsymbol{u}}_{www}(\cdot)$ on CPU time will depend on how complicated $\boldsymbol{B}$ and $\boldsymbol{\xi}$ are in relation to $\mathcal{W}^{\boldsymbol{u}}_{www}(\cdot)$. There are several factors that can mitigate the contributions of $\boldsymbol{B}$ and $\boldsymbol{\xi}$ that will be important for us. These are:

1. $\boldsymbol{B}$ is simple as given. For example, in the case of affine registration problems, multiplication with $\boldsymbol{B}$ involves only only $12|\mathcal{G}|$ scalar multiplications and $3|\mathcal{G}|$ memory transfers. Conversley, evaluating $\mathcal{W}^{\boldsymbol{u}}_{www}(\cdot)$ can require $132|\mathcal{G}|$ multiplications and $65|\mathcal{G}|$ memory transfers (see Section 5.2).

2. The minimization algorithm being used involves line searches. Looking at $\Phi(\boldsymbol{\alpha})$ along a hypothetical ray $\{\boldsymbol{\alpha} + \ell\Delta\boldsymbol{\alpha} \,:\, 0 \leq \ell \in \mathbb{R}\}$, one obtains, in light of (5.3),

$$a(\ell) = \boldsymbol{\xi}(\mathcal{W}^{\boldsymbol{u}}_{www}(\underbrace{\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{\alpha}}_{\triangleq \boldsymbol{x}} + \ell \underbrace{\boldsymbol{B}\Delta\boldsymbol{\alpha}}_{\triangleq \Delta\boldsymbol{x}})). \tag{5.6}$$

   Hence, the repeated evaluation of $a(\ell)$ for different ℓ in line search operations is not encumbered by multiplications with $\boldsymbol{B}$, once the vectors $\boldsymbol{x}$ and $\Delta\boldsymbol{x}$ have been computed.[1] The other operations $\boldsymbol{\xi}$ and $\mathcal{W}^{\boldsymbol{u}}_{www}(\cdot)$ therefore become the computational bottlenecks during the line search.

3. $\boldsymbol{\xi}$ is simple as given. In $\Phi_{\text{SSD}}(\boldsymbol{\alpha})$, for example, $\boldsymbol{\xi} = \frac{1}{2}||\boldsymbol{z} - \boldsymbol{f}^{\text{targ}}||^2_{\ell_2}$ is a simple norm evaluation and its gradient is a simple vector subtraction $\boldsymbol{z} - \boldsymbol{f}^{\text{targ}}$.

4. Minimization of $\Phi(\boldsymbol{\alpha})$ is accomplished via an MM algorithm, in which the tangent majorants are derived by taking $\boldsymbol{\xi}\,(\boldsymbol{z})$ in (5.3) and replacing it with a simpler $\phi(\boldsymbol{z}, \bar{\boldsymbol{z}})$. This scenario arises in Section 6.6.1.

The rest of the chapter is organized as follows. In Section 5.2, we dissect computations involving $\mathcal{W}^{\boldsymbol{u}}_{www}(\cdot)$ and $\boldsymbol{B}$, looking at the different contributions to their computational expense. In Section 5.3, we describe acceleration techniques that we devised. One of these ideas is a tabulation techniques that reduces the number of grid points participating in each interpolation. The second idea uses block alternation to make various computed quantities

[1] This is a recognized line search implementation strategy for cost functions involving compositions with an affine function (e.g., [10, p. 508]).

reusable. Some tests of these techniques are made in Section 5.4. In Section 5.4.1, the techniques are tested in a simple 2D torso phantom registration via the SSD criterion (3.15). In Section 5.4.2, we present some computational tests relevant to the minimizations with respect to $\boldsymbol{\alpha}$ in the JEDM method of Chapter 4.

5.2 The Components of the Computations

In this section, we look at the number of multiplications and the number of memory transfers to and from RAM in the various operations associated with the evaluation of $\boldsymbol{W}(\boldsymbol{\alpha}) = \mathcal{W}^{\boldsymbol{u}}_{www}(\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{\alpha})$.

5.2.1 The Components of the Intensity Interpolations

We start be considering the number of multiplications and memory transfers needed to compute the RHS of (5.5) for each $\boldsymbol{d}_j$. Because the cubic B-splines β_3 are of width 4 grid points, the interpolation can be limited to a $4 \times 4 \times 4$ neighbourhood of grid points u_{mnp} around $\boldsymbol{d}_j(\boldsymbol{\alpha})$. Moreover, since the 3D interpolator is a tensor product of 1D interpolation functions, efficient computation demands that one interpolate the u_{mnp} in consecutive passes, first along the m-direction, then the n-direction, and finally in the p-direction. There are two components to this calculation. First, one must compute the interpolation weights $\beta_3(d_{jX}(\boldsymbol{\alpha}_X) - m)$, $\beta_3(d_{jY}(\boldsymbol{\alpha}_Y) - n)$, and so forth. For the interpolation along each grid axis, there will be 4 such weights, each one computed by evaluating a cubic polynomial. Each cubic polynomial evaluation takes 4 multiplications, for a total of $4 \times 4 \times 3 = 48$ multiplications. Secondly, one must take these weights and carry out the actual 3-pass interpolation of the $4 \times 4 \times 4$ grid values in u_{mnp}. This requires $64 + 16 + 4 = 84$ multiplications. The total over all of these steps is $48 + 84 = 132$ multiplications. This breakdown is summarized in Table 5.1. To aid later discussion, we have also include there the analgous breakdown in 2D registration for the interpolations over 4×4 regions done

by $\mathcal{W}^{\boldsymbol{u}}_{ww}(\cdot)$.

Estimates of the number of memory transfers of these operations are tabulated in Table 5.2. In the case of $\mathcal{W}^{\boldsymbol{u}}_{www}(\cdot)$, each interpolation operation requires that a $4 \times 4 \times 4$ sub-block of u_{mnp} be accessed from RAM. The result of the interpolation is then put back in RAM for a total of $64 + 1 = 65$ memory transfers.

5.2.2 The Components of the Deformation Map Computations

Here, we consider the number of multiplications and memory transfers needed to carry out a multiplication with $\boldsymbol{B}$ (or $\boldsymbol{B}^T$) needed to compute the deformation map vector.

For affine deformations the basis functions are $\{b_{Ck}(x, y, z)\} = \{x, y, z, 1\}$, $C \in \{X, Y, Z\}$ (see also Section 2.3.2) meaning that each $\boldsymbol{B}_C$, is a $|\mathcal{G}| \times 4$ matrix. Multiplication with $\boldsymbol{B}$ therefore involves $4 \times 3 = 12$ multiplications per grid point. In terms of memory transfers, it is reasonable to suppose that the 12 motion parameters can be held permanently in cache and that the values of $\boldsymbol{B}$ can be generated on-the-fly with in the CPU. Therefore, the only memory transfers of significance are those required to put the $3|\mathcal{G}|$ resulting values of $\{\boldsymbol{d}_j(\boldsymbol{\alpha})\}^b_{j\in\mathcal{G}}$ ack into RAM.

In the case of cubic B-spline based deformations (i.e., with q =3 in (2.25)), we have for each $C \in \{X, Y, Z\}$ and $(m', n', p') \in \mathcal{G}$

$$[\boldsymbol{B}_C\boldsymbol{\alpha}]_{m'n'p'} = \sum_{m=0}^{N_X/\Delta_X} \sum_{n=0}^{N_Y/\Delta_Y} \sum_{p=0}^{N_Z/\Delta_Z} \alpha_{Cmnp}\beta_3\left(\frac{m'}{\Delta_X} - m\right)\beta_3\left(\frac{n'}{\Delta_Y} - n\right)\beta_3\left(\frac{p'}{\Delta_Z} - p\right) \tag{5.7}$$

One sees that these values are downsamples of a discrete convolution. The summations in (5.7), like those in (5.2), can therefore be done in separable passes, although this time over the entire array α_{Cmnp}, rather than in $4 \times 4 \times 4$ blocks. In the course of these passes, the array size changes and this has a bearing on the amount of computation. When done first

in m, then in n, then in p, the total number of multiplications is

$$\begin{aligned}
&\text{No. of multiplications} \\
&\simeq (4\Delta_X)\frac{N_X}{\Delta_X}\frac{N_Y}{\Delta_Y}\frac{N_Z}{\Delta_Z} + (4\Delta_Y)N_X\frac{N_Y}{\Delta_Y}\frac{N_Z}{\Delta_Z} + +(4\Delta_Z)N_XN_Y\frac{N_Z}{\Delta_Z} \\
&= 4|\mathcal{G}|(1+\frac{1}{\Delta_Z}+\frac{1}{\Delta_Y\Delta_Z}) \qquad (5.8) \\
&\leq 12|\mathcal{G}| \qquad (5.9)
\end{aligned}$$

Summing this over $C \in \{X, Y, Z\}$, we see from (5.8) and (5.9) that the number of multiplications can range between 12-36 per grid point. However, typical choices of control point grid spacing parameters Δ_C, $C \in \{X, Y, Z\}$, would be large enough so that the actual value, given by (5.8), will tend toward the lower end of the range.

These calculations are also summarized in Table 5.1, again with the analogous case in 2D. Another aspect of the deformation map computations, also summarized in Table 5.1, is that no on-the-fly interpolation weight computation is required. The weights used in each separable pass in Equation (5.7), e.g., $\beta_3\left(\frac{m'}{\Delta_X} - m\right)$ are easily pre-computed and stored.

In a similar way, one can also count the number of memory transfers in the sequence of passes over α_{Cmnp}. The tally below includes transfers both in and out of RAM,

$$\begin{aligned}
&\text{No. of memory transfers} \\
&= |\mathcal{G}|(1+\frac{2}{\Delta_Z}+\frac{2}{\Delta_Y\Delta_Z}+\frac{1}{\Delta_X\Delta_Y\Delta_Z}) \qquad (5.10) \\
&\leq 6|\mathcal{G}|. \qquad (5.11)
\end{aligned}$$

and summing over $C \in \{X, Y, Z\}$ leads to a range of $3 - 18$ memory transfers per grid point. As before, for typical Δ_C, the actual number would tend toward the lower end of this range.

In the tables, one observes that, even near the worst case bounds, the implied number of multiplications and memory transfers is significantly smaller than that required by $\mathcal{W}^{\boldsymbol{u}}_{www}(\cdot)$. Interestingly, the worst case bounds are also independent of the spacings Δ_C, and hence also the number of unknowns α_{Cmnp}. This is due, in part, to the fact that, when one reduces the control grid spacing Δ_C, the support of the basis functions reduces in the same proportion. One can see this mathematically in (2.25) and graphically in Figure 2.2. The computational demands of $\boldsymbol{B}$ become even less weighty in algorithms that use line search operations, as discussed earlier. Since the deformation cost functions of the type in (5.3) depend on $\boldsymbol{\alpha}$ entirely through $\boldsymbol{B\alpha}$, we therefore expect overall that the computation, per iteration, of cost minimizations will be rather insensitive to the dimension of $\boldsymbol{\alpha}$.

Table 5.1: A tally of the number of multiplications per image grid point in image deformation operations.

	2D			3D		
	$\mathcal{W}^{\boldsymbol{u}}_{\beta_3\beta_3\beta_3}(\cdot)$	affine $\boldsymbol{B}$	β_3-based $\boldsymbol{B}$	$\mathcal{W}^{\boldsymbol{u}}_{\beta_3\beta_3}(\cdot)$	affine $\boldsymbol{B}$	β_3-based $\boldsymbol{B}$
weight computations	32	0	0	48	0	0
interpolation	20	6	8-16	84	12	12-36
TOTAL	52	6	8-16	132	12	12-36

Table 5.2: A tally of the number of memory transfers per image grid point in image deformation operations.

	2D			3D		
	$\mathcal{W}^{\boldsymbol{u}}_{\beta_3\beta_3\beta_3}(\cdot)$	affine $\boldsymbol{B}$	β_3-based $\boldsymbol{B}$	$\mathcal{W}^{\boldsymbol{u}}_{\beta_3\beta_3}(\cdot)$	affine $\boldsymbol{B}$	β_3-based $\boldsymbol{B}$
memory transfers	17	2	2-8	65	3	3-18

5.3 Acceleration Techniques

In this section, we describe two techniques for accelerating interpolation. One involves using block alternation to make certain intermediate computed quantities reusable. The second eliminates one of of the separable interpolation steps in (5.5) with approximate, pre-tabulated values. For ease of presentation, we shall describe the techniques in a 2D setting. That is, we will be discussing how to accelerate the evaluation of $\mathcal{W}^{\boldsymbol{u}}_{ww}(\cdot)$, where

again $w = \beta_3$. The same ideas carry over to 3D.

5.3.1 State Variable Hold-Over

In this section, we discuss an acceleration technique that we call *state variable hold-over*. The technique is applicable to algorithms that alternatingly update $\boldsymbol{\alpha}_X$ and $\boldsymbol{\alpha}_Y$ as follows:

Algorithm 5.3.1 (Skeleton of a block alternating algorithm) *Initialization.* Choose a number of iterations $L > 0$ and sub-iteration parameters $L_X, L_Y > 0$.

For $i = 0, \ldots, L$

For $k := 0, \ldots, L_X - 1$

Update $\boldsymbol{\alpha}_X$ only: $(\boldsymbol{\alpha}_X^{\text{old}}, \boldsymbol{\alpha}_Y^{\text{old}}) \mapsto (\boldsymbol{\alpha}_X^{\text{new}}, \boldsymbol{\alpha}_Y^{\text{old}})$

end

For $k := 0, \ldots, L_Y - 1$

Update $\boldsymbol{\alpha}_Y$ only: $(\boldsymbol{\alpha}_X^{\text{old}}, \boldsymbol{\alpha}_Y^{\text{old}}) \mapsto (\boldsymbol{\alpha}_X^{\text{old}}, \boldsymbol{\alpha}_Y^{\text{new}})$

end

end

When certain $\boldsymbol{\alpha}_C$ are held fixed, quantities that are functions of the fixed variables can be held in memory and re-used each time the non-fixed variables are updated. Thus, additional, consecutive updates of the non-fixed variable are more cheaply obtained than if all quantities were freshly computed. The fixed quantities include, for example, the deformation maps associated with the fixed $\boldsymbol{\alpha}_C$. It also includes the intensity interpolation weights associated with the fixed $\boldsymbol{\alpha}_C$. These concepts are illustrated in Figure 5.1 for the case where $\boldsymbol{\alpha}_X$ is the fixed variable. There, we see that, when $\boldsymbol{\alpha}_Y$ is updated, the x-coordinate of $\boldsymbol{d}_j^{\text{new}}$ is the same as of $\boldsymbol{d}_j^{\text{old}}$. Thus, the local interpolation rectangle has

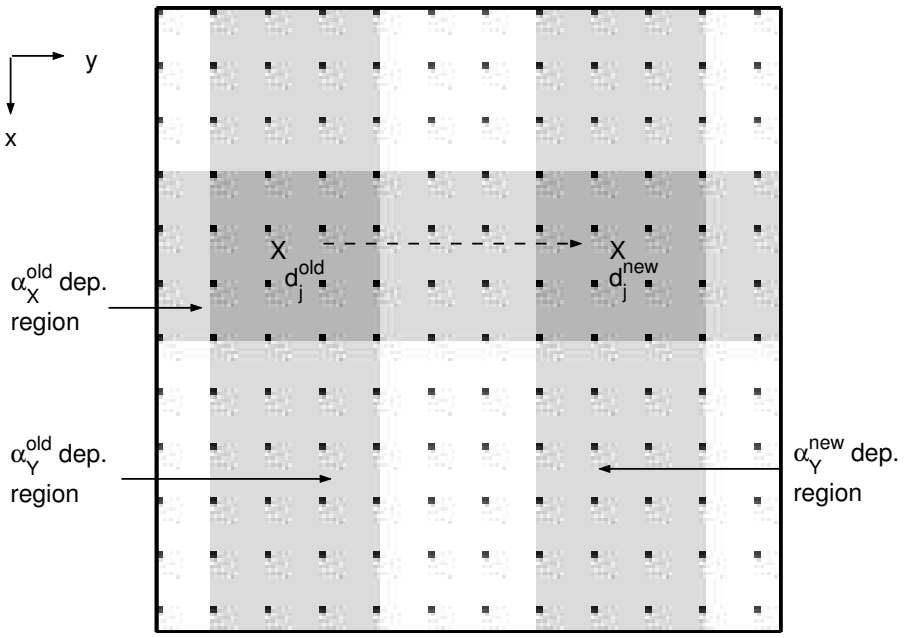

Figure 5.1: Depiction of how interpolation neighborhoods shift during block alternation state variable holdover. There is no shift in the direction corresponding to the fixed variables. Therefore, no new interpolation weights are needed.

the same x-coordinates as for $\boldsymbol{d}_j^{\text{old}}$. Locating the new interpolation rectangle only requires computing the new y-coordinates. Similarly, the x-direction interpolation weights applied to the new rectangle is the same as for the old. Hence, only the y-direction weights need to be recomputed.

5.3.2 Table-Lookup Elimination

In this section, we propose a table-lookup technique that eliminates the need to interpolate along at least one axis of the intensity coefficient array u_{mn}. The idea is to pre-interpolate along that axis at a small sampling interval, compared to the original grid spacing of $\boldsymbol{u}$, and tabulate the results. During the iterations of the minimization algorithm, interpolation along this axis may then be replaced with nearest-neighbor table-lookup. Figure 5.2 illustrates this for when the x-axis (but not the y-axis) is finely sampled. As shown in the figure, each region of interpolated grid values reduces from a 4×4 region to a 4×1 region. This means half as many interpolation weight computations, 4 times fewer

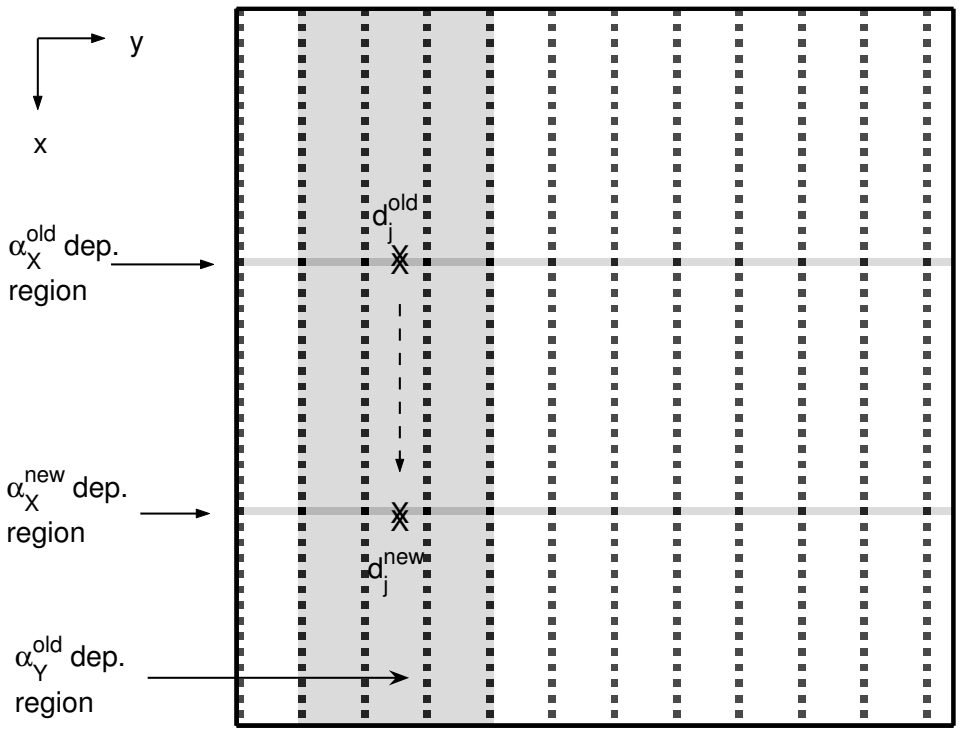

Figure 5.2: Illustration of table-lookup elimination.

interpolation weight multiplications, and 4 times fewer memory transfers of the u_{mn} array are required.

The table-assisted interpolations in Figure 5.2 approximate each $[\mathcal{W}_{ww}^{\boldsymbol{u}}(\boldsymbol{\nu}+\boldsymbol{B}\boldsymbol{\alpha})]_j$ as follows,

$$[\mathcal{W}_{ww}^{\boldsymbol{u}}(\boldsymbol{\nu}+\boldsymbol{B}\boldsymbol{\alpha})]_j \approx \sum_{n=1}^{N_Y} w(y-n) \left[\sum_{m=1}^{N_X} w(x-m) u_{mn} \right] \Bigg|_{\substack{x=\langle d_{jX}(\boldsymbol{\alpha}_X)\rangle \\ y=d_{jY}(\boldsymbol{\alpha}_Y)}} . \tag{5.12}$$

Here $\langle d_{jX} \rangle$ is d_{jX} rounded to the nearest in a set of finely spaced x-values $\{x_{\ell s}\}$ where

$$x_{\ell s} = \ell + s/S, \qquad 0 \le \ell \le N_X, 0 \le s \le S-1. \tag{5.13}$$

The parameter $S \in \mathbb{N}$ is the upsampling ratio.

The table-lookup technique assumes that the quantity $\sum_{m=1}^{N_X} w(x-m)u_{mn}$ has been pre-tabulated at all $x_{\ell s}$. The number of the elements in the table will be $S|\mathcal{G}|$, i.e., it is equivalent to storing S images. Our conjecture is that, for many applications, $S = 5$ or $S =$

10 would be an effective trade off between sampling accuracy and memory consumption. If sub-pixel accuracy by more than a factor of 10 is desired, one could at least exploit this table-assisted scheme in the preliminary stages of a multi-resolution registration.

5.3.3 Combining Table-Lookup and State Variable Hold-Over

If one can afford the extra memory cost, then on-the-fly interpolation weight computation can be almost completely eliminated by combining table-lookup and state variable hold-over. To do so, one pre-tabulates interpolation values along each Cartesian axis of u_{mn}, as described in Section 5.3.2. Suppose, as in Figure 5.2, that $\boldsymbol{\alpha}_X$, but not $\boldsymbol{\alpha}_Y$, was being updated in the course of the block alternations of Algorithm 5.3.1. If state variable hold-over is used, the weights for the y-direction interpolation (which is the only direction in which interpolation occurs) are held in memory throughout the consecutive updates of $\boldsymbol{\alpha}_X$. Thus, updating $\mathcal{W}^{\boldsymbol{u}}_{ww}(\boldsymbol{\nu} + \boldsymbol{B}\boldsymbol{\alpha})$ simply involves multiplying the values in the new interpolation rectangle by these already available weights and summing. No new weight calculations are required. One of course needs an additional table to cover the case when the roles of $\boldsymbol{\alpha}_X$ and $\boldsymbol{\alpha}_Y$ are reversed. That is, $2|\mathcal{G}|S$ pre-computed values must be stored, whereas before only $|\mathcal{G}|S$ elements were stored.

5.4 Computational Tests

5.4.1 Sum of Squared Differences in 2D

Here, we apply the acceleration techniques to the problem of the registration of two 2D images, according to the SSD criterion, with a low complexity deformation map matrix $\boldsymbol{B}$. As discussed in Section 5.1, this is the case where interpolation operations are the most intensive among the computations involved. We can therefore get an idea of the maximum effect of the techniques that we have proposed. These experiments would also be relevant to the case where line searches are used, even for a more complicated $\boldsymbol{B}$. As

is also discussed in Section 6.1, the overhead associated with $\boldsymbol{B}$ becomes marginalized by the repeated cost evaluations required by the line search.

Accordingly, we set out to register the 128×128 torso image shown in Figure 5.3(a) with the image shown in Figure 5.3(b). The absolute difference between these images appears in Figure 5.3(d). The array of pixel values in Figure 5.3(a) played the role of the source image data, $\boldsymbol{u}$. By applying a certain warping $\boldsymbol{W}(\boldsymbol{\alpha}^{\text{true}})\boldsymbol{u}$ to $\boldsymbol{u}$, we generated the target image data vector $\boldsymbol{f}^{\text{targ}}$ which is the image displayed in Figure 5.3(b).

For deformation basis functions, we used 2D tensor products of cubic B-splines, of width 4, but irregularly and densely positioned. The basis functions $\{b_{Xk}(x,y)\}_{k=1}^{120}$ were each centered at a pixel in the shaded region about the left lung boundary in Figure 5.3(c). Similarly, deformation basis functions $\{b_{Yk}(x,y)\}_{k=1}^{283}$ were situated at pixels in the shaded region by the right lung. The true parameter values $\alpha_{Xk}^{\text{true}}$ and $\boldsymbol{\alpha}_{Yk}^{\text{true}}$ were all equal to -10 pixel lengths.

To minimize Φ_{SSD}, we employed the following special case of Algorithm 5.3.1:

Algorithm 5.4.1 (A Block-Alternating Newton-like Method) For $i = 0, \ldots, L$

For $C \in \{X, Y\}$

For $k = 0, \ldots, L_C - 1$

$$\begin{aligned} \boldsymbol{\mathcal{H}} &:= \boldsymbol{B}_C^T \boldsymbol{H}_{CC}(\boldsymbol{\alpha}_X, \boldsymbol{\alpha}_Y) \boldsymbol{B}_C \\ \boldsymbol{g} &:= \boldsymbol{B}_C^T \boldsymbol{g}_C(\boldsymbol{\alpha}_X, \boldsymbol{\alpha}_Y) \\ \boldsymbol{\alpha}_C &:= \boldsymbol{\alpha}_C - \{\boldsymbol{\mathcal{H}}\}_+^{-1} \boldsymbol{g} \end{aligned} \tag{5.14}$$

end

end

end

In (5.14), the transformation $\{\cdot\}_+$ is some operation that makes $\boldsymbol{\mathcal{H}}$ positive definite. When $\{\cdot\}_+$ is simply the identity mapping, $\{\boldsymbol{\mathcal{H}}\}_+ = \boldsymbol{\mathcal{H}}$, these update equations reduce to Newton steps in $\boldsymbol{\alpha}_X$ and $\boldsymbol{\alpha}_Y$ respectively. However, Φ_{SSD} is non-convex, so $\{\cdot\}_+$ would normally be something non-trivial.

In the present experiment, $\{\boldsymbol{\mathcal{H}}\}_+$ was the transformation that alters the eigendecomposition of $\boldsymbol{\mathcal{H}}$ by replacing each eigenvalue, λ_k, with $\max\{|\lambda_k|, 0.1\}$.[2] This can be viewed as a refinement of the Levenberg-Marquardt method.[3] In the Levenberg-Marquardt method, the Hessian eigenvalues are all increased by a common constant. Conversely, in our method, eigenvalues that are sufficiently large to begin with are left alone, thus avoiding unnecessarily large curvatures and slower convergence.

We implemented Algorithm 5.4.1 using all combinations of the acceleration techniques in Section 5.3. Each implementation was then tested on the source and target image data. In all cases, the iteration/sub-iteration loop sizes were $L = 5$ and $L_X = L_Y = 10$. Also, the components of the initial $\boldsymbol{\alpha}$ were set, in each test, to -5 pixel lengths. The implementations employing the table-lookup technique did so with $S = 10$. Two sets of tables were maintained. One set was derived from pre-interpolation (with w and all relevant derivates) in the x-direction. The second set was similarly derived by pre-interpolating in the y-direction.[4]

In Figures 5.4 and 5.5, the progress of each algorithm implementation is plotted versus CPU time. Figure 5.4 only accounts for the time spent on gradient and Hessian evaluations,

---

[2]The threshold value of 0.1 was empirically chosen. More generally, one desires a constant threshold that is small compared to the Hessian eigenvalues in the neighbourhood of the solution.

[3]The application of Levenberg-Marquardt, without coordinate block alternation, was considered for the least squares registration problem [53, 54].

[4]This allowed us to maximally exploit the combination of state variable hold-over and the table-lookup technique as described in Section 5.3.3. In practice, of course, one may decide to conserve memory by tabulating only 0-th derivatives and only for one Cartesian direction. A variety of memory-saving schemes along these lines are possible.

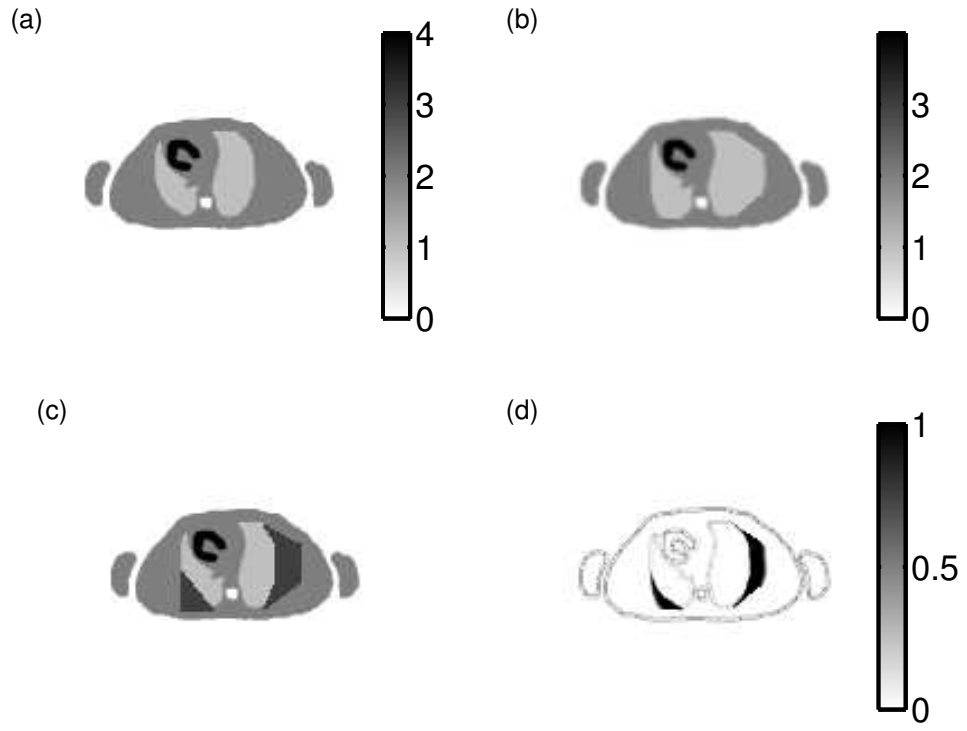

Figure 5.3: **(a)** The source image. **(b)** The target image. **(c)** Locations of the deformation basis functions. A separate 2D cubic B-spline basis function $b_{Ck}(x, y)$ was situated at each pixel in the darkened regions near the lung boundaries. Basis functions $\{b_{Xk}\}_{k=1}^{120}$ were placed in the region at the left lung and $\{b_{Yk}\}_{k=1}^{283}$ were placed in the region at the right lung, for a total of 383 irregularly, but densely positioned control points. **(d)** Absolute difference between the source and target images.

which is where all of the interpolation operations occur. Figure 5.5 plots descent against the total CPU time expended up to that point in the program execution. The latter took into account overhead for generating the lookup tables before the start of the iterations and computing $\{\boldsymbol{\mathcal{H}}\}_{+}^{-1}$. The trials marked 'Plain' refer to the case where none of the acceleration techniques were used, i.e., interpolations were done entirely-on-the fly.

One thing that becomes evident when looking at Figures 5.4 and 5.5 is that the impact of the table-lookup scheme comes primarily from the way it conserves memory transfer operations, as opposed to the number of arithmetic operations. Comparing the rate of descent when table-lookup alone was used to the case when no acceleration method was used, one observes a factor of 4 speed-up. The table-lookup scheme, we know, simplifies

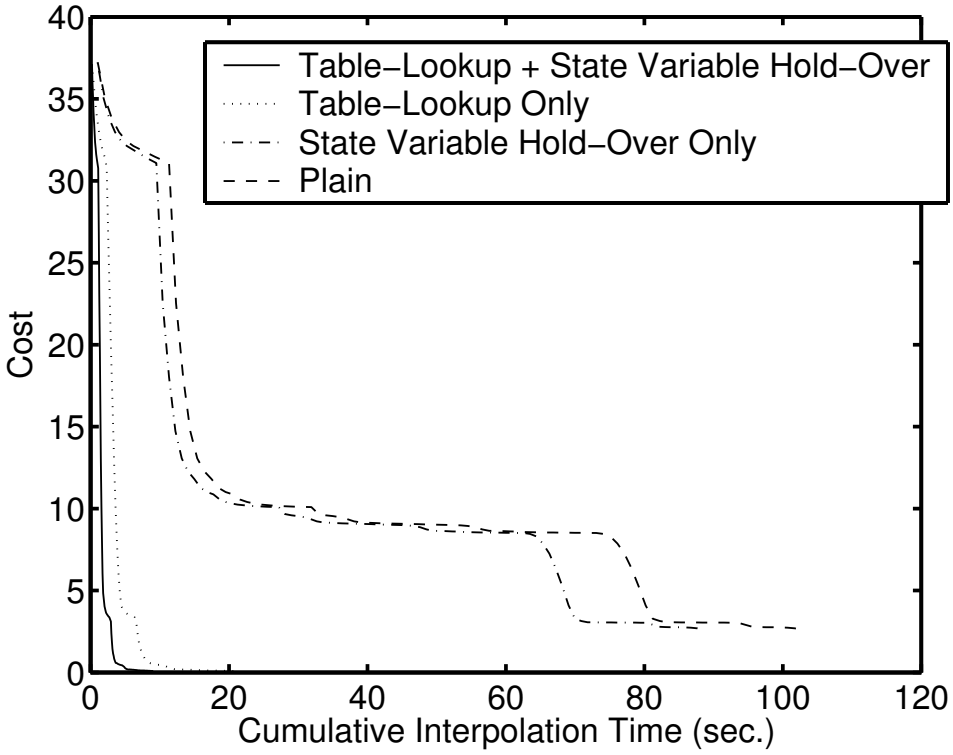

Figure 5.4: The progress of the algorithm implementations versus cumulative time spent on interpolation operations, i.e., on gradient and Hessian computations.

the interpolation steps from the processing of 4×4 regions to the processing of 4×1 regions of u_{mnp}. This of course entails 4 times fewer memory transfers.

The number of multiplications is also reduced, but not by as large a factor. As discussed in Section 5.3.2, half the weight computations are required (16 multiplications) while interpolating the 4×1 region requires 4 multiplications. Thus, the arithmetic involves 20 multiplications in comparison to the (see Table 5.1) 52 multiplications required in the non-accelerated case. Thus, if arithmetic were dominant, we would expect to see a speed-up factor of little more than 2 in Figure 5.4 and perhaps less in Figure 5.5, which accounts for other overhead. Instead, however, we see the same factor of 4 speed-up in both.

Once the table-lookup technique is in play, however, the reduction in the number of operations effected by state variable hold-over seems to have an impact. An additional speed factor of about 2 is observed when state variable hold-over and table-lookup are

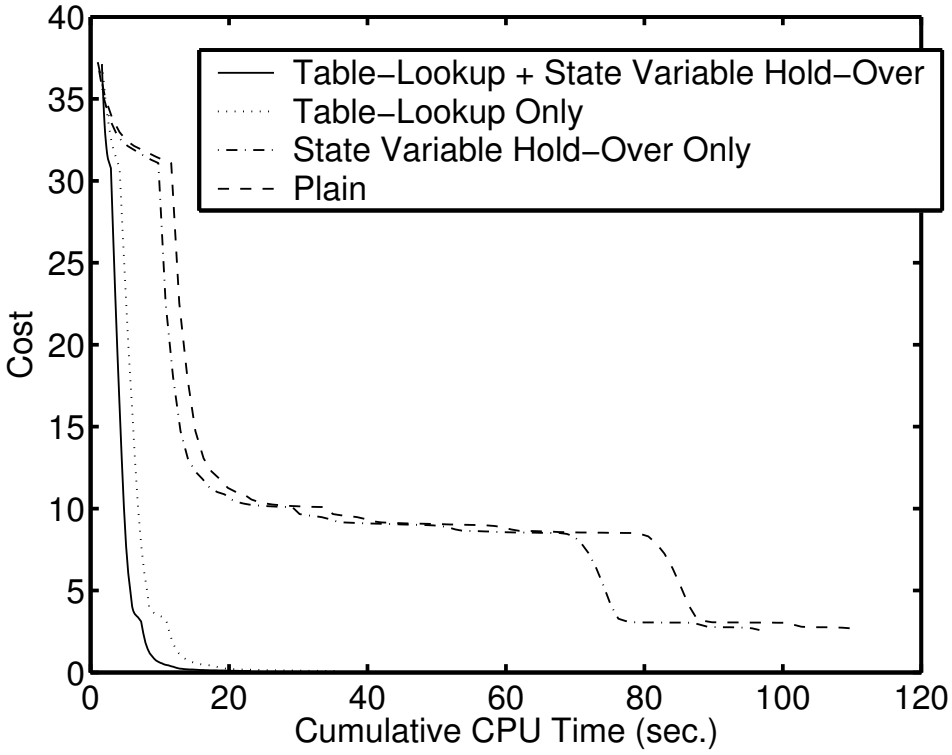

Figure 5.5: The progress of the algorithm implementations versus total CPU time.

combined.

5.4.2 Interpolation Effort in JEDM

In Section 5.4.1, we saw that the table-lookup strategy produced the dominant benefit. We are now interested in seeing to what extent this benefit carries over to relevant minimization tasks in our implementation of JEDM (see Chapter 4, Section 4.4.2). In particular, as described in Section 4.4.2, we employ steepest descent with the Armijo line search rule to minimize Φ_{JEDM} with respect to each $\boldsymbol{\alpha}_t$. It is of interest to know to what extent the table-lookup elimination technique can reduce the repeated cost function evaluations in these line searches. Relative to the SSD tests in Section 5.4.1, this case is complicated due to the presence of forward projection operations. In terms of (5.3), we have a situation where the outer function

$$\boldsymbol{\xi}(\boldsymbol{z}) = \mathrm{KL}(\boldsymbol{y}_t, \tau_t(\boldsymbol{P}\boldsymbol{z} + \boldsymbol{r}_t))$$

has its own significant level of complexity due to the required multiplications with $\boldsymbol{P}$. The overhead due to $\boldsymbol{P}$ might bottleneck the computations and reduce the impact of the accelerated interpolations.

We have conducted timing tests to compare, for various 3D image sizes, the CPU time of the interpolations (with and without table-lookup assistance) to projection time. Because memory transfer issues appeared to dominate in our tests of the previous section, we ran these tests for two cases. In one case, we set the deformations to zero. This forces the interpolation operations to extract series of $4 \times 4 \times 4$ blocks sequentially arranged in memory. The results are shown in Figure 5.6. In the second case, we set the deformation map to a random value, thereby simulating the case where the $4 \times 4 \times 4$ blocks would be accessed in a highly varied order. The results for this latter case are shown in Figure 5.7. In the left panel of each plot, we show the CPU time ranging from very small to PET sized image volumes. To explore an extreme, we have added to the plot, in the right hand panel of each figure, the case of a much larger, comparable to CT image sizes.

The forward projection matrix $\boldsymbol{P}$, was implemented as a sparse matrix using tools from the ASPIRE library, based on a 2D line integral model. The relative CPU times for projection and interpolation operations will obviously be affected by the projection model and the size of the sinogram array used. In our experiments in Chapter 4, we found that accurate motion correction was possible via a reduced number of projection angles. Accordingly, we parallel these conditions in our tests here and used a number of projection angles equal to the length of the image. (In real-life circumstances, one might re-incorporate the full set of projection data once the motion estimation was sufficiently complete.)

In Figure 5.6, one sees that the interpolation times for both implementations scale quite linearly with image volume, even up to the CT-sized case. Moreover, a factor of 4 speed-

up is again observed. In Figure 5.7, it is seen that the randomness in the access order had a significant impact on the total interpolation times in all cases, increasing them by at least a factor of three. The acceleration factor confered by table-lookup was impacted as well, but less so. One sees in the randomized cases that the speed-up factor reduced from 4 to 3.

Regarding the projection times, one sees that the interpolation time is a significant fraction of the projection time in PET sized volumes and even significantly exceeds it in the random access case. Given the plots, one can expect to save about $25 - 50\%$ CPU time on line searches using table look-up assisted interpolation.

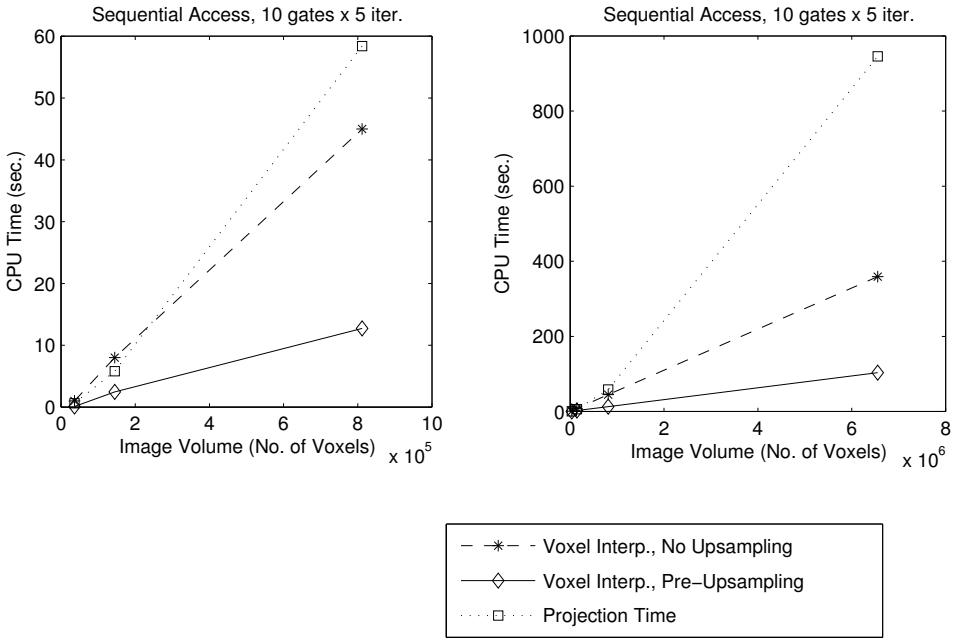

Figure 5.6: Comparing CPU times of accelerated interpolation schemes and forward projection. Here a zero deformation map is used, simulating the case where the grid values u_{mnp} are accessed in the most sequential and favorable order. In the right hand plot, we have added an additional point to reflect an exaggerated, CT sized volume.

5.5 Conclusion

This chapter has discussed acceleration methods applicable to interpolation based transformation models in various image registration problems. The method of dominant effect was a table look-up technique that makes the operations $\boldsymbol{W}(\boldsymbol{\alpha})\boldsymbol{u} = \mathcal{W}^{\boldsymbol{u}}_{www}(\boldsymbol{\nu}+\boldsymbol{B}\boldsymbol{\alpha})$ more

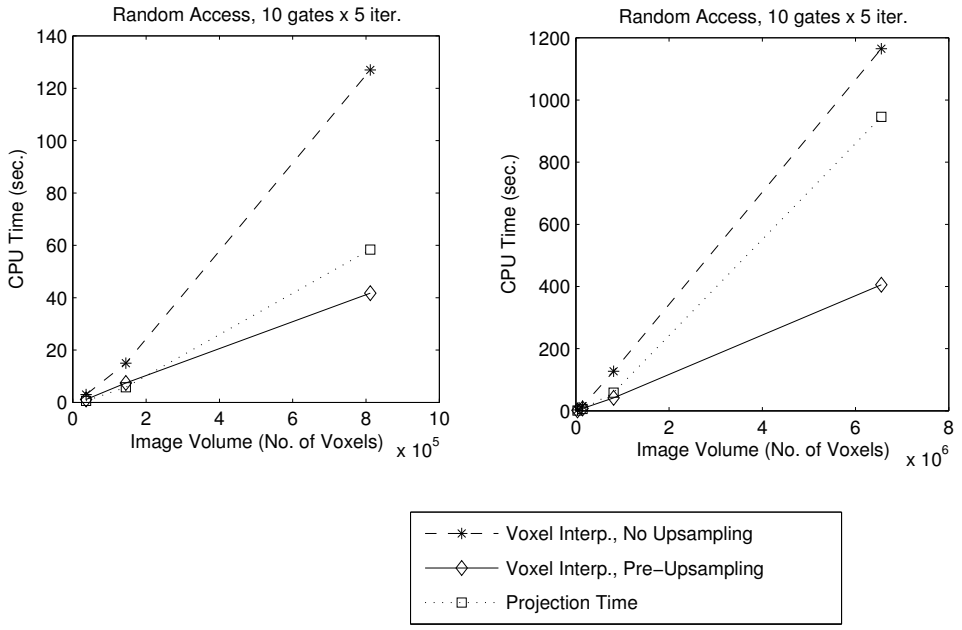

Figure 5.7: Similar to Figure 5.6, except that here a random deformation map is used, simulating the case where the grid values u_{mnp} are accessed in the most variable and least favorable order.

sparse. This reduces the required number of memory transfers resulting in a speed-up by the same factor. The overall interpolation time is greatly impacted by the sequentiallity with which the image data u_{mnp} is accessed, but the acceleration factor is only slightly impacted. This may motivate us to consider, in future work, the idea of pre-sorting the deformed grid values $\boldsymbol{d}_j(\boldsymbol{\alpha})$ before looping through the image.

The impact of the acceleration methods depends on the particular deformation cost function considered and other bottlenecks that may exist. In the case of SSD registration, the interpolations are the dominant computation. We therefore expected the acceleration methods to have a strong impact and this was indeed corroborated in our experiments in Section 5.4.1. In the case of the JEDM cost function, it was found in Section 5.4.2, that the table-lookup scheme could accelerate line search operations to some degree, with certain assumptions on the projection matrix. However, the speed-up is by a more modest fraction than in SSD image domain registration, due to overhead introduced by the forward projection operations. As mentioned in Section 5.1, one way to combat overhead such as

this is to use the MM methodology and replace the given cost function with a simpler one. This is a path we embark upon in Chapter 6.

CHAPTER 6

Extending the Theory of Majorize-Minimize (MM) Algorithms

6.1 Introduction

6.1.1 Scope and Context

This chapter presents our final branch of work, which deals with generalizing the theory of Majorize-Minimize (MM) algorithms. Some background on MM has already been given in Section 2.4. The scope of our treatment here will be the application of MM to minimization problems of the general form,

$$\text{min. } \Phi(\boldsymbol{\theta}) \quad \text{s.t.} \quad \boldsymbol{\theta} \in \Theta, \tag{6.1}$$

where $\Phi(\boldsymbol{\theta}) : \Theta \subset \mathbb{R}^p \rightarrow \mathbb{R}$ is a continuously differentiable (but possibly non-convex) cost function, $\mathbb{R}^p$ is the space of length p column vectors,[1] and Θ is a convex feasible set. In making the analysis that follows, we have tried to anticipate the needs of a range of applications and MM algorithm design ideas.

In relation to the motion-correction application emphasized in this dissertation, we mean this chapter to be viewed as a stepping stone to future work. The generalized MM theory that we present opens up certain algorithm design possibilities for implementing JEDM. We discuss some of them in Section 6.6, and how the analyses we do here sup-

[1] When $\boldsymbol{\theta}$ is a scalar variable, we shall use the notation θ instead.

port them. The hope is that they will lead to better performing alternatives to the ones we currently use to implement JEDM in Chapter 4. For now, though, our intent is to provide a flexible algorithm design framework, one which lets us consider various candidate algorithms and gives us a basis for knowing whether or not they will converge.

6.1.2 Desirable Generalizations of MM

As new minimization problems are encountered, the creativity of MM algorithm designers leads to new kinds of tangent majorants and corresponding algorithms. Because of this trend, it is desirable to have as general a theory as possible regarding the kinds of tangent majorants that one can use and the kinds of cost functions that one can apply them to. The literature complicates this somewhat, because various analyses have been made under a variety of different conditions. To our knowledge, a fairly complete repository of current MM theory would be obtained by consolidating [72] and [32, 33]. Prior to these works, MM algorithms studied in the literature used only a single majorant generator $\phi(\cdot;\cdot)$. Furthermore, convergence analyses mostly treated convergence to interior points of the feasible set only. Those analyses that did consider convergence to the boundary were valid only for specific problems and tangent majorants (e.g., [13, 88]). Beginning with [32, 33], the majorant generators were allowed to be iteration-dependent, resulting in various benefits like block alternation. In [32], only convergence to interior points of the feasible set was considered. The work of [33] extended [32] and tried to address convergence to the boundary when Θ was the non-negative orthant and when Φ was strictly convex. However due to an error (see Remark 6.4.6), the extension is valid only under much more restrictive assumptions than intended. In [72], a very unified treatment of constraints is given, one that covers even non-convex constraints. However, the analysis there does not allow iteration-dependent majorant generators, as in [32, 33].

Although more general than their forerunners, the collective fruits of [32, 33, 72] have lately proven insufficient to cover various contemporary problems and MM algorithms devised for them. One limitation is that [32, 33] do not treat constrained, non-convex problems comprehensively. Constraints are considered in [33], but the analysis contains an error as mentioned above. Furthermore, only the case in which Φ is strictly convex and in which Θ is the non-negative orthant is addressed. These restrictions apply to certain penalized likelihood minimization problems commonly encountered in emission tomography, and it is these problems that the authors of [33] had in mind when they made that analysis. However, more recent work in medical imaging applications has given rise to more complicated cost functions that are not convex, but for which MM algorithms can be derived (e.g., [27, 104, 46, 47]). Moreover, the kinds of constraints now encountered in medical imaging go beyond mere non-negativity constraints. For example, in non-rigid image registration, feasible sets of a more complicated polyhedral form may be desired (see [49, p. 60]) to ensure physically realistic solutions. These more complicated constraints would be covered by [72] so long as iteration-independent majorant generators were used. However, iteration-dependent $\phi^i(\cdot;\cdot)$ are often desirable, at minimum because they allow block alternation, which can simplify algorithm design.

Further limitations of [33] are that the tangent majorants are required to be twice differentiable, strongly convex, and defined throughout the feasible set Θ. In [47], we derived several kinds of convex tangent majorants for a non-convex problem. However, the tangent majorant domains were strict subsets of Θ. Furthermore, many imaging problems involve cost functions with convex, but only once-differentiable penalty terms. This motivates certain MM algorithm designs with only once-differentiable convex tangent majorants.

In this chapter, we generalize the analysis in [33] resulting in a much more versatile algorithm design framework. Our analysis is more general than [33] in several respects.

Firstly, arbitrary convex feasible sets are permitted in our framework. In this way, we marry some of the generality of [72] with that of [33]. Secondly, the tangent majorant domains can be strict subsets of the feasible set. Thirdly, the technique of block alternation is considered for feasible sets of a more general Cartesian product form. Fourth, Φ is not required to be convex and the tangent majorants $\{\phi^i(\cdot;\boldsymbol{\theta}^i)\}$ are often allowed to be non-convex as well. Finally, our analysis does not require the cost function and the tangent majorants to be more than once continuously differentiable. Since we treat only convex feasible sets, the scope of possible constraints is more restrictive than in [72]. However, unlike [72], the use of iteration-dependent tangent majorants is covered in the presence of constraints (and hence also, the error in [33] is remedied).

6.1.3 Chapter Organization

The rest of the chapter is organized as follows. In Section 6.2, we formalize the class of MM algorithms considered in this chapter. Next, in Section 6.3, we give a few additional mathematical preliminaries and describe various conditions imposed in the subsequent analysis. In Section 6.4, we analyze the asymptotic behavior of MM. Results are developed showing stationarity of MM limit points in both the block alternating and non-block alternating case. In each case, two sets of conditions are applied. One set involves traditional kinds of continuity assumptions on the majorant generators. None of these conditions are more restrictive than [33]. The second set involves local curvature bounds on the tangent majorants. We then deduce convergence of MM in norm (see Theorem 6.4.5) in a standard way by imposing discreteness assumptions on the set of stationary points of (6.1). Non-isolated stationary points are not generally stable (cf. [7, p. 22]) under perturbations of Φ. Therefore, whether or not an algorithm converges in norm to such points seems mainly a question of theoretical interest. It is for such reasons that algorithm users

often settle for algorithms with stationary limit points. Nevertheless, we have done some work on convergence to non-isolated stationary points, which the interested reader can find in Section 6.7.

When Φ is non-convex, local behavior of MM becomes important and is the subject of Section 6.5. Here we restrict our attention to the case where the tangent majorants have path-connected sublevel sets (e.g., as in the case when the tangent majorants are convex). For this family of tangent majorants, it is shown that the iterates $\{\boldsymbol{\theta}^i\}$ are captured by local basin-like regions in the graph of Φ. This property allows us to derive a local analogue, namely Theorem 6.5.6, to the convergence described in Theorem 6.4.5. An implication of Theorem 6.5.6 is that local convergence will occur over a larger neighborhood of a global minimizer than can typically be guaranteed with more standard algorithms. In addition, various non-convex minimization strategies involve basin-probing steps. The basin capture property of connected tangent majorants makes MM algorithms particularly suitable for implementing these steps.

In Section 6.6, we discuss the relevance of our results to the JEDM problem, in particular, how one might devise tangent majorants to accomplish the cost reductions (4.22). Section 6.7 considers the question of whether convergence in norm will occur to stationary points that are *not* isolated. This question might be of theoretical interest only, because of stability issues alluded to above. However, various MM algorithms ([13, 88, 64]) have been observed to converge in norm, even to non-isolated minima, so specialists are apt to wonder if this behavior can be proved more generally. For single variable problems, we show, in Theorem 6.7.2, that, if the iterate sequence $\{\theta^i\}$ is bounded and the $\{\phi^i(\cdot;\theta^i)\}$ have a uniform lower curvature bound, then convergence is assured, regardless of whether or not continua of stationary points exist. Moreover, we argue (see Example 6.7.1) that these conditions are about as weak as one can consider. For multi-variable problems, we

find that these conditions are insufficient for convergence. This is demonstrated via an example in $\mathbb{R}^2$ (see Example 6.7.5). At this time, we are unable to extend Theorem 6.7.2 to the multi-variable case. However, the aforementioned 2D example provides some intuition as to what conditions may be sufficient, and so is a useful starting point for future work.

6.2 Mathematical Description of MM Algorithms

In this section, we describe the class of MM algorithms considered in this chapter. With no loss of generality, we assume that the feasible set Θ is a Cartesian product of $M \leq p$ convex sets, i.e.,

$$\Theta = \Theta_1 \times \Theta_2 \times \ldots \times \Theta_M, \tag{6.2}$$

where $\Theta_m \subset \mathbb{R}^{p_m}$, $m = 1, \ldots, M$ and $\sum_{m=1}^{M} p_m = p$. Since Θ is assumed convex, such a representation is always possible with $M = 1$.

To facilitate discussion, we first introduce some indexing conventions. Given $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p) \in \Theta$, we can represent $\boldsymbol{\theta}$ as a concatenation of vector partitions $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_M)$ where $\boldsymbol{\theta}_m \in \Theta_m$, $m = 1, \ldots, M$. If $\mathcal{S} = \{m_1, m_2, \ldots, m_q\}$ is a subset of $\{1, \ldots, M\}$, then we write

$$\boldsymbol{\theta}_{\mathcal{S}} = (\boldsymbol{\theta}_{m_1}, \boldsymbol{\theta}_{m_2}, \ldots, \boldsymbol{\theta}_{m_q})$$

$$\Theta_{\mathcal{S}} = \Theta_{m_1} \times \Theta_{m_2} \times \ldots \times \Theta_{m_q}$$

$$\mathbb{R}_{\mathcal{S}} = \mathbb{R}^{p_{m_1} + p_{m_2} + \ldots + p_{m_q}}$$

to indicate certain Cartesian sub-products and their elements. Thus, one can write $\boldsymbol{\theta}_{\mathcal{S}} \in \Theta_{\mathcal{S}} \subset \mathbb{R}_{\mathcal{S}}$. The complement of $\mathcal{S}$ shall be denoted $\tilde{\mathcal{S}}$. We may also represent a given $\boldsymbol{\theta} \in \Theta$ in the partitioned form $\boldsymbol{\theta} = (\boldsymbol{\theta}_{\mathcal{S}}, \boldsymbol{\theta}_{\tilde{\mathcal{S}}})$, and $\Phi(\boldsymbol{\theta})$ may be equivalently written $\Phi(\boldsymbol{\theta}_{\mathcal{S}}, \boldsymbol{\theta}_{\tilde{\mathcal{S}}})$.

The following example illustrates these indexing conventions and gives the flavor of the problems that this framework is meant to accommodate.

Example 6.2.1 Consider the following minimization problem with $\Theta_1 \subset \mathbb{R}^3$, $\Theta_2 \subset \mathbb{R}$, and $\Theta_3 \subset \mathbb{R}^2$ specified by the constraints as indicated.

$$\text{min.} \quad \Phi(\theta_1, \ldots, \theta_6) = \left\{ \sum_{j=1}^{6} \theta_j \right\} - \log \left\{ \sum_{j=1}^{6} \theta_j \right\}$$

subject to

$$\begin{aligned}
&\Theta_1 \left\{ \begin{array}{l} \theta_1,\, \theta_2,\, \theta_3 \geq 0 \\ \theta_1 + \theta_2 + \theta_3 = 10 \\ \theta_1 + 2\theta_2 = 5 \end{array} \right. \\
&\Theta_2 \left\{ \begin{array}{l} 1 \leq \theta_4 \leq 6 \end{array} \right. \\
&\Theta_3 \left\{ \begin{array}{l} \theta_5^2 + \theta_6^2 \leq 9. \end{array} \right.
\end{aligned}$$

Thus, we obtain the decomposition $\Theta = \Theta_1 \times \Theta_2 \times \Theta_3$, a particular case of (6.2) with $p = 6$, $M = 3$, $p_1 = 3$, $p_2 = 1$, and $p_3 = 2$. Given $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_6)$ then, according to our indexing conventions, $\boldsymbol{\theta}_1 = (\theta_1, \theta_2, \theta_3)$, $\boldsymbol{\theta}_2 = (\theta_4)$, and $\boldsymbol{\theta}_3 = (\theta_5, \theta_6)$. If, for example, we let $\mathcal{S} = \{1, 3\}$, then $\boldsymbol{\theta}_{\mathcal{S}} = (\theta_1, \theta_2, \theta_3, \theta_5, \theta_6)$, $\Theta_{\mathcal{S}} = \Theta_1 \times \Theta_3$, and $\mathbb{R}_{\mathcal{S}} = \mathbb{R}^5$. Also, $\tilde{\mathcal{S}} = \{2\}$, $\boldsymbol{\theta}_{\tilde{\mathcal{S}}} = \theta_4$, $\Theta_{\tilde{\mathcal{S}}} = \Theta_2$, and $\mathbb{R}_{\tilde{\mathcal{S}}} = \mathbb{R}$. Observe, as in the case of Θ_1 above, that any Θ_m can have an empty interior in its corresponding space $\mathbb{R}^{p_m}$. That is, we are not assuming that the Θ_m are *solid* subsets of $\mathbb{R}^{p_m}$.

Given an index set $\mathcal{S} \subset \{1, \ldots, M\}$ and a point-to-set mapping $D(\cdot)$ such that $\bar{\boldsymbol{\theta}}_{\mathcal{S}} \in D(\bar{\boldsymbol{\theta}}) \subset \Theta_{\mathcal{S}}$ for all $\bar{\boldsymbol{\theta}} \in \Theta$, we define a *majorant generator* $\phi(\cdot;\cdot)$ as a function mapping each $\bar{\boldsymbol{\theta}} \in \Theta$ to what we call a *tangent majorant*, a function $\phi(\cdot; \bar{\boldsymbol{\theta}}) : D(\bar{\boldsymbol{\theta}}) \subset \Theta_{\mathcal{S}} \to \mathbb{R}$ satisfying

$$\Phi(\boldsymbol{\xi}, \bar{\boldsymbol{\theta}}_{\tilde{\mathcal{S}}}) - \Phi(\bar{\boldsymbol{\theta}}) \leq \phi(\boldsymbol{\xi}; \bar{\boldsymbol{\theta}}) - \phi(\bar{\boldsymbol{\theta}}_{\mathcal{S}}; \bar{\boldsymbol{\theta}}) \qquad \forall \boldsymbol{\xi} \in D(\bar{\boldsymbol{\theta}}). \tag{6.3}$$

We call $\bar{\boldsymbol{\theta}}$ the *expansion point* of the tangent majorant. Given the point-to-set mapping $D(\cdot)$, we can also write $\phi(\cdot;\cdot) : \mathcal{D} \to \mathbb{R}$, in which

$$\mathcal{D} = \left\{(\boldsymbol{\xi};\bar{\boldsymbol{\theta}}) \,:\, \boldsymbol{\xi} \in D(\bar{\boldsymbol{\theta}}) \subset \Theta_{\mathcal{S}},\, \bar{\boldsymbol{\theta}} \in \Theta\right\}$$

denotes the domain of the majorant generator. An equivalent way of expressing (6.3) is

$$\min_{\boldsymbol{\xi} \in D(\bar{\boldsymbol{\theta}})} \{\phi(\boldsymbol{\xi};\bar{\boldsymbol{\theta}}) - \Phi(\boldsymbol{\xi},\bar{\boldsymbol{\theta}}_{\tilde{\mathcal{S}}})\} = \phi(\bar{\boldsymbol{\theta}}_{\mathcal{S}};\bar{\boldsymbol{\theta}}) - \Phi(\bar{\boldsymbol{\theta}}). \tag{6.4}$$

Certain properties of tangent majorants (see, for example, Note A.2) are more obvious from this definition.

To design an *MM algorithm*, one selects an initial point $\boldsymbol{\theta}^0 \in \Theta$, a sequence of index sets $\{\mathcal{S}^i\}_{i=0}^{\infty}$, and a sequence of majorant generators $\{\phi^i(\cdot;\cdot) : \mathcal{D}^i \to \mathbb{R}\}_{i=0}^{\infty}$ with domains

$$\mathcal{D}^i = \left\{(\boldsymbol{\xi};\bar{\boldsymbol{\theta}}) \,:\, \boldsymbol{\xi} \in D^i(\bar{\boldsymbol{\theta}}) \subset \Theta_{\mathcal{S}^i},\, \bar{\boldsymbol{\theta}} \in \Theta\right\}.$$

where the $D^i(\cdot) \subset \Theta_{\mathcal{S}^i}$ are point-to-set mappings, each satisfying $\bar{\boldsymbol{\theta}}_{\mathcal{S}^i} \in D^i(\bar{\boldsymbol{\theta}})$ for all $\bar{\boldsymbol{\theta}} \in \Theta$. The simplest case is when $D^i(\bar{\boldsymbol{\theta}}) = \Theta_{\mathcal{S}^i}$ and $\mathcal{D}^i = \Theta_{\mathcal{S}^i} \times \Theta$ for all i. This was the assumption made in [33]. This assumption does not hold, however, for the MM algorithms in [21, 47]. Once the majorant generators are chosen, the MM algorithm is implemented by generating an iteration sequence $\left\{\boldsymbol{\theta}^i \in \Theta\right\}_{i=0}^{\infty}$ satisfying,

$$\boldsymbol{\theta}^{i+1}_{\mathcal{S}^i} \in \operatorname*{argmin}_{\boldsymbol{\xi} \in D^i(\boldsymbol{\theta}^i)} \phi^i(\boldsymbol{\xi};\boldsymbol{\theta}^i) \tag{6.5}$$

$$\boldsymbol{\theta}^{i+1}_{\tilde{\mathcal{S}}^i} = \boldsymbol{\theta}^i_{\tilde{\mathcal{S}}^i}. \tag{6.6}$$

Here, we assume that the set of minimizers in (6.5) is non-empty. We shall refer to the total sequence $\left\{\boldsymbol{\theta}^i\right\}_{i=0}^{\infty}$ produced this way as an *MM sequence*. In the simplest case, in which one chooses $\phi^i(\boldsymbol{\theta}_{\mathcal{S}^i};\bar{\boldsymbol{\theta}}) = \Phi(\boldsymbol{\theta}_{\mathcal{S}^i},\bar{\boldsymbol{\theta}}_{\tilde{\mathcal{S}}^i})$ for all i, (6.5) and (6.6) become a generalization of block coordinate descent (e.g., [7, p. 267]), in which the coordinate blocks are not necessarily disjoint. By virtue of (6.3) and (6.5), $\{\Phi(\boldsymbol{\theta}^i)\}$ is monotonically non-increasing.

A tangent majorant is a mild generalization of what we call a *true tangent majorant*. A function $\phi(\cdot;\bar{\boldsymbol{\theta}})$ satisfying (6.3) is a true tangent majorant if it also satisfies

$$\phi(\boldsymbol{\xi};\bar{\boldsymbol{\theta}}) \geq \Phi(\boldsymbol{\xi},\bar{\boldsymbol{\theta}}_{\bar{\mathcal{S}}}) \qquad \forall \boldsymbol{\xi} \in D(\bar{\boldsymbol{\theta}}), \tag{6.7}$$

$$\phi(\bar{\boldsymbol{\theta}}_{\mathcal{S}};\bar{\boldsymbol{\theta}}) = \Phi(\bar{\boldsymbol{\theta}}). \tag{6.8}$$

That is, $\phi(\cdot;\bar{\boldsymbol{\theta}})$ majorizes $\Phi(\cdot,\bar{\boldsymbol{\theta}}_{\bar{\mathcal{S}}})$ over $D(\bar{\boldsymbol{\theta}})$ and is tangent to it in the sense that equality holds[2] at $\bar{\boldsymbol{\theta}}_{\mathcal{S}}$. These considerations motivate our choice of the term *tangent majorant*.[3] We shall refer to a majorant generator that produces true tangent majorants as a true majorant generator . Also, we abbreviate (6.7) and (6.8) via the notation,

$$\phi(\cdot;\bar{\boldsymbol{\theta}}) \underset{D(\bar{\boldsymbol{\theta}})}{\overset{\bar{\boldsymbol{\theta}}}{\succ}} \Phi(\cdot,\bar{\boldsymbol{\theta}}). \tag{6.9}$$

The relational operator $\underset{D(\bar{\boldsymbol{\theta}})}{\overset{\bar{\boldsymbol{\theta}}}{\succ}}$ describes a partial ordering between functions on $D(\bar{\boldsymbol{\theta}})$. Any tangent majorant can be made into a true tangent majorant by adding to it an appropriate global constant. Doing so does not influence the update formulae (6.5) and (6.6). The distinction between tangent majorants and true tangent majorants is therefore irrelevant in studying MM sequences. The distinction becomes important, however, when deriving tangent majorants by composition of functions (see Note A.1).

When the sets $\mathcal{S}^i$ vary non-trivially with the iteration number i, we say that the algorithm is *block alternating* (cf. [32, 33]). Conversely, if all $\mathcal{S}^i = \{1,\ldots,M\}$, then $\Theta_{\mathcal{S}^i} = \Theta$ for all i, and we say that the algorithm is not block alternating (or, that the updates are *simultaneous*). In the latter case, (6.3) simplifies to

$$\Phi(\boldsymbol{\xi}) - \Phi(\bar{\boldsymbol{\theta}}) \leq \phi(\boldsymbol{\xi};\bar{\boldsymbol{\theta}}) - \phi(\bar{\boldsymbol{\theta}};\bar{\boldsymbol{\theta}}) \qquad \forall \boldsymbol{\xi} \in D(\bar{\boldsymbol{\theta}}), \tag{6.10}$$

---

[2]It is also tangent to it in the sense that the directional derivatives of $\phi(\cdot;\bar{\boldsymbol{\theta}})$ and $\Phi(\cdot,\bar{\boldsymbol{\theta}}_{\bar{\mathcal{S}}})$ match at $\bar{\boldsymbol{\theta}}_{\mathcal{S}}$ except in special circumstances (see Note A.2).

[3]In some literature, the term *surrogate* has been used, however much more general use of this term has been used in other works. We feel that the term *tangent majorant* is much more descriptive of the kind of surrogate functions used in MM specifically.

while (6.5) and (6.6) reduce to

$$\boldsymbol{\theta}^{i+1} \in \operatorname*{argmin}_{\boldsymbol{\theta} \in D^i(\boldsymbol{\theta}^i)} \phi^i(\boldsymbol{\theta}; \boldsymbol{\theta}^i), \tag{6.11}$$

The technique of block alternation can be advantageous because it can be simpler to derive and minimize tangent majorants satisfying (6.3), which involve functions of fewer variables, than tangent majorants satisfying (6.10). Block alternation can also provide faster alternatives to certain non-block alternating algorithm designs [32]. To apply block alternation meaningfully, Θ must be decomposable into the Cartesian product form (6.2) with $M > 1$. When this is not the case, one can sometimes find a subset $\Theta' \subset \Theta$ that does have this form, and which contains a solution to (6.1). One can then reformulate the problem by substituting Θ' for Θ.

6.3 Mathematical Preliminaries and Assumptions

In this section, we overview mathematical ideas and assumptions that will arise in the analysis to follow.

6.3.1 General Mathematical Background

A closed d-dimensional ball of radius r and centered at $\boldsymbol{x} \in \mathbb{R}^d$ is denoted

$$B^d(r, \boldsymbol{x}) \stackrel{\triangle}{=} \left\{ \boldsymbol{x}' \in \mathbb{R}^d \,:\, ||\boldsymbol{x}' - \boldsymbol{x}|| \leq r \right\}.$$

where $||\cdot||$ is the standard Euclidean norm. For the minimization problem (6.1), we shall also use the notation

$$\mathcal{B}_{\mathcal{S}}(r, \boldsymbol{\xi}) \stackrel{\triangle}{=} \Theta_{\mathcal{S}} \cap \{ \boldsymbol{\xi}' \in \mathbb{R}_{\mathcal{S}} \,:\, ||\boldsymbol{\xi}' - \boldsymbol{\xi}|| \leq r \}.$$

to denote certain constrained balls.

Given vectors $\boldsymbol{x}_j \in \mathbb{R}^d$ and real scalars α_j, $j = 1, \ldots, N$ for which $\sum_{j=1}^N \alpha_j = 1$, the form $\sum_{j=1}^N \alpha_j \boldsymbol{x}_j$ is called an *affine combination* of these vectors. A set $G \in \mathbb{R}^d$ is

called affine if it contains all affine combinations of its members. Given a set $G \subset \mathbb{R}^d$, the *affine hull* $\text{aff}(G)$ of G is defined as the smallest affine set containing G or, equivalently, the set of all affine combinations of elements in G. A point $\boldsymbol{x} \in \mathbb{R}^d$ is said to lie in the *relative interior* $\text{ri}(G)$ if there exists some $r > 0$ such that $B^d(r, \boldsymbol{x}) \cap \text{aff}(G) \subset G$. When $\text{aff}(G) = \mathbb{R}^d$, then $\text{ri}(G)$ is simply the interior of G. We denote the closure of G by $\text{cl}(G)$. Recall that $\text{cl}(G)$ is the smallest closed set containing G or, equivalently, the set of all limits of sequences of points in G. The notation ∂G will denote the relative boundary, $\text{cl}(G) \setminus \text{ri}(G)$.

A set $G \in \mathbb{R}^d$ is said to be *discrete* if for each $\boldsymbol{x} \in G$, there exists an $r > 0$ such that $B^d(r, \boldsymbol{x}) \cap G = \{\boldsymbol{x}\}$. The points in G are then said to be *isolated*. A function $f : D \subset \mathbb{R}^d \to \mathbb{R}$ is said to be *connected* on a set $D_0 \subset D$ if (see [75, p. 98]), given any $\boldsymbol{x}, y \in D_0$, there exists a continuous function $\boldsymbol{g} : [0,\ 1] \to D_0$ such that $\boldsymbol{g}(0) = \boldsymbol{x}$, $\boldsymbol{g}(1) = y$, and

$$f(g(\alpha)) \leq \max\{f(\boldsymbol{x}), f(\boldsymbol{y})\}$$

for all $\alpha \in (0,\ 1)$. A set $C \subset \mathbb{R}^d$ is said to be *path-connected* if, given any $\boldsymbol{x}, \boldsymbol{y} \in C$ there exists a continuous function $\boldsymbol{g} : [0,\ 1] \to C$ such that $\boldsymbol{g}(0) = \boldsymbol{x}$ and $\boldsymbol{g}(1) = \boldsymbol{y}$. Convex and quasi-convex functions are simple examples of connected functions with $\boldsymbol{g}(\alpha) = \alpha\boldsymbol{y} + (1 - \alpha)\boldsymbol{x}$. Also, it has been shown (e.g., Theorem 4.2.4 in [75, p. 99]) that a function is connected if and only if its sublevel sets are path-connected.

Often, we will need to take gradients with respect to a subset of the components of a function's argument. Given a function $f(\boldsymbol{x}; \boldsymbol{y})$, we shall denote its gradient with respect to its first argument, $\boldsymbol{x}$, as $\nabla^{10} f(\boldsymbol{x}; \boldsymbol{y})$. Likewise, $\nabla^{20} f(\boldsymbol{x}; \boldsymbol{y})$ shall denote the Hessian with respect to $\boldsymbol{x}$. An expression like $\nabla_m \Phi(\boldsymbol{\theta})$, $m \in \{1, \ldots, M\}$ shall denote the gradient with respect to the sub-vector $\boldsymbol{\theta}_m \in \Theta_m$ of $\boldsymbol{\theta}$. Similarly, $\nabla_{\mathcal{S}} \Phi(\boldsymbol{\theta})$ is the gradient with respect to $\boldsymbol{\theta}_{\mathcal{S}}$.

A key question in the analysis to follow is whether the limit points of an MM algorithm (i.e., the limits of subsequences of $\{\boldsymbol{\theta}^i\}$) are stationary points of (6.1). By a stationary point of (6.1), we mean a feasible point $\boldsymbol{\theta}^*$ that satisfies the first order necessary optimality condition,[4]

$$\langle \nabla \Phi(\boldsymbol{\theta}^*),\, \boldsymbol{\theta} - \boldsymbol{\theta}^* \rangle \geq 0 \qquad \forall \boldsymbol{\theta} \in \Theta. \tag{6.12}$$

Here $\langle \cdot,\, \cdot \rangle$ is the usual Euclidean inner product. Henceforth, when an algorithm produces a sequence $\{\boldsymbol{\theta}^i\}$ whose limit points (if any exist) are stationary points of (6.1), we say that the algorithm and the sequence $\{\boldsymbol{\theta}^i\}$ are *asymptotically stationary* .

6.3.2 Assumptions on MM Algorithms

Throughout the chapter, we consider cost functions Φ and tangent majorants $\phi(\cdot; \bar{\boldsymbol{\theta}})$ that are continuously differentiable throughout open supersets of Θ and $D(\bar{\boldsymbol{\theta}})$ respectively. For every $\bar{\boldsymbol{\theta}}$, the domain $D(\bar{\boldsymbol{\theta}})$ is assumed convex. In addition, for a given MM algorithm and corresponding sequence $\{\phi^i(\cdot; \boldsymbol{\theta}^i)\}$, we impose conditions that fall into one of two categories. Conditions in the first category, listed next, are what we think of as regularity conditions. In this list, a condition enumerated (Ri.j) denotes a stronger condition than (Ri), i.e., (Ri.j) implies (Ri). Typical MM algorithms will satisfy these conditions to preclude certain degenerate behavior that could otherwise be exhibited.

(R1) The sequence $\{\boldsymbol{\theta}^i\}$ lies in a closed subset of Θ. Thus, any limit point of $\{\boldsymbol{\theta}^i\}$ is feasible.

(R1.1) The sequence $\{\boldsymbol{\theta}^i\}$ is contained in a compact (i.e., closed and bounded) subset of Θ.

---

[4] Recall that (6.12) is a more fundamental first order condition than the KKT conditions. Condition (6.12) is necessary for $\boldsymbol{\theta}^*$ to be a local minimizer of Φ and, if Φ is convex, sufficient for $\boldsymbol{\theta}^*$ to be a global minimizer (see Proposition 2.1.2 in [7, p. 194]).

(R2) For each i and $\boldsymbol{\xi} \in \Theta_{\mathcal{S}^i}$, the Gateaux differential,

$$\eta^i(\boldsymbol{\theta}; \boldsymbol{\xi}) \stackrel{\triangle}{=} \left\langle \nabla^{10}\phi^i(\boldsymbol{\theta}_{\mathcal{S}^i}; \boldsymbol{\theta}),\, \boldsymbol{\xi} - \boldsymbol{\theta}_{\mathcal{S}^i} \right\rangle \tag{6.13}$$

is continuous as a function of $\boldsymbol{\theta}$ throughout Θ. Furthermore,

$$\eta^i(\boldsymbol{\theta}^i; \boldsymbol{\xi}) = \left\langle \nabla_{\mathcal{S}^i}\Phi(\boldsymbol{\theta}^i),\, \boldsymbol{\xi} - \boldsymbol{\theta}^i_{\mathcal{S}^i} \right\rangle. \tag{6.14}$$

Thus, the directional derivatives of the tangent majorants $\{\phi^i(\cdot; \boldsymbol{\theta}^i)\}$ at their expansion points match those of the cost function in feasible directions.

(R2.1) For every i and $\bar{\boldsymbol{\theta}} \in \Theta_{\mathcal{S}^i}$,

$$\nabla^{10}\phi^i(\bar{\boldsymbol{\theta}}_{\mathcal{S}^i}; \bar{\boldsymbol{\theta}}) = \nabla_{\mathcal{S}^i}\Phi(\bar{\boldsymbol{\theta}}). \tag{6.15}$$

Here, the tangent majorant and cost function derivatives match in *all* directions (not just feasible ones) and at *all* expansion points (not just at the $\{\boldsymbol{\theta}^i\}$). Note that, under (R2.1), the continuity of any $\eta^i(\cdot; \boldsymbol{\xi})$ follows from (6.15) and the fact that Φ is continuously differentiable.

(R3) There exists an $r > 0$ such that $\mathcal{B}_{\mathcal{S}^i}(r, \boldsymbol{\theta}^i_{\mathcal{S}^i}) \subset D^i(\boldsymbol{\theta}^i)$ for all i. In other words, each tangent majorant is defined on a feasible neighborhood of some minimum size around its expansion point.

Aside from the above regularity conditions, most results will require specific combinations of the following technical conditions. Similar to before, a condition denoted (Ci.j) implies (Ci).

(C1) Each tangent majorant $\phi^i(\cdot; \boldsymbol{\theta}^i)$ is connected on its respective domain $D^i(\boldsymbol{\theta}^i)$.

(C2) The elements of the sequence $\{\phi^i(\cdot;\cdot)\}$ are chosen from a finite set of majorant generators.

(C3) For each fixed i, the majorant generator $\phi^i(\cdot;\cdot)$ is continuous throughout its domain $\mathcal{D}^i$. In addition, for any closed subset $\mathcal{Z}$ of Θ, there exists an $r^i_{\mathcal{Z}} > 0$ such that the set $\left\{(\boldsymbol{\xi},\boldsymbol{\theta}) \,:\, \boldsymbol{\xi} \in \mathcal{B}_{\mathcal{S}^i}(r^i_{\mathcal{Z}}, \boldsymbol{\theta}^i_{\mathcal{S}^i}), \boldsymbol{\theta} \in \mathcal{Z}\right\}$ lies in a closed subset of $\mathcal{D}^i$.

(C4) There exists an integer $J > 0$ and, for each $m \in \{1,\ldots,M\}$, an index set $\mathcal{S}^{(m)}$ containing m, a majorant generator $\phi^{(m)}(\cdot;\cdot)$, and a set $\mathcal{I}_m = \left\{i \,:\, \mathcal{S}^i = \mathcal{S}^{(m)}, \phi^i = \phi^{(m)}\right\}$ such that

$$\forall n \geq 0, \exists i \in [n, n+J] \text{ s.t. } i \in \mathcal{I}_m.$$

That is, every sub-vector $\boldsymbol{\theta}_m \in \Theta_m$, $m = 1 \ldots M$ of $\boldsymbol{\theta}$ is updated regularly by some $\phi^{(m)}$.

(C5) $\lim_{i\to\infty} ||\boldsymbol{\theta}^{i+1} - \boldsymbol{\theta}^i|| = 0$.

(C5.1) The sequence $\{\boldsymbol{\theta}^i\}$ has at least one feasible limit point. Also, there exists a $\gamma^- > 0$, such that for all i and $\boldsymbol{\xi}, \boldsymbol{\psi} \in D^i(\boldsymbol{\theta}^i)$,

$$\left\langle \nabla^{10}\phi^i(\boldsymbol{\xi};\boldsymbol{\theta}^i) - \nabla^{10}\phi^i(\boldsymbol{\psi};\boldsymbol{\theta}^i),\, \boldsymbol{\xi} - \boldsymbol{\psi} \right\rangle \geq \gamma^- ||\boldsymbol{\xi} - \boldsymbol{\psi}||^2.$$

In other words, the $\{\phi^i(\cdot;\boldsymbol{\theta}^i)\}$ are *strongly convex* with curvatures that are uniformly lower bounded in i. The fact that (C5.1) implies (C5) is proven in Lemma 6.3.5(c).

(C6) In addition to (R3), there exists a $\gamma^+ \geq 0$, such that for all i and $\boldsymbol{\xi} \in \mathcal{B}_{\mathcal{S}^i}(r, \boldsymbol{\theta}^i_{\mathcal{S}^i})$ (here $\mathcal{B}_{\mathcal{S}^i}(r, \boldsymbol{\theta}^i_{\mathcal{S}^i})$ is as in (R3)),

$$\left\langle \nabla^{10}\phi^i(\boldsymbol{\xi};\boldsymbol{\theta}^i) - \nabla^{10}\phi^i(\boldsymbol{\theta}^i_{\mathcal{S}^i};\boldsymbol{\theta}^i),\, \boldsymbol{\xi} - \boldsymbol{\theta}^i_{\mathcal{S}^i} \right\rangle \leq \gamma^+ ||\boldsymbol{\xi} - \boldsymbol{\theta}^i_{\mathcal{S}^i}||^2.$$

In other words, the curvatures of the tangent majorants are uniformly upper bounded along line segments emanating from their expansion points. The line segments must extend to the boundary of a feasible neighborhood of size r around the expansion points.

There are a variety of standard conditions under which Condition (R1) will hold. The simplest case is if Θ is itself closed. Alternatively, (R1) will hold if one can show that the sublevel sets $\mathrm{sublev}_\tau \Phi \triangleq \{\boldsymbol{\theta} \in \Theta : \Phi(\boldsymbol{\theta}) \leq \tau\}$ of Φ are closed, which is often a straightforward exercise. In the latter case, with $\tau_0 = \Phi(\boldsymbol{\theta}^0)$, the level set $\mathrm{sublev}_{\tau_0} \Phi$ is closed, and because $\{\Phi(\boldsymbol{\theta}^i)\}$ is montonically non-increasing, it follows that the entire sequence $\{\boldsymbol{\theta}^i\}$ is contained in this set. Similarly, if Θ (or just $\mathrm{sublev}_{\tau_0} \Phi$) is compact, then (R1.1) holds. The closure or compactness of sublevel sets often follows if Φ is coercive, i.e., tends to infinity at the boundary of Θ.

The simplest case in which (R3) holds is when $D^i(\boldsymbol{\theta}) = \Theta_{\mathcal{S}^i}$ for all i and $\boldsymbol{\theta} \in \Theta$. A typical situation in which (C4) holds is if the index sets $\{\mathcal{S}^i\}$ and the majorant generators $\{\phi^i(\cdot;\cdot)\}$ are chosen cyclically. Condition (C5) has frequently been encountered in the study of feasible direction methods (e.g., [75, p. 474]). Condition (C5.1) is a sufficient condition for (C5) that is relatively easy to verify. It is essentially a generalization of Condition 5 in [33].

Remark 6.3.1 In the MM literature, the stronger condition (R2.1) is used customarily to ensure (R2). However, this can be excessive as discussed in Note A.3.

Remark 6.3.2 Equation (6.14) is, in fact, implied whenever $\mathrm{aff}(D^i(\bar{\boldsymbol{\theta}})) = \mathrm{aff}(\Theta_{\mathcal{S}^i})$ and $\bar{\boldsymbol{\theta}}_{\mathcal{S}^i} \in \mathrm{ri}(D^i(\bar{\boldsymbol{\theta}}))$. For details, see Note A.2.

6.3.3 More Preliminaries

We now give several lemmas that facilitate the analysis in this chapter. Most of these lemmas are slight generalizations of existing results.

Lemma 6.3.3 (Functions with curvature bounds) *Suppose $f : D \subset \mathbb{R}^d \to \mathbb{R}$ is a continuously differentiable function on a convex set D and let $\boldsymbol{y} \in D$.*

(a) If $\langle \nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y}),\, \boldsymbol{x} - \boldsymbol{y} \rangle \leq \gamma^+ ||\boldsymbol{x} - \boldsymbol{y}||^2$ *for some* $\gamma^+ > 0$ *and* $\forall \boldsymbol{x} \in D$*, then likewise*

$$f(\boldsymbol{x}) - f(\boldsymbol{y}) \leq \langle \nabla f(\boldsymbol{y}),\, \boldsymbol{x} - \boldsymbol{y} \rangle + \frac{1}{2}\gamma^+ ||\boldsymbol{x} - \boldsymbol{y}||^2 \qquad \forall \boldsymbol{x} \in D.$$

(b) If $\langle \nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y}),\, \boldsymbol{x} - \boldsymbol{y} \rangle \geq \gamma^- ||\boldsymbol{x} - \boldsymbol{y}||^2$*, for some* $\gamma^- > 0$ *and* $\forall \boldsymbol{x} \in D$*, then likewise*

$$f(\boldsymbol{x}) - f(\boldsymbol{y}) \geq \langle \nabla f(\boldsymbol{y}),\, \boldsymbol{x} - \boldsymbol{y} \rangle + \frac{1}{2}\gamma^- ||\boldsymbol{x} - \boldsymbol{y}||^2 \qquad \forall \boldsymbol{x} \in D.$$

Proof. Assume first that the assumptions of part (a) hold. Since D is convex, the scalar function $f(\boldsymbol{y} + t(\boldsymbol{x} - \boldsymbol{y}))$ is defined for $t \in [0,\, 1]$. Moreover, since f is continuousl$\boldsymbol{y}$ differentiable, then the directional derivative $\langle \nabla f(\boldsymbol{y} + t(\boldsymbol{x} - \boldsymbol{y})),\, \boldsymbol{x} - \boldsymbol{y} \rangle$ is Riemann integrable as a function of t in the interval $[0,\, 1]$. Thus, by the fundamental theorem of calculus,

$$\begin{aligned}
f(\boldsymbol{x}) - f(\boldsymbol{y}) &= \int_0^1 \langle \nabla f(\boldsymbol{y} + t(\boldsymbol{x} - \boldsymbol{y})),\, \boldsymbol{x} - \boldsymbol{y} \rangle \, \mathrm{d}t \\
&= \langle \nabla f(\boldsymbol{y}),\, \boldsymbol{x} - \boldsymbol{y} \rangle + \int_0^1 \langle \nabla f(\boldsymbol{y} + t(\boldsymbol{x} - \boldsymbol{y})) - \nabla f(\boldsymbol{y}),\, \boldsymbol{x} - \boldsymbol{y} \rangle \, \mathrm{d}t \\
&\leq \langle \nabla f(\boldsymbol{y}),\, \boldsymbol{x} - \boldsymbol{y} \rangle + \gamma^+ ||\boldsymbol{x} - \boldsymbol{y}||^2 \int_0^1 t \, \mathrm{d}t \\
&= \langle \nabla f(\boldsymbol{y}),\, \boldsymbol{x} - \boldsymbol{y} \rangle + \frac{1}{2}\gamma^+ ||\boldsymbol{x} - \boldsymbol{y}||^2.
\end{aligned}$$

Virtually identical manipulations, but with reversed inequalities, establish part (b). □

Remark 6.3.4 Results similar to Lemma 6.3.3 are often proved under slightly stronger assumptions (e.g., Proposition A.24 in [7, p. 667]).

Lemma 6.3.5 (Consequences of the existence of limit points) *Suppose that* $\{\boldsymbol{\theta}^i\}$ *is an MM sequence with a limit point* $\boldsymbol{\theta}^* \in \Theta$*. Then*

(a) $\{\Phi(\boldsymbol{\theta}^i)\} \searrow \Phi(\boldsymbol{\theta}^*)$.

(b) If $\boldsymbol{\theta}^{**} \in \Theta$ *is another limit point of* $\{\boldsymbol{\theta}^i\}$*, then* $\Phi(\boldsymbol{\theta}^{**}) = \Phi(\boldsymbol{\theta}^*)$.

(c) If, in addition, (C5.1) *holds then,* $\lim_{i\to\infty} ||\boldsymbol{\theta}^i - \boldsymbol{\theta}^{i+1}|| = 0$.

Proof.

(a) Let $\{\boldsymbol{\theta}^{i_k}\}$ be a subsequence converging to $\boldsymbol{\theta}^*$. The continuity of Φ then implies that $\Phi(\boldsymbol{\theta}^{i_k}) \to \Phi(\boldsymbol{\theta}^*)$. Since $\{\Phi(\boldsymbol{\theta}^i)\}$ is monotonically non-increasing, we conclude that $\{\Phi(\boldsymbol{\theta}^i)\} \searrow \Phi(\boldsymbol{\theta}^*)$.

(b) Immediate from part (a) and the uniqueness of the limit of $\{\Phi(\boldsymbol{\theta}^i)\}$.

(c) Since (C5.1) holds, then Lemma 6.3.3(b) applies with $f = \phi^i(\cdot;\boldsymbol{\theta}^i)$, $D = D^i(\boldsymbol{\theta}^i)$, $x = \boldsymbol{\theta}^i_{\mathcal{S}^i}$, and $y = \boldsymbol{\theta}^{i+1}_{\mathcal{S}^i}$,

$$\phi^i(\boldsymbol{\theta}^i_{\mathcal{S}^i};\boldsymbol{\theta}^i) - \phi^i(\boldsymbol{\theta}^{i+1}_{\mathcal{S}^i};\boldsymbol{\theta}^i) \geq \left\langle \nabla^{10}\phi^i(\boldsymbol{\theta}^{i+1}_{\mathcal{S}^i};\boldsymbol{\theta}^i),\, \boldsymbol{\theta}^i_{\mathcal{S}^i} - \boldsymbol{\theta}^{i+1}_{\mathcal{S}^i}\right\rangle + \frac{1}{2}\gamma^- ||\boldsymbol{\theta}^i_{\mathcal{S}^i} - \boldsymbol{\theta}^{i+1}_{\mathcal{S}^i}||^2 \quad (6.16)$$

for all i. Since $\phi^i(\cdot;\boldsymbol{\theta}^i)$ is convex with minimizer $\boldsymbol{\theta}^{i+1}$,

$$\left\langle \nabla^{10}\phi^i(\boldsymbol{\theta}^{i+1}_{\mathcal{S}^i};\boldsymbol{\theta}^i),\, \boldsymbol{\theta}^i_{\mathcal{S}^i} - \boldsymbol{\theta}^{i+1}_{\mathcal{S}^i}\right\rangle \geq 0.$$

In addition, due to (6.3),

$$\phi^i(\boldsymbol{\theta}^i_{\mathcal{S}^i};\boldsymbol{\theta}^i) - \phi^i(\boldsymbol{\theta}^{i+1}_{\mathcal{S}^i};\boldsymbol{\theta}^i) \leq \Phi(\boldsymbol{\theta}^i) - \Phi(\boldsymbol{\theta}^{i+1}).$$

Incorporating these observations into (6.16),

$$||\boldsymbol{\theta}^i_{\mathcal{S}^i} - \boldsymbol{\theta}^{i+1}_{\mathcal{S}^i}||^2 \leq \frac{2}{\gamma^-}(\Phi(\boldsymbol{\theta}^{i+1}) - \Phi(\boldsymbol{\theta}^i)),$$

and since $\boldsymbol{\theta}^i_{\tilde{\mathcal{S}}^i} = \boldsymbol{\theta}^{i+1}_{\tilde{\mathcal{S}}^i}$, this is equivalent to

$$||\boldsymbol{\theta}^i - \boldsymbol{\theta}^{i+1}||^2 \leq \frac{2}{\gamma^-}(\Phi(\boldsymbol{\theta}^{i+1}) - \Phi(\boldsymbol{\theta}^i)).$$

Due to part (a), the limit of the right hand side of this inequality is 0 and the result follows.

□

Lemma 6.3.6 (Convergence to isolated stationary points) *Suppose* $\{\boldsymbol{\theta}^i\}$ *is a sequence of points lying in a compact set* $\mathcal{K} \subset \Theta$ *and whose limit points* $S \subset \mathcal{K}$ *are stationary points of* (6.1)*. Let* $\mathcal{C}$ *denote the set of all stationary points of* (6.1) *in* $\mathcal{K}$*. If either of the following is true,*

(a) $\mathcal{C}$ *is a singleton, or*

(b) Condition (C5) *holds and* $\mathcal{C}$ *is a discrete set.*

then $\{\boldsymbol{\theta}^i\}$ *in fact converges to a point in* $\mathcal{C}$*.*

Proof. Since $\{\boldsymbol{\theta}^i\}$ lies in a compact set, convergence is established by showing that S is a singleton. The fact that S contains only stationary points implies that $S \subset \mathcal{C}$. Therefore, in case (a) it is readily seen that S is a singleton. Alternatively, suppose that (b) is true. Then, since $S \subset \mathcal{C}$ and $\mathcal{C}$ is discrete, then likewise S is discrete. In addition, since $\mathcal{K}$ is bounded and (C5) holds, then S is also connected (see p.173 of [76]). Since S is both discrete and connected, it is a singleton. □

6.4 Asymptotic Stationarity and Convergence to Isolated Stationary Points

In this section, we establish conditions under which MM algorithms are asymptotically stationary. Convergence in norm is then proved under standard supplementary assumptions that the stationary points are isolated (see Theorem 6.4.5). Theorem 6.4.1, our first result, establishes that non-block alternating MM sequences are asymptotically stationary under quite mild assumptions. Two sets of assumptions are considered. One set involves (C3), a continuity condition similar to that used in previous MM literature (e.g., [102, 33, 72]). In the second set, the central condition is (C6), which requires a uniform local upper bound on the tangent majorant curvatures. To our knowledge, we are the first to consider such a condition. Note also that, in Theorem 6.4.1, the tangent majorants can

be non-convex.

Theorem 6.4.1 (Asymptotically stationary: non-block alternating case) *Suppose that all* $\mathcal{S}^i = \{1, \ldots, M\}$*, that* $\{\boldsymbol{\theta}^i\}$ *is an MM sequence generated by* (6.11)*, and that the regularity conditions* (R1)*,* (R2)*, and* (R3) *hold. Suppose further that either* (C6) *or the pair of conditions* {(C2), (C3)} *holds. Then any limit point of* $\{\boldsymbol{\theta}^i\}$ *is a stationary point of* (6.1)*.*

Proof. Suppose $\boldsymbol{\theta}^* \in \Theta$ is a limit point of $\{\boldsymbol{\theta}^i\}$ (it must lie in Θ due to (R1)) and, aiming for a contradiction, let us assume that it is not a stationary point. Then there exists a $\boldsymbol{\theta}' \neq \boldsymbol{\theta}^* \in \Theta$ such that

$$\left\langle \nabla\Phi(\boldsymbol{\theta}^*),\ \frac{\boldsymbol{\theta}' - \boldsymbol{\theta}^*}{||\boldsymbol{\theta}' - \boldsymbol{\theta}^*||} \right\rangle < 0. \tag{6.17}$$

Since $\nabla\Phi$ is continuous, then, with (R2) and (R3), it follows that there exists a constant $c < 0$ and a subsequence $\{\boldsymbol{\theta}^{i_k}\}$ satisfying, for all k,

$$||\boldsymbol{\theta}' - \boldsymbol{\theta}^{i_k}|| \geq \min(r, ||\boldsymbol{\theta}' - \boldsymbol{\theta}^*||/2) \stackrel{\triangle}{=} \bar{t}, \tag{6.18}$$

where r is as in (R3), and

$$\left\langle \nabla^{10}\phi^k(\boldsymbol{\theta}^{i_k}; \boldsymbol{\theta}^{i_k}),\ \frac{\boldsymbol{\theta}' - \boldsymbol{\theta}^{i_k}}{||\boldsymbol{\theta}' - \boldsymbol{\theta}^{i_k}||} \right\rangle \leq c. \tag{6.19}$$

Define the unit-length direction vectors

$$\mathbf{s}^k \stackrel{\triangle}{=} \frac{\boldsymbol{\theta}' - \boldsymbol{\theta}^{i_k}}{||\boldsymbol{\theta}' - \boldsymbol{\theta}^{i_k}||}, \quad \mathbf{s}^* \stackrel{\triangle}{=} \frac{\boldsymbol{\theta}' - \boldsymbol{\theta}^*}{||\boldsymbol{\theta}' - \boldsymbol{\theta}^*||}$$

and the scalar functions

$$h_k(t) \stackrel{\triangle}{=} \phi^{i_k}(\boldsymbol{\theta}^{i_k} + t\mathbf{s}^k; \boldsymbol{\theta}^{i_k}) - \left[\phi^{i_k}(\boldsymbol{\theta}^{i_k}; \boldsymbol{\theta}^{i_k}) - \Phi(\boldsymbol{\theta}^{i_k})\right], \qquad t \in [0,\ \bar{t}\,]. \tag{6.20}$$

Due to (R3) and (6.18), all h_k are well-defined on the common interval indicated. The next several inequalities follow from (6.11), (6.10), and Lemma 6.3.5(a), respectively,

$$h_k(t) \geq \phi^{i_k}(\boldsymbol{\theta}^{i_k+1}; \boldsymbol{\theta}^{i_k}) - \left[\phi^{i_k}(\boldsymbol{\theta}^{i_k}; \boldsymbol{\theta}^{i_k}) - \Phi(\boldsymbol{\theta}^{i_k})\right] \geq \Phi(\boldsymbol{\theta}^{i_k+1}) \tag{6.21}$$

$$\geq \Phi(\boldsymbol{\theta}^*). \tag{6.22}$$

The remainder of the proof addresses separately the cases where {(C6)} and {(C2), (C3)} hold.

First, assume that (C6) holds. This, together with Lemma 6.3.3(a), implies that for $t \in [0,\ \bar{t}\,]$,

$$h_k(t) - h_k(0) \leq \dot{h}_k(0)t + \frac{\gamma^+}{2}t^2.$$

However, $h_k(0) = \Phi(\boldsymbol{\theta}^{i_k})$, while $\dot{h}_k(0) \leq c$ due to (6.19). These observations, together with (6.22), leads to

$$\Phi(\boldsymbol{\theta}^*) - \Phi(\boldsymbol{\theta}^{i_k}) \leq ct + \frac{\gamma^+}{2}t^2 \qquad t \in [0,\ \bar{t}\,]\,.$$

Passing to the limit in k,

$$ct + \frac{\gamma^+}{2}t^2 \geq 0, \qquad t \in [0,\ \bar{t}\,]\,.$$

Finally, dividing this relation through by t and letting $t \searrow 0$ yields $c \geq 0$, contradicting the assumption that $c < 0$, and completing the proof for this case.

Now, assume {(C2), (C3)}. In light of (C2), we can redefine our subsequence $\{\boldsymbol{\theta}^{i_k}\}$ so that, in addition to (6.18) and (6.19), $\phi^k(\cdot;\cdot)$ equals some fixed function $\hat{\phi}(\cdot;\cdot)$ for all k. That and (6.21) give, for $t \in [0,\ \bar{t}\,]$,

$$h_k(t) = \hat{\phi}(\boldsymbol{\theta}^{i_k} + t\mathbf{s}^k; \boldsymbol{\theta}^{i_k}) - \left[\hat{\phi}(\boldsymbol{\theta}^{i_k}; \boldsymbol{\theta}^{i_k}) - \Phi(\boldsymbol{\theta}^{i_k})\right] \geq \Phi(\boldsymbol{\theta}^{i_k+1}). \tag{6.23}$$

From (R1), we know that $\{\boldsymbol{\theta}^{i_k}\}$ lies in a closed subset $\mathcal{Z}$ of Θ. With (C3), there therefore exists a positive $r_{\mathcal{Z}} \leq \bar{t}$ such that $h_k(t)$, as given in (6.23), converges as $k \to \infty$ to $h^*(t) \triangleq \hat{\phi}(\boldsymbol{\theta}^* + t\mathbf{s}^*; \boldsymbol{\theta}^*) - \left[\hat{\phi}(\boldsymbol{\theta}^*; \boldsymbol{\theta}^*) - \Phi(\boldsymbol{\theta}^*)\right]$ for all $t \in [0,\ r_{\mathcal{Z}}]$. Letting $k \to \infty$ in (6.23) therefore yields,

$$h^*(t) \geq \Phi(\boldsymbol{\theta}^*) \qquad \forall t \in [0,\ r_{\mathcal{Z}}]\,. \tag{6.24}$$

The function $h^*(t)$ is differentiable at $t = 0$ due to (R2). Now, $h_k(0) = \Phi(\boldsymbol{\theta}^{i_k})$, so that in the limit, $h^*(0) = \Phi(\boldsymbol{\theta}^*)$. Thus, we have that (6.24) holds with equality at $t = 0$, from

which it follows that

$$\dot{h}^*(0) \geq 0. \tag{6.25}$$

However, $\dot{h}_k(0) \leq c$ due to (6.19), and the continuity requirement in (R2) implies that $\dot{h}_k(0)$ converges to $\dot{h}^*(0)$ as $k \to \infty$. Thus, we have in the limit that $\dot{h}^*(0) \leq c < 0$, contradicting (6.25). □

The following example provides a simple illustration of how an MM algorithm can be non-asymptotically stationary when the assumptions of Theorem 6.4.1 are not met. From this example, one can see that the requirements of Theorem 6.4.1 are not excessive and give meaningful guidelines for the design of majorant generators.

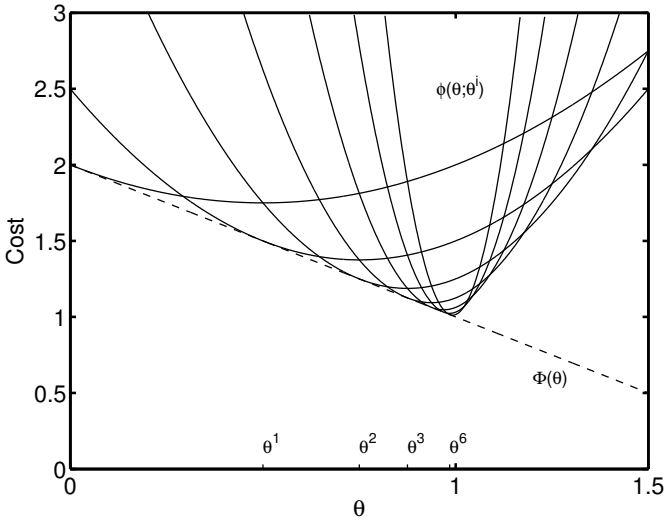

Figure 6.1: Illustration of Example 6.4.2. The MM sequence $\{\theta^i\}$ converges to a non-stationary point. This is possible since the conditions of Theorem 6.4.1 are not satisfied.

Example 6.4.2 Consider the case $\mathbb{R}^p = \mathbb{R}$, $\Theta = [0, 1.5]$, and $\Phi(\theta) = 2 - \theta$. Take $\theta^0 = 0$

and let $\{\theta^i\}$ be the sequence generated via (6.11) with

$$\phi^i(\theta;\bar{\theta}) = \phi(\theta;\bar{\theta}) \stackrel{\triangle}{=} c(\bar{\theta})(\theta-\bar{\theta})^2 + \Phi(\theta)$$

$$c(\bar{\theta}) \stackrel{\triangle}{=} \begin{cases} 1 & \bar{\theta} = 1 \\ \frac{1}{|\bar{\theta}-1|} & \bar{\theta} \neq 1 \end{cases}$$

It is immediate from the definition of $\phi(\theta;\bar{\theta})$ that every $\phi(\cdot;\bar{\theta})$ is a true tangent majorant. The resulting sequence of iterates $\{\theta^i\}$ and tangent majorants $\phi(\cdot;\theta^i)$ are depicted for several iterations in Figure 6.1. By induction, one can readily determine that $\theta^i = 1 - 2^{-i}$. Hence, $\{\theta^i\}$ converges to 1 which is not a stationary point. This presents no conflict with Theorem 6.4.1, however. The tangent majorants do not satisfy condition (C6), since the tangent majorant curvatures $\{c(\theta^i) = 2^i\}$ tend to infinity. Also, $\phi(\theta;\cdot)$ is discontinuous at $\bar{\theta} = 1$, so (C3) is not satisfied. Consequently, the hypothesis of Theorem 6.4.1 does not hold.

Remark 6.4.3 The kind of discontinuities exhibited in Example 6.4.2 can arise in EM majorant generators because of a discontinuity in the KL distance (see Note A.4).

The next result addresses the block alternating case, but requires additional conditions, namely (C4) and (C5). (Although, Condition (C2) is no longer required.) These conditions, however, are no stronger than those invoked previously in [33]. Condition (C4) is a generalization of [33, Condition 6]. Condition (C5) is an implied condition in [33], as shown in Lemma 3 in that paper.

Theorem 6.4.4 (Asymptotic stationarity: block alternating case) *Suppose that* $\{\boldsymbol{\theta}^i\}$ *is an MM sequence generated by* (6.5) *and* (6.6) *and that the regularity conditions* (R1)*,* (R2)*, and* (R3) *hold. Suppose, further, that* (C4)*,* (C5) *and either* (C6) *or* (C3) *holds. Then any limit point of* $\{\boldsymbol{\theta}^i\}$ *is a stationary point of* (6.1).

Proof. Suppose $\boldsymbol{\theta}^* \in \Theta$ is a limit point of $\{\boldsymbol{\theta}^i\}$ (it must lie in Θ due to (R1)) and, aiming for a contradiction, let us assume that it is not a stationary point. In light of (6.2), there therefore exists a $\boldsymbol{\theta}' \neq \boldsymbol{\theta}^* \in \Theta$ and an $m \in \{1, \ldots, M\}$, such that

$$\langle \nabla_m \Phi(\boldsymbol{\theta}^*),\, \boldsymbol{\theta}'_m - \boldsymbol{\theta}^*_m \rangle < 0 \tag{6.26}$$

and such that $\boldsymbol{\theta}'_{\tilde{m}} = \boldsymbol{\theta}^*_{\tilde{m}}$, $\forall \tilde{m} \neq m$. Then, with $\mathcal{S}^{(m)}$ as in (C4), it follows from (6.26) that,

$$\left\langle \nabla_{\mathcal{S}^{(m)}} \Phi(\boldsymbol{\theta}^*),\, \frac{\boldsymbol{\theta}'_{\mathcal{S}^{(m)}} - \boldsymbol{\theta}^*_{\mathcal{S}^{(m)}}}{||\boldsymbol{\theta}'_{\mathcal{S}^{(m)}} - \boldsymbol{\theta}^*_{\mathcal{S}^{(m)}}||} \right\rangle < 0. \tag{6.27}$$

Now, consider a subsequence $\{\boldsymbol{\theta}^{i_k}\}$ converging to $\boldsymbol{\theta}^*$. We can assume that $\mathcal{S}^{i_k} = \mathcal{S}^{(m)}$ and $\phi^{i_k} = \phi^{(m)}$, for otherwise, in light of (C4), we could construct an alternative subsequence $\{\boldsymbol{\theta}^{i_k + J_k}\}$, $J_k \leq J$ which does have this property. Furthermore, this alternative subsequence would converge to $\boldsymbol{\theta}^*$ due to (C5).

In light of (6.27), we can also choose $\{\boldsymbol{\theta}^{i_k}\}$ so that, similar to the proof of Theorem 6.4.1,

$$||\boldsymbol{\theta}' - \boldsymbol{\theta}^{i_k}|| \geq \min(r, ||\boldsymbol{\theta}' - \boldsymbol{\theta}^*||/2) \stackrel{\triangle}{=} \bar{t}.$$

and

$$\left\langle \nabla^{10} \phi^{(m)}(\boldsymbol{\theta}^{i_k}_{\mathcal{S}^{(m)}}; \boldsymbol{\theta}^{i_k}),\, \frac{\boldsymbol{\theta}'_{\mathcal{S}^{(m)}} - \boldsymbol{\theta}^{i_k}_{\mathcal{S}^{(m)}}}{||\boldsymbol{\theta}'_{\mathcal{S}^{(m)}} - \boldsymbol{\theta}^{i_k}_{\mathcal{S}^{(m)}}||} \right\rangle \leq c.$$

for some $c < 0$. Now define

$$\mathbf{s}^k \stackrel{\triangle}{=} \frac{\boldsymbol{\theta}'_{\mathcal{S}^{(m)}} - \boldsymbol{\theta}^{i_k}_{\mathcal{S}^{(m)}}}{||\boldsymbol{\theta}'_{\mathcal{S}^{(m)}} - \boldsymbol{\theta}^{i_k}_{\mathcal{S}^{(m)}}||}$$

and

$$h_k(t) \stackrel{\triangle}{=} \phi^{(m)}(\boldsymbol{\theta}^{i_k}_{\mathcal{S}^{(m)}} + t\mathbf{s}^k; \boldsymbol{\theta}^{i_k}) - \left[\phi^{(m)}(\boldsymbol{\theta}^{i_k}_{\mathcal{S}^{(m)}}; \boldsymbol{\theta}^{i_k}) - \Phi(\boldsymbol{\theta}^{i_k})\right], \qquad t \in [0,\, \bar{t}\,].$$

The form and properties of this $h_k(t)$ is a special case of that defined in (6.20). Under (C6), a verbatim argument as in the proof of Theorem 6.4.1 therefore leads to the contradiction $c \geq 0$, completing the proof for this case. Likewise, the $h_k(t)$ above has the same form

and properties as in (6.23). The arguments in the proof of Theorem 6.4.1 following (6.23) relied only on (C3), and complete the proof of this theorem as well. □

In the following theorem, we deduce convergence in norm by adding discreteness assumptions on the stationary points of (6.1).

Theorem 6.4.5 (Convergence of MM sequences to isolated stationary points) *Suppose* $\{\boldsymbol{\theta}^i\}$ *is an MM sequence satisfying* (R1.1)*, as well as the conditions of either Theorem 6.4.1 or Theorem 6.4.4. Suppose, in addition, that either of the following is true.*

(a) The problem (6.1) *has a unique solution as its sole stationary point, or*

(b) Condition (C5) *holds and* (6.1) *has a discrete set of stationary points.*

Then $\{\boldsymbol{\theta}^i\}$ *converges to a stationary point. Moreover, in case* (a)*, the limit is the unique solution of* (6.1)*.*

Proof. Under (R1.1), $\{\boldsymbol{\theta}^i\}$ lies in a compact subset of Θ. Moreover, the limit points of $\{\boldsymbol{\theta}^i\}$ are all guaranteed to be stationary by either Theorem 6.4.1 or Theorem 6.4.4. The result then follows from Lemma 6.3.6. □

Remark 6.4.6 The convergence analysis in [33] is less general than stated due to an error in the proof of Lemma 6 in that paper. The error occurs where it is argued "if $\nabla_k^{10}\phi^{(k)}(\boldsymbol{\theta}^i_{\mathcal{S}^{(k)}};\boldsymbol{\theta}^i) > 0$ then $\boldsymbol{\theta}_k^{i+1} > \boldsymbol{\theta}_k^i$". This argument would be valid only if, in addition to what was already assumed, $\phi^{(k)}(\cdot;\boldsymbol{\theta}^i)$ were a function of a single variable. Due to the analysis in the present chapter, however, we can claim that the *conclusions* of [33] are indeed valid, even if the arguments are not. This follows from Theorem 6.4.5(a) above, which implies convergence under conditions no stronger than those assumed in [33].

6.5 The Capture Property of Connected Tangent Majorants

When Φ is non-convex, one often thinks of its graph as consisting of many *capture basins*, i.e., high dimensional analogues of valley-shaped regions, each containing a local minimum. In this section, we show that, if the tangent majorants are connected, the MM algorithm will remain confined to such a region. This property, which we call the *capture property* of MM, has a variety of consequences that we shall discuss.

To proceed with our analysis, we require a formal mathematical definition of a capture basin. The following definition describes what we call a generalized capture basin. It includes the kind of regions that one traditionally thinks of as a capture basin as a special case.

Definition 6.5.1 We say that a set $G \subset \Theta$ is a *generalized capture basin* (with respect to the minimization problem (6.1)) if, for some $\boldsymbol{\theta} \in G$, the following is never violated

$$\Phi(\boldsymbol{\theta}) < \Phi(\tilde{\boldsymbol{\theta}}), \qquad \tilde{\boldsymbol{\theta}} \in \mathrm{cl}(G) \cap \mathrm{cl}(\Theta \setminus G). \tag{6.28}$$

Moreover, we say that such a $\boldsymbol{\theta}$ is *well-contained* in G.

Thus, a point is well-contained in G if it has lower cost than any point $\tilde{\boldsymbol{\theta}}$ in the common boundary $\mathrm{cl}(G) \cap \mathrm{cl}(\Theta \setminus G)$ between G and its complement. The definition is worded so that $\mathrm{cl}(G) \cap \mathrm{cl}(\Theta \setminus G)$ can be empty. Thus, for example, the whole feasible set Θ always constitutes a generalized capture basin (provided that it contains some $\boldsymbol{\theta}$), because $\mathrm{cl}(\Theta) \cap \mathrm{cl}(\Theta \setminus \Theta)$ is empty, implying that (6.28) can never be violated.

Remark 6.5.2 The regions described by Definition 6.5.1 are a bit more general than traditional notions of a capture basin in a few ways. In particular, the definition requires neither that Φ be unimodal over G, nor that G be path-connected, nor that Φ attain its maximum over G in ∂G. However, it is straightforward to show that any generalized capture basin

G must have the same dimension as Θ, in the sense that $\text{aff}(G) = \text{aff}(\Theta)$ (see Note A.5). Thus, for example, if $\Theta = \mathbb{R}^2$, no line segment inside Θ can constitute a generalized capture basin. This is consistent with common intuition.

The following proposition lays the foundation for the results of this section. It asserts that, if the expansion point of a connected tangent majorant is well-contained in a generalized capture basin G, then any point that decreases the cost value of that tangent majorant (relative to the expansion point) is likewise well-contained in G. For the case where the tangent majorant takes arguments in $\mathbb{R}^p$ (i.e., excluding tangent majorants used for block alternation), Figure 6.2 shows how this result can be interpreted in terms of the tangent majorant sublevel sets.

Proposition 6.5.3 *Suppose that $\phi(\cdot;\bar{\boldsymbol{\theta}})$ is a tangent majorant that is connected on its domain $D(\bar{\boldsymbol{\theta}}) \subset \Theta_{\mathcal{S}}$ and whose expansion point $\bar{\boldsymbol{\theta}} \in \Theta$ is well-contained in a generalized capture basin G. Suppose, further, that $\boldsymbol{\theta} \in \Theta$ satisfies*

$$\begin{aligned} &\boldsymbol{\theta}_{\mathcal{S}} \in D(\bar{\boldsymbol{\theta}}), \quad \boldsymbol{\theta}_{\tilde{\mathcal{S}}} = \bar{\boldsymbol{\theta}}_{\tilde{\mathcal{S}}}, \\ &\phi(\boldsymbol{\theta}_{\mathcal{S}};\bar{\boldsymbol{\theta}}) \leq \phi(\bar{\boldsymbol{\theta}}_{\mathcal{S}};\bar{\boldsymbol{\theta}}), \end{aligned} \tag{6.29}$$

Then $\boldsymbol{\theta}$ is likewise well-contained in G.

Proof. It is sufficient to show that $\boldsymbol{\theta} \in G$. For taking any $\tilde{\boldsymbol{\theta}} \in \text{cl}(G) \cap \text{cl}(\Theta \setminus G)$, and then combining (6.29), (6.3), and the fact that $\bar{\boldsymbol{\theta}}$ is well-contained in G,

$$\Phi(\boldsymbol{\theta}) \leq \Phi(\bar{\boldsymbol{\theta}}) < \Phi(\tilde{\boldsymbol{\theta}}), \tag{6.30}$$

implying that $\boldsymbol{\theta}$ is also well-contained in G. Aiming for a contradiction, suppose that $\boldsymbol{\theta} \in \Theta \setminus G$. Since $\phi(\cdot;\bar{\boldsymbol{\theta}})$ is connected on $D(\bar{\boldsymbol{\theta}})$, there exists a continuous function $\mathbf{g}$:

$[0,\ 1] \to \Theta$ with $\mathbf{g}(0) = \bar{\boldsymbol{\theta}}$, $\mathbf{g}(1) = \boldsymbol{\theta}$, and such that, for all $\alpha \in (0,\ 1)$, one has

$$[\mathbf{g}(\alpha)]_{\mathcal{S}} \in D(\bar{\boldsymbol{\theta}}), \quad [\mathbf{g}(\alpha)]_{\tilde{\mathcal{S}}} = \bar{\boldsymbol{\theta}}_{\tilde{\mathcal{S}}},$$

$$\phi([\mathbf{g}(\alpha)]_{\mathcal{S}}; \bar{\boldsymbol{\theta}}) \leq \max\{\phi(\bar{\boldsymbol{\theta}}_{\mathcal{S}}; \bar{\boldsymbol{\theta}}),\ \phi(\boldsymbol{\theta}_{\mathcal{S}}; \bar{\boldsymbol{\theta}})\} = \phi(\bar{\boldsymbol{\theta}}_{\mathcal{S}}; \bar{\boldsymbol{\theta}}), \tag{6.31}$$

where the equality in (6.31) is due to (6.29). Also, since $\mathbf{g}(0) = \bar{\boldsymbol{\theta}} \in G$,

$$\alpha^* \stackrel{\triangle}{=} \sup\{\alpha \in [0,1] \,:\, \mathbf{g}(\alpha) \in G\}$$

is well-defined. Finally, let $\boldsymbol{\psi} = \mathbf{g}(\alpha^*)$.

We now argue that $\boldsymbol{\psi} \in \text{cl}(G) \cap \text{cl}(\Theta \setminus G)$. Firstly, due to the definition of α^*, there must exist a sequence $0 \leq \hat{\alpha}_j \leq \alpha^*$, $j = 1, 2, \ldots$ with $\hat{\alpha}_j \to \alpha^*$. Since $\mathbf{g}(\hat{\alpha}_j) \to \boldsymbol{\psi}$, by continuity, and all $\mathbf{g}(\hat{\alpha}_j) \in G$, it follows that $\boldsymbol{\psi} \in \text{cl}(G)$. Secondly, the definition of α^* also implies that, if $\alpha^* < \alpha \leq 1$, then $\mathbf{g}(\alpha) \in \Theta \setminus G$. Together with the fact that $\mathbf{g}(1) = \boldsymbol{\theta} \in \Theta \setminus G$, it follows that there is a sequence $\alpha^* \leq \check{\alpha}_j \leq 1$, $j = 1, 2, \ldots$ with $\check{\alpha}_j \to \alpha^*$ and $\mathbf{g}(\check{\alpha}_j) \in \Theta \setminus G$. Since $\mathbf{g}(\check{\alpha}_j) \to \boldsymbol{\psi}$, we have that $\boldsymbol{\psi} \in \text{cl}(\Theta \setminus G)$ as well. We conclude that $\boldsymbol{\psi} \in \text{cl}(G) \cap \text{cl}(\Theta \setminus G)$ as claimed. Therefore, from the rightmost inequality in (6.30), we have, with $\tilde{\boldsymbol{\theta}} = \boldsymbol{\psi}$,

$$\Phi(\bar{\boldsymbol{\theta}}) < \Phi(\boldsymbol{\psi}) = \Phi([\mathbf{g}(\alpha^*)]_{\mathcal{S}}, \bar{\boldsymbol{\theta}}_{\tilde{\mathcal{S}}}). \tag{6.32}$$

With (6.3), this implies that $\phi([\mathbf{g}(\alpha^*)]_{\mathcal{S}}; \bar{\boldsymbol{\theta}}) > \phi(\bar{\boldsymbol{\theta}}_{\mathcal{S}}; \bar{\boldsymbol{\theta}})$ contradicting (6.31). □

Using Proposition 6.5.3, we obtain the following result as an immediate consequence. It articulates the capture property of MM for generalized capture basins.

Theorem 6.5.4 (Capture property of MM and generalized capture basins) *Suppose that* $\{\boldsymbol{\theta}^i\}$ *is an MM sequence generated by* (6.5) *and* (6.6)*. In addition, suppose that some iterate* $\boldsymbol{\theta}^n$ *is well-contained in a generalized capture basin* G *and that the tangent majorant sequence* $\{\phi^i(\cdot; \boldsymbol{\theta}^i)\}_{i=n}^{\infty}$ *satisfies* (C1)*. Then likewise* $\boldsymbol{\theta}^i$ *is well-contained in* G *for all* $i > n$*.*

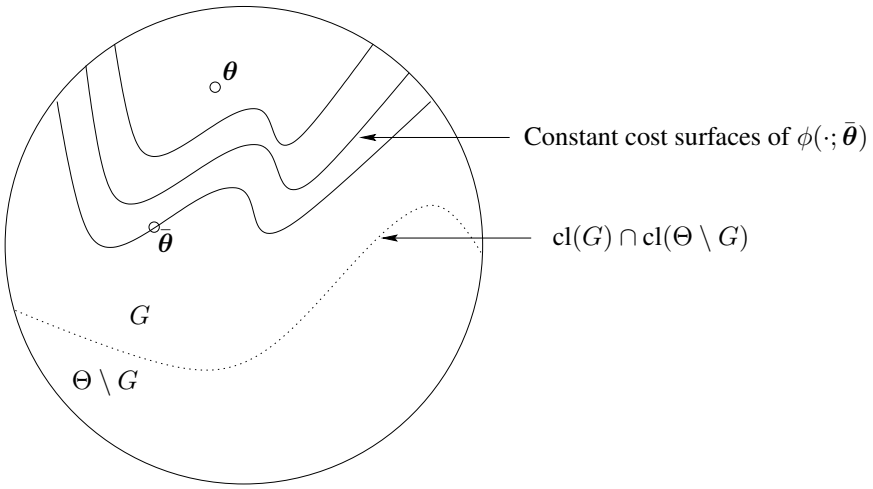

Figure 6.2: Illustration of the sublevel set containment property of connected tangent majorants. Suppose that $\phi(\cdot;\bar{\boldsymbol{\theta}})$ takes arguments in $\mathbb{R}^p$ and satisfies the assumptions of Proposition 6.5.3. Then the proposition implies that the sublevel sets of $\phi(\cdot;\bar{\boldsymbol{\theta}})$ of level $\phi(\bar{\boldsymbol{\theta}};\bar{\boldsymbol{\theta}})$ or less are strictly contained in G.

Proof. The result follows from Proposition 6.5.3 and an obvious induction argument. □

Remark 6.5.5 Note that Proposition 6.5.3 and Theorem 6.5.4 are fundamental properties in that they rely on (C1), but none of the regularity conditions described in Section 6.3.2. Also, using Proposition 6.5.3, one can obtain the same conclusions as in Theorem 6.5.4 if the sequence $\{\boldsymbol{\theta}^i\}$ merely satisfied $\phi^i(\boldsymbol{\theta}^{i+1}_{\mathcal{S}^i};\boldsymbol{\theta}^i) \leq \phi^i(\boldsymbol{\theta}^i_{\mathcal{S}^i};\boldsymbol{\theta}^i)$ rather than (6.5). This is relevant to practical situations, since one often does not obtain the exact minimizers in (6.5).

The capture property of MM is linked to the global information implicit in tangent majorants in general, and in connected tangent majorants in particular. The algorithm designer uses insights into the global shape of Φ to derive a function satisfying the defining property (6.3). Still more global information is needed to ensure that the tangent majorant is connected. This collective information allows the algorithm descent mechanism to respect the boundaries of a generalized capture basin, even though the location of these boundaries may not be known explicitly to the algorithm designer. Textbook-variety algorithms not assisted by such global information clearly will not imitate the capture prop-

erty reliably. Such algorithms include derivative-based feasible direction methods (e.g., steepest descent or Newton's method), possibly combined with *ad hoc* constant step-size choices or numerical line-search operations.

Algorithms using constant step-sizes will clearly escape a generalized capture basin if the step-size is chosen too large in comparison to the size of this region. Avoiding such large choices of step-sizes therefore requires foreknowledge of the size of the surrounding generalized capture basin, a degree of global information no less than that inherent in MM. Proposition 1.2.5 in [7, p. 52] describes a capture property for gradient methods with constant step-sizes. However, the region of capture in that Proposition is smaller than the set of well-contained points and becomes smaller as the step-size is increased.

Common numerical line search methods (bisection, Armijo, ...) can likewise let the algorithm escape a generalized capture basin. This is because many points on the search line can satisfy the termination criteria of the line search method and not all of these points are guaranteed to lie within the smallest surrounding generalized capture basin. Bisection, for example, can find any 1D stationary point on the search line and, for non-convex Φ, many such points may exist, some lying within the local generalized capture basin and some without. To ensure capture, one would need to restrict the search operations to the line segment intersecting the surrounding generalized capture basin. Here again, though, global information would be required to locate the boundaries of this line segment.

Our first application of the capture property is in deriving the following local version of Theorem 6.4.5.

Theorem 6.5.6 (Convergence of MM sequences to isolated stationary points (local form)) *In addition to the assumptions of Theorem 6.5.4, suppose that the conditions of either Theorem 6.4.1 or Theorem 6.4.4 are satisfied. Suppose further that G is bounded and either of the following are true*

(a) $\mathrm{cl}(G)$ *contains a single stationary point, or*

(b) Condition (C5) *holds and the set of stationary points in* $\mathrm{cl}(G)$ *is discrete.*

Then $\{\boldsymbol{\theta}^i\}$ *converges to a stationary point* $\boldsymbol{\theta}^* \in \mathrm{cl}(G)$.

Proof. Since G is bounded, it follows from Theorem 6.5.4 that the sequence $\{\boldsymbol{\theta}^i\}$ lies in the compact set $\mathcal{K} = \mathrm{cl}(G)$. Moreover, all limit points of $\{\boldsymbol{\theta}^i\}$ are stationary, as assured by either Theorem 6.4.1 or Theorem 6.4.4. The conclusions of the Theorem then follow from Lemma 6.3.6. □

Naturally, an instance of part (a) of primary interest is the case where the region $\mathrm{cl}(G)$ contains a single stationary point which is also a global minimizer. For then, the theorem guarantees convergence to a solution of (6.1). Traditionally, local convergence results for minimization algorithms, such as Proposition 1.2.5 in [7, p. 52], identify a region of convergence that is unimodal and basin-like around the local minimizer. The converse, however, is not usually true. Given a unimodal, basin-like region around a local minimizer, one cannot usually conclude that convergence to that minimizer will take place from any point in that region. In a sense, therefore, Theorem 6.5.6(a) is the strongest kind of local convergence result, ensuring convergence over the largest possible region that one can hope for.

Apart from its role in local convergence, the capture property makes MM an appropriate instrument for implementing the basin-probing steps in various non-convex minimization strategies. Perhaps the most standard strategy is is try to obtain, by heuristics or *ad hoc* methods, an initial point believed to reasonably approximate the desired solution and to hope that this point lies in a unimodal capture basin around the global minimizer. The strategy then tries to descend locally, within the capture basin, to reach the global minimizer. Figure 2.3 illustrates how the MM capture property facilitates this kind of basin-search.

There, the MM sequence results from convex (and hence connected) tangent majorants. Consequently, the sequence is confined to the basin-like region in the interval $\{B,\ C]$ that, fortunately, contains the global minimizer.

If the graph of Φ is clustered with peaks and valleys, a single basin-probing step may not be sufficient. In this case, another standard strategy is to do several basin-searches, as outlined above, but using different initial points. The idea is to search locally around those points and to find the deepest basin. To implement this strategy in a *principled* way, it is highly desirable to do the basin-searches with an algorithm endowed with the capture property. Otherwise, the basin-searches could converge to any stationary point on the graph of Φ and one has no assurance that distinct basins will be probed.

A third example worth mentioning is a path-following method due to [8] called Graduated Non-Convexity (GNC). The GNC strategy employs a sequence of increasingly accurate approximations $\{F(\cdot, t_k)\}_{k=1}^{K}$ of $\Phi(\cdot)$, begining with a convex function $F(\cdot, t_1)$ that can be easily globally minimized, and ending with $F(\cdot, t_K) = \Phi(\cdot)$. By globally minimizing each $F(\cdot, t_k)$, a sequence $\{\boldsymbol{\theta}^*(t_k)\}_{k=1}^{K}$ is obtained which, one hopes, converges to the global minimum of Φ. Moreover, each minimization step is initialized with the result of the previous minimization so that the $\{\boldsymbol{\theta}^*(t_k)\}$, one hopes, are obtained incrementally. In well-behaved circumstances, the initial point of each minimization step will lie in a capture basin containing the solution to the current minimization problem (see Figure 6.3). Therefore, an algorithm endowed with the capture property is desirable here.

For certain problems, it may be too expensive to employ strategies involving multiple basin-searches. In such cases, and if Φ is clustered with peaks and valleys, then MM with connected tangent majorants is hazardous, because the likelihood of getting trapped at a sub-optimal solution is high. This limitation is important to recognize, since convex tangent majorants are the most common type used. In such situations, it is worthwhile to

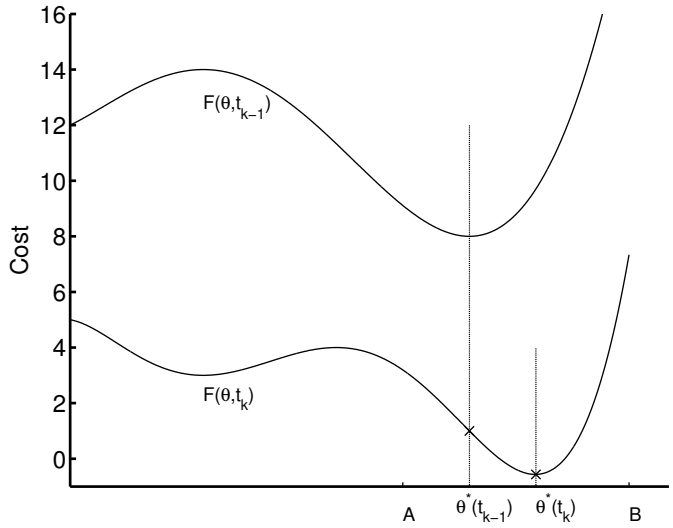

Figure 6.3: Illustration of the capture property as it pertains to the Graduated Non-Convexity (GNC) strategy.

consider algorithms not endowed with the capture property. For example, one may try to derive an MM algorithm with non-connected tangent majorants. However, we know of no instance where the absence of the capture property has been systematically exploited. Any success that such algorithms, be they MM or otherwise, have had in avoiding sub-optimal minima seems largely fortuitous.

6.6 An MM Algorithm Design for JEDM

One of the reasons why it is natural to consider MM algorithms in the context of this dissertation is that we are faced with difficult cost function reduction tasks such as (4.22) in Algorithm 4.4.1. As discussed in Section 4.4, we currently use an off-the-shelf line search method (the Armijo rule) to accomplish this, for lack of a better alternative.

However, the form of Φ_{JEDM} as a function of $\boldsymbol{\alpha}_t$ is not particularly line search friendly. Along the line $\{\boldsymbol{\alpha}_t + \ell\Delta\boldsymbol{\alpha}_t \,:\, 0 \leq \ell \in \mathbb{R}\}$, it looks like

$$a(\ell) = f(\boldsymbol{PW}(\boldsymbol{\alpha}_t + \ell\Delta\boldsymbol{\alpha}_t)\boldsymbol{u})$$

The nonlinear form of $\boldsymbol{PW}(\boldsymbol{\alpha}_t)\boldsymbol{u}$, as a function of $\boldsymbol{\alpha}_t$, cannot be simplifed in any apparent

way. Thus, for each evaluation of $a(\ell)$ in a line search, a forward projection step via $\boldsymbol{P}$ is needed. This is a substantial computational expense, and is a classic kind of situation that the MM methodology tries to circumvent.

In particular, it motivates us to try to find majorant generators for Φ_{JEDM} that are easy to minimize. For then, the reductions in (4.22) could be implemented, with greater computational ease, by minimizing corresponding the tangent majorants. Furthermore, if as before (4.21) is also implemented via MM steps, then Algorithm 4.4.1 becomes a block alternating algorithm overall, and the convergence theorems that we have developed here may apply.

The cost reduction in (4.22) is equivalent to reducing a cost function

$$\Phi_t(\boldsymbol{\alpha}_t) \stackrel{\triangle}{=} \text{KL}(\boldsymbol{y}_t, \tau_t(\boldsymbol{P}\boldsymbol{W}(\boldsymbol{\alpha}_t)\boldsymbol{u} + \boldsymbol{r}_t)) \tag{6.33}$$

where here $\boldsymbol{u} \geq \boldsymbol{0}$ is fixed. This cost function is one term of Φ_{JEDM}. All other terms are independent of $\boldsymbol{\alpha}_t$. In what follows, we discuss some ideas for constructing majorant generators for $\Phi_t(\boldsymbol{\alpha}_t)$. These ideas are similar to those in [47].

6.6.1 A Non-Convex Tangent Majorant

In [32], several majorant generators were presented for the function

$$L(\boldsymbol{\lambda}) = \text{KL}(\boldsymbol{y}_t, \tau_t(\boldsymbol{P}\boldsymbol{\lambda} + \boldsymbol{r}_t)). \tag{6.34}$$

One of these, from which the ML-EM-3 algorithm was derived, had the form

$$\phi(\boldsymbol{\lambda}; \bar{\boldsymbol{\lambda}}) = \text{KL}(\boldsymbol{e}(\bar{\boldsymbol{\lambda}}), \boldsymbol{\lambda} + \boldsymbol{m}) \tag{6.35}$$

for certain image vectors $0 < \boldsymbol{m} \in \mathbb{R}^{|\mathcal{G}|}$ and $0 \leq \boldsymbol{e}(\bar{\boldsymbol{\lambda}}) \in \mathbb{R}^{|\mathcal{G}|}$. By substituting $\boldsymbol{\lambda} = \boldsymbol{W}(\boldsymbol{\alpha}_t)\boldsymbol{u}$ in (6.34), one obtains $\Phi_t(\boldsymbol{\alpha}_t)$. Moreover, since the majorization relationship $\phi(\cdot; \bar{\boldsymbol{\lambda}}) \succ L(\cdot)$ is preserved by function composition, one can obtain a majorant generator

for $\Phi_t(\boldsymbol{\alpha}_t)$ by making the same substitution into (6.35),

$$\begin{aligned}\phi(\boldsymbol{\alpha}_t;\bar{\boldsymbol{\alpha}}_t) &= \mathrm{KL}(\boldsymbol{e}(\boldsymbol{W}(\bar{\boldsymbol{\alpha}}_t)\boldsymbol{u}), \boldsymbol{W}(\boldsymbol{\alpha}_t)\boldsymbol{u}+\boldsymbol{m}) \\ &= \sum_{j\in\mathbb{R}^{|\mathcal{G}|}} [[\boldsymbol{W}(\boldsymbol{\alpha}_t)\boldsymbol{u}]_j - e_j(\boldsymbol{W}(\bar{\boldsymbol{\alpha}}_t)\boldsymbol{u})\log([\boldsymbol{W}(\boldsymbol{\alpha}_t)\boldsymbol{u}]_j + m_j)] + \mathrm{const.} \qquad (6.36)\end{aligned}$$

The tangent majorant $\phi(\cdot;\bar{\boldsymbol{\alpha}}_t)$ is already more line search friendly than $\Phi_t(\boldsymbol{\alpha}_t)$. The operations $\boldsymbol{W}(\boldsymbol{\alpha}_t)\boldsymbol{u}$ are now the principal ones in the computation of the RHS of (6.36), and methods have been developed, namely in Chapter 5, to accelerate that still further.

It is straightforward to verify that an MM algorithm based on (6.36) satisfies both pairs of hypotheses in Theorem 6.4.1. Asymptotic stationarity is assured by the Theorem so long as the succesion of tangent majorants can be globally minimized. However, $\phi(\cdot;\boldsymbol{\alpha}_t)$ is non-convex and therefore this global minimization is not easily accomplished. One could partially minimize $\phi(\cdot;\boldsymbol{\alpha}_t)$ consistent with some sufficient decrease rule, however that rule would need to be derived. Alternatively, one could implement (4.22) by partially minimizing $\phi(\cdot;\boldsymbol{\alpha}_t)$, but infrequently interrupt Algorithm 4.4.1 with an iteration of non-block alternating steepest descent. The steepest descent iterations constitute so-called *spacer steps* and would guarantee asymptotic stationarity due to [7, Proposition 1.2.6].

6.6.2 A Convex Tangent Majorant

Another alternative is to derive a convex $\phi'(\cdot;\bar{\boldsymbol{\alpha}}_t)$ that is a tangent majorant for $\phi(\cdot;\bar{\boldsymbol{\alpha}}_t)$, and hence also one for $\Phi_t(\boldsymbol{\alpha}_t)$. Our idea for doing so is to construct $Q_j^+(\boldsymbol{\alpha}_{tC};\bar{\boldsymbol{\alpha}}_t)$ and $Q_j^-(\boldsymbol{\alpha}_{tC};\bar{\boldsymbol{\alpha}}_t)$, $C\in\{X,Y,Z\}$, that are respectively convex and concave functions of $\boldsymbol{\alpha}_{tC}$ and satisfy

$$Q_j^-(\cdot;\bar{\boldsymbol{\alpha}}_t) \overset{\bar{\boldsymbol{\alpha}}_t}{\prec} [\boldsymbol{W}(\cdot)\boldsymbol{u}]_j \overset{\bar{\boldsymbol{\alpha}}_t}{\prec} Q_j^+(\cdot;\bar{\boldsymbol{\lambda}},\bar{\boldsymbol{\alpha}}).$$

That is, Q_j^+ is a true majorant generator for $[\boldsymbol{W}(\cdot)\boldsymbol{u}]_j$ and Q_j^- is a true majorant generator for $-[\boldsymbol{W}(\cdot)\boldsymbol{u}]_j$, both with respect to $\boldsymbol{\alpha}_{tC}$. Since $-\log(\cdot)$ of a concave function is convex

and since $-\log(\cdot)$ is monotone increasing, it then follows (see also [47, Proposition 3.3]) that,

$$\phi'(\boldsymbol{\alpha}_{tC};\bar{\boldsymbol{\alpha}}_t) \triangleq \sum_j \left[Q_j^+(\boldsymbol{\alpha}_{tC};\bar{\boldsymbol{\alpha}}_t) - e_j(\boldsymbol{W}(\bar{\boldsymbol{\alpha}}_t)\boldsymbol{u})\log(Q_j^-(\boldsymbol{\alpha}_{tC};\bar{\boldsymbol{\alpha}}_t)+m_j)\right] \tag{6.37}$$

generates convex tangent majorants for Φ_t. It is the basis of a block alternating MM algorithm that alternates over blocks $\{\boldsymbol{u}, \boldsymbol{\alpha}_{tX}, \boldsymbol{\alpha}_{tY}, \boldsymbol{\alpha}_{tZ}\}$.

To derive such Q_j^+, one first observes that $[\boldsymbol{W}(\boldsymbol{\alpha}_t)\boldsymbol{u}]_j$ depends on each $\boldsymbol{\alpha}_{tC}$, $C \in \{X, Y, Z\}$, entirely through

$$d_{jC} = [\boldsymbol{\nu}_C + \boldsymbol{B}_C\boldsymbol{\alpha}_{tC}]_j.$$

Fixing, for example $C = Z$, one can write this as

$$[\boldsymbol{W}(\boldsymbol{\alpha}_t)\boldsymbol{u}]_j = F_j(\ell)\,\big|_{\ell=[\boldsymbol{\nu}_C+\boldsymbol{B}_C\boldsymbol{\alpha}_{tC}]_j} \tag{6.38}$$

where $F_j(\ell)$ is the 1D function

$$F_j(\ell) = \sum_{mn} w\left(d_{jX}(\boldsymbol{\alpha}_X) - m\right) w\left(d_{jY}(\boldsymbol{\alpha}_Y) - n\right) \times \underbrace{\sum_p \boldsymbol{u}_{mnp} w\left(\ell - p\right)}_{\triangleq f_{mn}(\ell)} \tag{6.39}$$

Since the substitution in (6.38) is affine in $\boldsymbol{\alpha}_{tC}$, the problem now reduces to replacing each 1D function $f_{mn}(\ell)$, defined in (6.39) with a 1D convex quadratic true tangent majorant,

$$q_{mn}^+(\ell,\bar{\ell}) = \bar{\ell} + \dot{f}_{mn}(\bar{\ell})(\ell-\bar{\ell}) + \frac{1}{2}v_{mn}(\bar{\ell})(\ell-\bar{\ell})^2.$$

Here the $v_{mn}(\bar{\ell}) \geq 0$ are curvatures chosen (ideally as small as possible and) so that $q_{mn}^+(\cdot,\bar{\ell}) \succ f_{mn}$. One can determine $v_{mn}(\bar{\ell}) \geq 0$ by looking at the 1D piecewise cubic difference polynomial

$$\delta(\ell) = q_{mn}^+(\ell,\bar{\ell}) - f_{mn}(\ell)$$

and computing how large v_{mn} needs to be so that $\bar{\ell}$ is the only root of $\delta(\ell)$. These v_{mn} can be tabulated over a suitably discretized interval of ℓ. Note that $v_{mn}(\ell)$ must be uniformly bounded above in ℓ since $f_{mn}(\ell)$ is of bounded curvature. We can also design $v_{mn}(\ell)$ so that it is uniformly bounded from below. The motivation for doing so will become clear shortly.

Once $q^+_{mn}(\ell, \bar{\ell})$ is substituted for $f_{mn}(\ell)$ in (6.39) one obtains a quadratic true tangent majorant $F^+_j(\cdot; \bar{\ell}) \succ F_j(\cdot)$ with curvatures,

$$V_j(\bar{\ell}) = \sum_{mn} v_{mn}(\bar{\ell}) w\left(d_{jX}(\boldsymbol{\alpha}_X) - m\right) w\left(d_{jY}(\boldsymbol{\alpha}_Y) - n\right).$$

Thus, once the v_{mn} have been tabulated, the curvature computation requires only a 2D interpolation (cf. Section 5.3.2). Finally, Q^+_j is obtained by replacing F_j with $F^+_j(\cdot; [\boldsymbol{\nu}_C + \boldsymbol{B}_C \bar{\boldsymbol{\alpha}}_{tC}]_j)$ in (6.38). An analogous process is used to derive Q^-_j except that instead of $f_{mn} \prec q^+_{mn}$, one has $v_{mn} \leq 0$ and $f_{mn} \succ q^-_{mn}$.

The composition of functions approach that was used to derive (6.37) has a complexity. Namely, since the $Q^-_j(\cdot; \bar{\boldsymbol{\alpha}}_t)$ are concave, the log terms in (6.37) is undefined except in the convex region $D(\bar{\boldsymbol{\alpha}}_t) = \left\{\boldsymbol{\alpha}_t \,:\, Q^-_j(\boldsymbol{\alpha}_t; \bar{\boldsymbol{\alpha}}_t) > -m_j\right\}$. Thus, we have a case where $D(\bar{\boldsymbol{\alpha}}_t)$ is a $\bar{\boldsymbol{\alpha}}_t$-dependent strict subset of the entire feasible set. Previous work on block alternating MM [32, 33] does not tell us whether such majorant generators induce asymptotic stationarity. Therefore we invoke our own result, Theorem 6.4.4. The main observations needed to do so is that (C5.1) and (C6) hold at iterations over $\boldsymbol{\alpha}_t$. The other conditions required by Theorem 6.4.4 are reasonably self-evident.

In the degenerate case where no limit points exist, clearly no non-stationary ones exist either. If limit points do exist, the remaining hypotheses of (C5.1) hold due to the design choice, mentioned above, that v_{mn} is uniformly bounded from below. This forces $Q^+_j(\cdot; \bar{\boldsymbol{\alpha}}_t)$ and hence also $\phi'(\cdot; \bar{\boldsymbol{\alpha}}_t)$ to be uniformly strongly convex as required. That (C6) holds is readily seen when one considers the following facts:

(a) The curvatures of $Q_j^+(\cdot;\bar{\boldsymbol{\alpha}}_t)$ are uniformly bounded from above (because $v_{mn}(\ell)$ is so bounded).

(b) In an MM sequence, the quantities $Q_j^-(\bar{\boldsymbol{\alpha}}_{tC}^i;\bar{\boldsymbol{\alpha}}_t^i)+m_j$ (i.e., the log term arguments in (6.37) when evaluated at the expansion points), must stay uniformly bounded away from zero. Otherwise, the value of the true tangent majorants at the expansion points, and hence also of the cost, would diverge to infinity in (6.37), thus contradicting the monotonicity of the algorithm.

(c) The derivatives of $Q_j^-(\cdot;\bar{\boldsymbol{\alpha}}_t^i)$ are globally bounded. This is because the first derivatives coincide with those of $[\boldsymbol{W}(\boldsymbol{\alpha}_t)\boldsymbol{u}]_j$, which are continuously differentiable, finitely supported functions. The second derivatives are bounded, once again by the design of v_{mn}.

The resulting tangent majorant $\phi'(\cdot;\bar{\boldsymbol{\alpha}}_t)$ is even more line search friendly than the tangent majorant $\phi(\cdot;\bar{\boldsymbol{\alpha}}_t)$, devised in Section 6.6.1. This can be seen by comparing the RHS of (6.36) with (6.37). Along the line $\{\boldsymbol{\alpha}_t+\ell\Delta\boldsymbol{\alpha}_t\,:\,0\leq\ell\in\mathbb{R}\}$, the form of the latter simplies to

$$a(\ell)=\sum_{j=1}^{|\mathcal{G}|}\left[q_j^+(\ell)-e_j\log(q_j^-(\ell))\right]$$

where the $q_j^{\pm}(\ell)$ are 1D quadratics. Each evaluation of $a(\ell)$ requires $|\mathcal{G}|$ log operations and, due to the quadratics, $5|\mathcal{G}|$ memory transfers and $4\,|\mathcal{G}|$ multiplications. This is to be contrasted with the line evaluations in (6.36). There, the need to evaluate the various $[\boldsymbol{W}(\cdot)\boldsymbol{u}]_j$ by interpolation results in a minimum of $17|\mathcal{G}|$ memory transfers, $20\,|\mathcal{G}|$ multiplications and $|\mathcal{G}|$ log operations. This minimum figure corresponds to the case where the table-assisted interpolation technique of Chapter 5 is used.

6.7 Toward Conditions for Convergence to Non-Isolated Stationary Points*

In previous sections, we established convergence of MM sequences under the assumption that the stationary points of (6.1) are isolated. However, other investigators have found that certain MM algorithms are convergent even when a continuum of stationary points is present. This has been observed in certain applications of the EM algorithm (e.g., [13, 88]) and also coordinate descent [64], both of which, as noted earlier, are special cases of MM algorithms. It is natural to wonder, therefore, whether there are general conditions ensuring convergence when the stationary points are possibly not isolated. In this section, we discuss, and examine by way of examples, what conditions might be required.

It seems clear that (R1.1) must be hypothesized *a priori*. This is because an MM sequence can easily become unbounded if it is initialized in a region where the graph of Φ has a pathological shape. Taking a 1D example, suppose $\Phi(\theta) = \theta \exp(-\theta)$, $\Theta = \{\theta \in \mathbb{R} : \theta \geq 0\}$, and connected tangent majorants are used that satisfy the conditions of Theorem 6.4.1. Initializing with $\theta^0 > 1$ can only produce an unbounded sequence $\{\theta^i\}$. This must be the case since $G = (1, \infty)$ is a generalized capture basin where Φ has no stationary points. Thus, by Theorem 6.4.1 and Theorem 6.5.4, $\{\theta^i\}$ is confined to $(1, \infty)$ but has no limit points there. (Conversely, if $\theta^0 \in [0, 1)$, then $\{\theta^i\}$ must converge to the stationary point at $\theta = 0$.)

Another hypothesis that seems necessary for convergence is that the tangent majorants $\phi^i(\cdot; \boldsymbol{\theta}^i)$ have a uniform positive lower bound on their curvatures. Otherwise, they can become asymptotically flat in places, which makes oscillations possible. The next example demonstrates this.

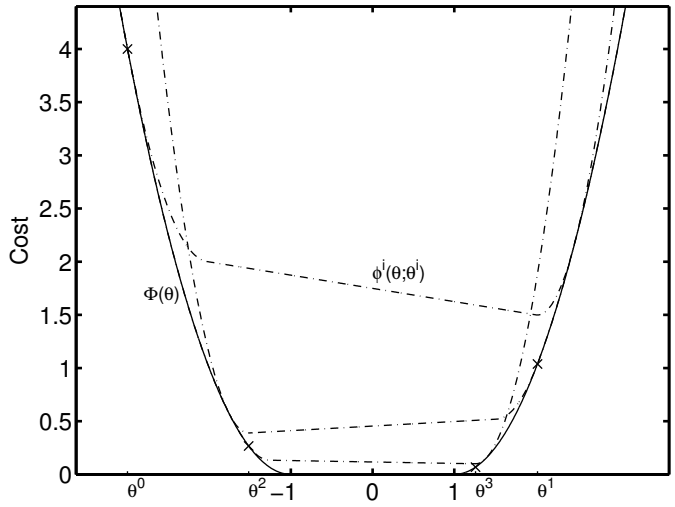

Figure 6.4: Illustration of Example 6.7.1. The sequence $\{\theta^i\}$ oscillates between points in the intervals $(-\infty,\ -1)$ and $(1,\ \infty)$, and therefore does not converge. The oscillations are possibly because the tangent majorants become asymptotically flat in the interval $[-1,\ 1]$.

Example 6.7.1 Suppose $p = 1$ and consider the cost function

$$\Phi(\theta) = \begin{cases} 0.25(|\theta| - 1)^2, & |\theta| \geq 1 \\ 0, & |\theta| < 1. \end{cases}$$

This cost function is piece-wise quadratic. In this example, we present an MM sequence $\{\theta^i\}$, obtained from tangent majorants that are also piece-wise quadratic, which, as in Figure 6.4, oscillates between the intervals $(-\infty,\ -1)$ and $(1,\ \infty)$. To construct the relevant tangent majorants, we first define, for any $d > 0$,

$$\begin{aligned} \gamma_d &= 3 + \left(4 - \sqrt{\frac{17}{4}\, d^2 + 16\, d + 16}\right) d^{-1} \\ B_d &= 2d(\gamma_d - 1) \\ A_d &= 1 + 2(B_d - d)^2/d^2 \\ g_1(t, d) &= (t + 1)^2 \\ g_2(t, d) &= 2(t + d + 1)^2 - 2d(t + d + 1) + d^2 \\ g_3(t, d) &= (0.5d^2\gamma_d^2 - d^2\gamma_d + d^2) + B_d(t + d + 1 - 0.5d\gamma_d) \\ g_4(t, d) &= A_d(t - 1 - 0.5d)^2 + B_d(t - 1 - 0.5d) + 0.375d^2, \end{aligned}$$

and

$$g(t;d) = \begin{cases} g_1(t,d) & t \leq -d-1 \\ g_2(t,d) & -d-1 < t \leq -d-1+0.5d\gamma_d \\ g_3(t,d) & -d-1+0.5d\gamma_d < t \leq 1+0.5d \\ g_4(t,d) & t > 1+0.5d. \end{cases}$$

One can verify the following facts about the foregoing definitions. Firstly, $0 \leq \gamma_d \leq 1$. Secondly, $g(\cdot;d)$ is continuously differentiable with a non-decreasing derivative (and hence is convex). Thirdly, $g(\cdot;d) \overset{d}{\underset{\mathbb{R}}{\succ}} \Phi(\cdot)$. Fourthly, the unique minimizer of $g(\cdot;d)$ is

$$t_d^{\min} = 1 + 0.5d - \frac{B_d}{2A_d}. \tag{6.40}$$

Define the following majorant generator

$$\phi(\theta;\bar{\theta}) = \begin{cases} \Phi(\theta), & \left|\bar{\theta}\right| \leq 1 \\ g(-\operatorname{sign}(\bar{\theta})\theta; \left|\bar{\theta}\right| - 1), & \left|\bar{\theta}\right| > 1. \end{cases}$$

One can readily verify, from the aforementioned properties of f, that $\phi(\cdot;\cdot)$ satisfies the requisite properties of a majorant generator.

Let $\{\theta^i\}$ be the MM sequence produced by

$$\theta^0 = -3$$

$$\theta^{i+1} = \operatorname*{argmin}_{t\in\mathbb{R}} \phi(t;\theta^i)$$

The first few iterations of this sequence are shown in Figure 6.4. By considering (6.40) and the definition of ϕ, one can verify that $\theta^{i+1} > 1$ if $\theta^i < -1$ and vice-versa for all i. Thus, the sequence $\{\theta^i\}$ oscillates between the intervals $(-\infty,\ -1)$ and $(1,\ \infty)$ as claimed, and so cannot converge. This oscillatory behavior is possible precisely because the tangent majorants become progressively flatter in the interval $[-1,\ 1]$.

Because the algorithm is monotonic and Φ has compact sublevel sets, (R1.1) holds, so $\{\theta^i\}$ has a bounded set of limit points. It is natural to wonder whether these limit points are stationary. If non-stationary limit points exist, it means that $\{\Phi(\theta^i)\}$ cannot converge to the global minimum value of Φ. In fact, though, one can easily verify that the assumptions of Theorem 6.4.1 are satisfied. In the present example, both condition sets (C6) and $\{$(C2), (C3)$\}$ are met. Therefore, the limit points of $\{\theta^i\}$ are indeed stationary and consist of the two global minimizers $\{-1,\ 1\}$.

From the above discussion and example, boundedness of $\{\boldsymbol{\theta}^i\}$ and a lower bound on the tangent majorant curvatures seem to be minimum requirements for a convergence theory. For one-dimensional problems, i.e., when $p = 1$, we can show that these assumptions are also sufficient. This is formalized in the following theorem.

Theorem 6.7.2 *Suppose that* $p = 1$ *and that* $\{\theta^i\}$ *is an MM sequence generated by* (6.11). *Suppose, further, that* (R1.1) *and* (C5.1) *hold. Then* $\{\theta^i\}$ *converges to a point in* Θ.

Proof. Due to (R1.1), the sequence $\{\theta^i\}$ is bounded and has a compact set of limit points in Θ. Furthermore, since (C5.1) holds, then likewise (C5) holds, so that $\lim_{i\to\infty} ||\theta^{i+1} - \theta^i|| = 0$. A bounded sequence with this property has a connected set of limit points (see [76, p. 173]). In this context, where $\mathbb{R}^p = \mathbb{R}$, we can therefore conclude that the limit points of $\{\theta^i\}$ form a closed, bounded interval $[a,\ b] \subset \Theta$. It remains to show that $a = b$.

Aiming for a contradiction, suppose $a < b$, so that the limit points form an interval of positive length. Then, there exists some k such that $\theta^k \in (a,\ b)$. In addition, due to Lemma 6.3.5(b), $\Phi(\theta)$ is constant throughout $[a,\ b]$, and consequently $\frac{d\Phi(\theta^k)}{d\theta} = 0$. By first order minimality conditions, Equation (6.10) can therefore be satisfied only if

$$\frac{d\phi^k(\theta;\theta^k)}{d\theta}\,|_{\theta=\theta^k} = 0. \tag{6.41}$$

Now due to (C5.1), $\phi^k(\cdot;\theta^k)$ is strongly convex. Thus, (6.41) implies that θ^k is the unique

global minimizer of $\phi^k(\cdot;\theta^k)$. This, in turn, implies that $\theta^{k+1} = \theta^k$. Repeating this argument inductively, we can conclude that $\theta^i = \theta^k$ for all $i \geq k$. But then $\{\theta^i\}$ converges to θ^k, contradicting the assumption that the limit points constitute an interval of positive length. □

Corollary 6.7.3 *Suppose that* $p = 1$ *and that* $\{\theta^i\}$ *is an MM sequence generated by* (6.11). *Furthermore, suppose* (C5.1) *and the assumptions of Theorem 6.4.1 hold. Then* $\{\theta^i\}$ *converges to a stationary point in* Θ.

Proof. The assumptions of Theorem 6.7.2 hold, so $\{\theta^i\}$ converges to a feasible point. Since the assumptions of Theorem 6.4.1 holds as well, this limit is stationary. □

Remark 6.7.4 The assumptions of Theorem 6.7.2 but not Theorem 6.4.1 are satisfied in Example 6.4.2. The sequence $\{\theta^i\}$ therefore converges, but the limit is non-stationary.

It is common to obtain intuition about the behavior of multi-variable problems from single variable ones. Theorem 6.7.2, however, is one case where this can be misleading. As the next example shows, convergence may not occur if $p > 1$, even if the other assumptions of Theorem 6.7.2 are met.

Example 6.7.5 Suppose $\mathbb{R}^p = \mathbb{R}^2$ and consider the quadratic cost function

$$\Phi(\boldsymbol{\theta}) = \frac{1}{2}\left[(\theta_1)^2 + (\theta_2)^2\right]$$

In this example, we present a non-convergent MM sequence $\{\boldsymbol{\theta}^i\}$, produced by tangent majorants that are also quadratic, which starts at the point $\boldsymbol{\theta}^0 = [2\ 0]^T$ and spirals in such a way that its limit points are the entire unit circle (see Figure 6.5).

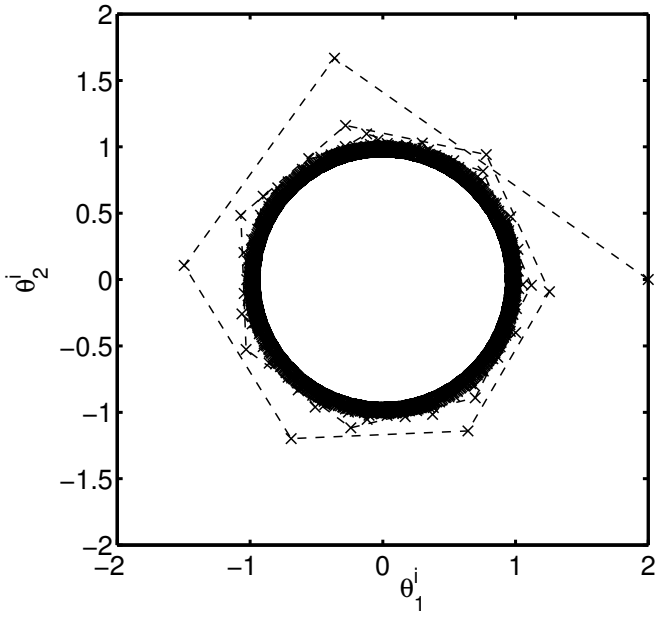

Figure 6.5: Illustration of Example 6.7.5. The sequence $\{\boldsymbol{\theta}^i\}$ spirals into an asymptotic orbit about the unit circle.

We begin with a series of definitions:

$$r_i = 1 + \sin\left(\frac{\pi}{2(i+1)}\right)$$

$$\alpha_i = \begin{cases} \sum_{j=1}^{i} \frac{\pi}{4j}, & i \geq 1 \\ 0, & i = 0 \end{cases}$$

$$\boldsymbol{\xi}^i = [r_i \cos\alpha_i,\ r_i \sin\alpha_i]$$

$$\lambda_i = \frac{r_i \cos\left(\frac{\pi}{4(i+1)}\right)}{r_i \cos\left(\frac{\pi}{4(i+1)}\right) - r_{i+1}}$$

$$R_i = \begin{bmatrix} \cos\alpha_{i+1} & -\sin\alpha_{i+1} \\ \sin\alpha_{i+1} & \cos\alpha_{i+1} \end{bmatrix}$$

$$\phi^i(\boldsymbol{\theta}; \bar{\boldsymbol{\theta}}) = (\boldsymbol{\theta} - \boldsymbol{\xi}^{i+1})^T R_i^T \begin{bmatrix} \lambda_i & 0 \\ 0 & 1 \end{bmatrix} R_i (\boldsymbol{\theta} - \boldsymbol{\xi}^{i+1})$$

We now argue that $\{\boldsymbol{\theta}^i\} = \{\boldsymbol{\xi}^i\}$ is an MM sequence produced using $\{\phi^i(\cdot;\cdot)\}$. To do so, we show that the $\{\phi^i(\cdot;\cdot)\}$ satisfy (6.4) and that $\{\boldsymbol{\theta}^i\}$ satisfies (6.11).

One can verify, from the definition of $\phi^i(\cdot;\cdot)$, that (6.15) holds. It is now sufficient to show that $\lambda_i \geq \frac{1}{2}$ for all i. For if this is true, the first implication is that all the Hessians $\nabla^{20}\phi^i(\cdot;\boldsymbol{\theta}^i)$ are non-negative definite throughout $\mathbb{R}^2$. This, in turn, implies that the quadratics $\phi^i(\cdot;\boldsymbol{\theta}^i)$ are convex with minima at $\boldsymbol{\xi}^{i+1}$ so that the recursion (6.11) is satisfied by $\{\boldsymbol{\theta}^i\} = \{\boldsymbol{\xi}^i\}$. In addition, if $\lambda_i \geq \frac{1}{2}$, one can verify that $\nabla^{20}\phi^i - \nabla^2\Phi$ is non-negative definite. Together with (6.15), this means that each $\phi^i(\cdot;\cdot)$ satisfies (6.4).

From the definition of λ_i, one can verify that $\lambda_i \geq \frac{1}{2}$ if

$$r_i \cos\left(\frac{\pi}{4(i+1)}\right) - r_{i+1} > 0 \tag{6.42}$$

Using Taylor's theorem with remainder, one can derive the following bounds for arbitrary $0 \leq t \leq 1$,

$$\begin{aligned} 1 + \sin\left(\frac{\pi t}{2}\right) &\geq 1 + \frac{\pi t}{2} - \frac{\pi^3 t^3}{48} \\ -\left(1 + \sin\left(\frac{\pi t}{2}\right)\right) &\geq -1 - \frac{\pi t}{2} \\ \cos\left(\frac{\pi t}{4}\right) &\geq 1 - \frac{\pi^2 t^2}{32}. \end{aligned}$$

Recalling the definition of r_i, the last three inequalities give lower bounds on the three trigonometric terms/factors on the LHS of (6.42). This leads to

$$r_i \cos\left(\frac{\pi}{4(i+1)}\right) - r_{i+1} > \frac{P(i)}{Q(i)}, \tag{6.43}$$

where $P(t)$ and $Q(t)$ are polynomials given by,

$$\begin{aligned} P(t) &= 1.2624\,t^4 + 3.6106\,t^3 + 2.1272\,t^2 - 1.3287\,t - 0.9085 \\ Q(t) &= (t+1)^5\,(t+2)\,. \end{aligned}$$

One can verify numerically that, for all $t \geq 1$, the rational function $\frac{P(t)}{Q(t)}$ has no real roots and is strictly positive. Hence, (6.42) follows from (6.43). We conclude that the $\{\phi^i(\cdot;\cdot)\}$

satisfy the requisite property (6.4) and that $\{\boldsymbol{\theta}^i\} = \{\boldsymbol{\xi}^i\}$ is the corresponding MM sequence.

From the definition of $\boldsymbol{\xi}^i$, one can see that $\{\boldsymbol{\theta}^i\}$ has the spiral, non-convergent trajectory of Figure 6.5. This is in spite of the fact that the conditions of Theorem 6.7.2 (apart from $p = 1$) are satisfied. The sequence is indeed bounded, again by direct inspection of the definition of $\boldsymbol{\xi}^i$. Moreover, the fact that $\lambda_i \geq \frac{1}{2}$ for all i implies that the tangent majorant curvatures are uniformly bounded from below in all directions. Hence, (C5.1) holds.

In the example, $\{\boldsymbol{\theta}^i\}$ does not converge even though it is a bounded sequence and the tangent majorant curvatures are bounded from below. These conditions are insufficient to prevent the increments $\{\boldsymbol{\theta}^{i+1} - \boldsymbol{\theta}^i\}$ from becoming asymptotically tangential. Thus, in the limit, the $\{\boldsymbol{\theta}^i\}$ move in a circular path about the origin. This observation suggests that the convergence of $\{\boldsymbol{\theta}^i\}$ might be ensured by preventing this kind of asymptotically tangential behavior. Preventing this behavior seems to require, at minimum, that *both* an upper and lower bound exist on the curvatures of the tangent majorants $\phi^i(\cdot; \boldsymbol{\theta}^i)$. In Example 6.7.5, asymptotic tangentiality comes about precisely because the $\{\lambda_i\}$, which determine the tangent majorant curvatures, become unbounded. We may explore these ideas further in future work.

6.8 Conclusion

We have generalized the analysis of MM given in [33] by relaxing the twice differentiability and convexity assumptions on the cost function Φ and tangent majorants $\{\phi^i(\cdot; \boldsymbol{\theta}^i)\}$. The analysis applies to any convex feasible set and allows the tangent majorant domains to be strict subsets of this set. We have also considered a more general version of the block alternation technique. Our analysis examined the asymptotic properties of such algorithms as well as the tendency of an MM algorithm to be captured in basin-like regions

in the graph of Φ.

The asymptotic analysis addressed separately the cases where block alternation was used and where block alternation was not used. When block alternation is not used, asymptotic stationarity is assured if the sequence $\{\phi^i(\cdot;\cdot)\}$ consists of majorant generators chosen from a finite set (condition (C2)) and, moreover, that each $\phi^i(\cdot;\cdot)$ satisfy continuity conditions described in (C3). Alternatively, one can require that the tangent majorant curvatures be uniformly upper bounded in the manner described in (C6). The tangent majorants need not be convex. In the block alternating case, we dropped (C2) and added (C4) and (C5). However, these modified assumptions are no stronger than those considered previously in [33]. In these various cases, convergence results followed (see Theorem 6.4.5) under standard discreteness assumptions on the problem's stationary points.

In addition to the generality of our assumptions, our asymptotic analysis is structured in a way that imparts several additional theoretical insights, as compared to previous literature. Firstly, we found in Theorem 6.4.1 that, in the non-block alternating case, conditions like (C5.1), or even its weaker version (C5), need not be known to hold *a priori* to establish asymptotic stationarity. Conversely, in [33, Condition 5], a strong convexity condition similar to (C5.1) is incorporated throughout the analysis. Secondly, our analysis shows that asymptotic stationarity can be assured by curvature conditions like (C6), and not merely continuity conditions like (C3). Conversely, previous convergence analyses are based mainly on continuity conditions. An advantage of the curvature conditions, in the non-block alternating case at least, is that they allow more flexible iteration-dependent behavior to be used. Note that, in Theorem 6.4.1, the continuity condition (C3) is accompanied by (C2). No such restriction is necessary when (C6) holds. Thirdly, our analysis clarifies when asymptotic stationarity can occur – for algorithms using iteration-dependent majorant generators – even when convergence in norm may not. By contrast, the line of

proof in [33] establishes convergence first, before investigating whether the limit is stationary.

In Section 6.5, we proved the capture property of MM algorithms that use connected (e.g., convex) tangent majorants, which is the most common practice. An implication of this property is that global minimizers attract the iterates over essentially the largest possible neighborhood. The capture property also makes MM a useful instrument in non-convex minimization strategies that rely on basin-probing steps. A negative implication of the capture property is that MM might have a higher tendency to get stuck in local minima than other algorithms. To mitigate this, one must run the algorithm from multiple initial points, and this can be computationally expensive.

Finally, we made a preliminary examination, in Section 6.7, of conditions that ensure the convergence of MM sequences in the presence of non-isolated stationary points. For 1D problems, convergence is assured provided that a uniform lower bound exists on the tangent majorant curvatures. However, for higher dimensional problems, this condition is insufficient, as was shown by way of a 2D example. Conditions that ensure convergence in higher dimensions were conjectured and may be considered further in future work.

CHAPTER 7

Summary and Future Work

7.1 Summary

We have conducted a comparison of the average tumor quantification accuracies of different motion correction schemes (Section 4.5.2). The comparison included a statistically principled estimation method (JEDM), proposed by us, and some intuitive, but less statistically principled, gate-wise reconstruction approaches, ones similar to those considered by other investigators. The experimentation used simulated data derived from real patient anatomy and motion. Moreover, we employed a regularization design scheme that, by excluding the known region of the lesion, allowed each method to exhibit better quantification than it otherwise would.

In this first round of tests, the JEDM method demonstrated the best average lesion quantification accuracy. However, there are still issues to be addressed. Apart from a desire to test the methods on yet more realistic measurement data, questions remain as to the the sensitivity of JEDM (and other methods) to the presence of local minima. Because of these issues, we do not expect the other reconstruction methods considered in Chapter 4 to be immediately dismissed. Even if JEDM continues to show superior performance in subsequent tests, the semi-statistical method, FWPR-PLC showed reasonable enough performance that it may provide a good way to initialize JEDM reconstructions.

With this in mind, we proceeded to address the minimization sub-problems required in all of the various methods compared in Chapter 4 and associated computational issues. All of these sub-problems involve interpolation operations that impose some degree or other of computational expense. In Chapter 5, we proposed various methods by which these operations could be accelerated. Foremost among these methods was a table-lookup approach wherein part of the interpolation computation was replaced by approximate, pre-tabulated quantities. This method reduced, by a certain factor, the size of the image blocks that each interpolation operation needs to process. A reduction in CPU time by approximately the same factor was observed in various computational tests.

The benefits of this method were the most pronounced in SSD registration problems. There, interpolation operations introduce the most overhead. In the minimization sub-problems associated with JEDM, the impact of these methods was appreciable, but somewhat less dramatic due to the overhead associated with forward projection operations. This was one motivation for considering MM algorithms. The MM algorithm design methodology can allow one to replace complicated parts of the given objective function with simpler ones, thereby simplifying the cost reduction steps that one would otherwise face.

Ideas for block alternating MM algorithms for the JEDM problem were conceptualized early on (some of them are discussed in Sections 6.6.1 and 6.6.2), but were of a form not covered by previous convergence analyses. Thus, we embarked upon efforts to generalize these analyses in Chapter 6. The analysis treats a framework that is simultaneously flexible in terms of the form of the tangent majorants, the form of their domains, the form of the cost function, and the form of the feasible set.

The analysis also characterized the local region of convergence for the broad family of MM algorithms employing connected tangent majorants. We believe that this is the first analysis in the MM literature to do so. We found that such algorithms will have

larger regions of convergence than traditional algorithms, namely the region covers the entire basin-like region surrounding a stationary point. This property can have negative consequences – namely connected tangent majorants may have more of a tendency to get stuck at sub-optimal local minima – but also positive potential applications to various non-convex minimization strategies.

7.2 Future Work

- We have tested the various motion corrected reconstruction methods of Chapter 4 in fairly realistic simulations. However, there are higher levels of realism to look at. Future simulations will look at the impact of modeling error, both on the part of the projection model and the deformation model. It is also desirable, of course, to test these methods on real data.

- The sensitivity of the different motion correction methods to the presence of local minima (and remedies for it) remains an issue. In tests not shown in Chapter 4, we have found that JEDM performs better when initialized with a FWPR-PLC reconstruction than when initialized from uniform parameter values. We will want to test the robustness of this approach. Various multi-resolution methods have been proposed in the conventional image registration literature (e.g., [99, 92]). These might be adapted to the projection domain deformation cost minimizations required in JEDM.

- We have carried out reconstructions so far using a fixed regularization penalty parameter β. It is desirable to examine how the lesion quantification performance of the motion-correction algorithms is influenced by varying β.

- Because effective motion correction methods considered in this dissertation involve gated data, there are N_{gates} as much data to process, as compared to the ungated case.

The benefits of motion correction are therefore bought at N_{gates} times more computation. It may be possible to eliminate, or at least reduce, this factor by employing ordered subsets or other incremental gradient minimization schemes. The partitioning of the projection data into gates opens the possibility of incrementalizing the data processing across gates in addition to the typical approach of incrementally processing subsets of projection angles. General experience with incremental methods suggests that this should accelerate convergence by a factor of N_{gates}, offsetting the increased size of the measured data.

- To date, all work on motion-corrected PET, that we know of, has addressed the motion of the activity image, but ignored the corresponding motion of the attenuation image. The modeling approach of Section 4.2 can easily be extended, so that both $\boldsymbol{\mu}$ and $\boldsymbol{\lambda}$ transform from gate to gate according to deformation parameters $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_{N_{\text{gates}}})$.

- Many of the performance trends observed for the frame-wise methods were consistent with the fact that these methods estimated each $\boldsymbol{\alpha}_t$ based only on the projection data acquired in the corresponding gate. This is in contrast to JEDM in which all $\boldsymbol{\alpha}_t$ were estimated based on the full set of projection data. This difference means that the frame-wise methods estimate motion with higher variance and this in turn has an impact on lesion recovery. However, some investigators (e.g., [51]) have proposed that frame-wise methods should use inter-gate regularization to couple the $\hat{\boldsymbol{\alpha}}_t$ to the entire set $\{\boldsymbol{y}_t\}$. This would lower the variance of the $\hat{\boldsymbol{\alpha}}_t$ and might narrow the performance gap between JEDM and the frame-wise methods.

- There is still some question as to whether the performance trends of FWPR-PA in Section 4.5.2 were due more to invertibility issues in (4.14) than to actual registration

error. It may be possible to test this in a simulation based on $\boldsymbol{\lambda}_0 = \boldsymbol{\mathcal{T}}(\boldsymbol{\alpha}_t)\boldsymbol{\lambda}_t$, $t = 1, \ldots, N_{\text{gates}} - 1$ instead of on (4.3).

- The timing experiments in Section 5.4.2 showed that the sequentiality of memory access has a significant effect on the CPU time of deformation options. A wise strategy may be to pre-sort the deformation map before conducting the series of interpolation operations, so as to make memory access more sequential.

- Ideas for MM algorithms for JEDM were proposed in Sections 6.6.2 and 6.6.1. These are yet to be tested.

- We believe that we can generalize the analysis of MM algorithms in Chapter 6 still further to accomodate non-convex feasible sets and inexact minimization of the tangent majorants. This was done in [72], but only for the case of a single, iteration-independent majorant generator.

- In Chapter 6, our analysis requires the tangent majorants to satisfy the majorization property only over its own domain $D(\bar{\boldsymbol{\theta}})$, which can be a subset of the overall feasible set. This looser majorization requirement should allow tangent majorants to more accurately approximate the cost function and so achieve faster convergence. This raises questions as to how to design $D(\bar{\boldsymbol{\theta}})$.

APPENDIX

APPENDIX A

Notes and Extended Discussions on MM Topics

Note A.1 (Tangent majorants vs. true tangent majorants) A sequence $\{\boldsymbol{\theta}^i\}$ satisfies (6.5) and (6.6) even if global constants are added to the $\{\phi^i(\cdot;\boldsymbol{\theta}^i)\}$. Thus, the behavior of MM sequences can be studied irrespective of whether the $\{\phi^i(\cdot;\boldsymbol{\theta}^i)\}$ are true tangent majorants, or merely tangent majorants. The distinction becomes important, however, when deriving tangent majorants by composition of functions. For example, suppose $f : \mathbb{R} \to \mathbb{R}$ and $g : \mathbb{R}^p \to \mathbb{R}$ with f monotone increasing and suppose $q(\cdot;\bar{\boldsymbol{\theta}}) \overset{\bar{\boldsymbol{\theta}}}{\underset{\Theta}{\succeq}} g(\cdot)$. Then the montonicity of f implies

$$Q(\cdot;\bar{\boldsymbol{\theta}}) \stackrel{\triangle}{=} f(q(\cdot;\bar{\boldsymbol{\theta}})) \overset{\bar{\boldsymbol{\theta}}}{\underset{\Theta}{\succeq}} f(g(\cdot,\bar{\boldsymbol{\theta}}_{\bar{\mathcal{S}}})). \tag{A.1}$$

Thus, deriving the tangent majorant $Q(\cdot;\bar{\boldsymbol{\theta}})$ for $f(g(\cdot,\bar{\boldsymbol{\theta}}_{\bar{\mathcal{S}}}))$ is accomplished by finding a true tangent majorant $q(\cdot;\bar{\boldsymbol{\theta}})$ to g. Note that (A.1) is not necessarily true if $q(\cdot;\bar{\boldsymbol{\theta}})$ is a tangent majorant to g, but not a true tangent majorant.

Note A.2 (Implied derivative matching) Often, MM majorant generators are designed so that, for all i and $\bar{\boldsymbol{\theta}} \in \Theta$

$$\eta^i(\bar{\boldsymbol{\theta}};\boldsymbol{\xi}) = \left\langle \nabla_{\mathcal{S}^i}\Phi(\bar{\boldsymbol{\theta}}),\, \boldsymbol{\xi} - \bar{\boldsymbol{\theta}}_{\mathcal{S}^i} \right\rangle \qquad \forall \boldsymbol{\xi} \in \Theta_{\mathcal{S}^i}. \tag{A.2}$$

That is, the feasible directional derivatives of the tangent majorants $\{\phi^i(\cdot;\boldsymbol{\theta}^i)\}$ should match the cost function at any possible expansion point. This is a stronger version of (6.14), which requires that derivatives match only at the iterates $\{\boldsymbol{\theta}^i\}$.

It is interesting to note that Equation (A.2) is an implied condition whenever $\text{aff}(D^i(\bar{\boldsymbol{\theta}})) = \text{aff}(\Theta_{\mathcal{S}^i})$ and $\bar{\boldsymbol{\theta}}_{\mathcal{S}^i} \in \text{ri}(D^i(\bar{\boldsymbol{\theta}}))$. This follows directly from the alternative definition of a majorant generator (6.4). The definition requires that the difference $\phi^i(\cdot;\boldsymbol{\theta}^i) - \Phi(\cdot)$ be minimized over $D^i(\boldsymbol{\theta}^i)$ at $\boldsymbol{\theta}^i$. By standard necessary optimality conditions for interior points, the derivatives of this difference must vanish in all feasible directions, i.e., all $\boldsymbol{\xi} - \bar{\boldsymbol{\theta}}_{\mathcal{S}^i}$ such that $\boldsymbol{\xi} \in \text{aff}(D^i(\bar{\boldsymbol{\theta}})) = \text{aff}(\Theta_{\mathcal{S}^i})$. Since $\Theta_{\mathcal{S}^i} \subset \text{aff}(\Theta_{\mathcal{S}^i})$, the derivatives must vanish in all directions $\boldsymbol{\xi} - \bar{\boldsymbol{\theta}}_{\mathcal{S}^i}$ such that $\boldsymbol{\xi} \in \Theta_{\mathcal{S}^i}$. This is precisely (A.2).

In some literature (e.g., Footnote 6 in [27]), it is claimed incorrectly that (A.2) is implied at *all* points in $D^i(\bar{\theta})$ and not just in its relative interior. However, since the above arguments rely on optimality conditions for interior points, one cannot expect it to remain true at boundary points. This is illustrated in the following simple example.

Example A.0.1 (Derivatives need not match at the boundaries) Consider the 1D cost function $\Phi(\theta) = \exp(-\theta) + \theta$ on $\Theta = \{\theta \geq 0\}$ and $\phi(\theta; 0) = \theta + 1$ with domain $D(0) = \Theta$ (see also Figure A.1). One can verify that $\phi(\cdot; 0)$ satisfies the requisite property (6.3) of a tangent majorant at expansion point $\bar{\theta} = 0$. However, the feasible directional derivatives of Φ and $\phi(\cdot; 0)$ do not match at the expansion point.

Furthermore, minimizing $\phi(\cdot; 0)$ does not produce a new expansion point. Any MM algorithm that uses $\phi(\cdot; 0)$ gets stuck at the non-stationary point $\bar{\theta} = 0$. It is for this reason that derivative matching must be directly enforced by the algorithm designer (e.g., through (R2) or (A.2)) to avoid degenerate behavior.

Note A.3 (Imposed gradient matching in non-solid sets) In existing MM literature, the stronger condition (R2.1) is customarily used to ensure (R2). Condition (R2.1) requires that the derivatives of the cost function and tangent majorants match in all directions, not just feasible ones. This might be excessive when $\Theta_{\mathcal{S}^i}$ is not a *solid* subset of $\mathbb{R}_{\mathcal{S}^i}$ (i.e., when $\mathbb{R}_{\mathcal{S}^i} \neq \text{aff}(\Theta_{\mathcal{S}^i})$), in which case the space of feasible directions in $\Theta_{\mathcal{S}^i}$ is smaller than $\mathbb{R}_{\mathcal{S}^i}$.

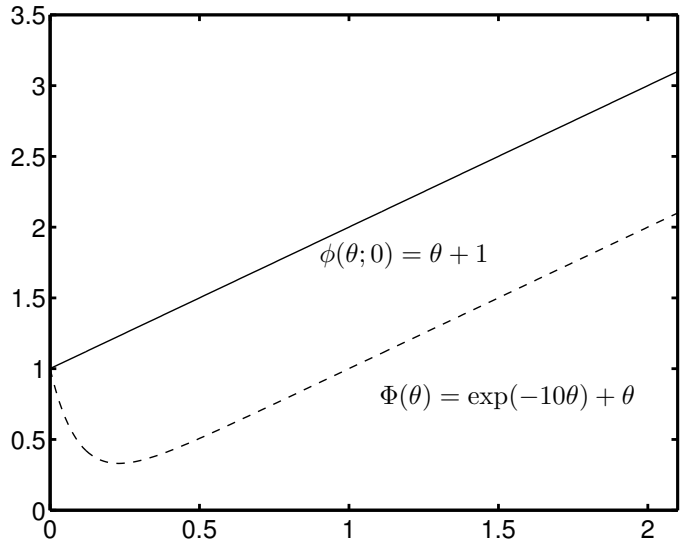

Figure A.1: Illustration that, when the expansion point of a tangent majorant is at the boundary of its domain, its derivatives need not match the cost function at that point. It also shows how this can cause a corresponding MM algorithm to prematurely stop at a non-stationary point.

A rudimentary example is obtained from Example 6.2.1 with $\mathcal{S} = 1$. In $\Theta_{\mathcal{S}} = \Theta_1$, the space of feasible directions at any point is a one-dimensional subset of $\mathbb{R}_{\mathcal{S}} = \mathbb{R}^3$. A less rudimentary example is encountered in applications to Positron Emission Tomography (PET) reconstruction. In PET, the sum of the measured data is, with high probability approximately equal to its mean. This motivates hyperplane (i.e., non-solid) constraints in addition to the usual positivity constraints. Hyperplane constraints were also imposed for this problem in [6], but with somewhat different motivation.

When dealing with non-solid $\Theta_{\mathcal{S}^i}$, it may be advantageous to impose (A.2). This condition is weaker than (R2.1) but stronger and perhaps more easy to verify, than (R2).

Note A.4 (EM as a special case of MM) As discussed in Section 6.1, the family of EM algorithms is a prominent special case of MM algorithms for minimizing the negative loglikelihood $\Phi(\boldsymbol{\theta}) = -\log \mathcal{P}(Y = y|\boldsymbol{\theta})$ of a measurement vector y. One develops an EM algorithm by inventing a joint distribution $\mathcal{P}(Y, Z|\boldsymbol{\theta})$ whose marginal with respect to Y coincides with the given measurement likelihood $\mathcal{P}(Y|\boldsymbol{\theta})$. In the EM literature, the

artifical random vector Z is called a *complete data* vector. An MM algorithm is then constructed using the majorant generator,

$$\phi(\boldsymbol{\theta}; \bar{\boldsymbol{\theta}}) = \Phi(\boldsymbol{\theta}) + \mathrm{KL}(\mathcal{P}(Z|Y = y, \bar{\boldsymbol{\theta}}), \mathcal{P}(Z|Y = y, \boldsymbol{\theta})). \tag{A.3}$$

Often a joint distribution $\mathcal{P}(Y, Z|\boldsymbol{\theta})$ is constructed by finding a complete data vector Z such that $Y = h(Z)$, where h is a deterministic function. This was the family of constructions considered in [25], but generalizations have been proposed, e.g., [32, 60, 69]. Note that the converse, i.e., constructing complete data so that $Z = h(Y)$ is generally ineffective since (A.3) then reduces to $\phi(\boldsymbol{\theta}; \bar{\boldsymbol{\theta}}) = \Phi(\boldsymbol{\theta})$.

Although it is a classical example of MM, the EM design methodology has frequently resulted in slow algorithms (as discussed, for example, in [31]). This has motivated certain investigators (e.g., [27, 28]) to depart from the EM framework and look for more general types of majorant generators.

The kind of discontinuities exhibited in Example 6.4.2 can arise in EM majorant generators (A.3) because (see (2.1)), $\mathrm{KL}(f, g)$ can be discontinuous wherever $f = g = 0$. This occurs, for example, in the emission tomography EM application of [88]. When such discontinuities exist, the limit points of $\{\boldsymbol{\theta}^i\}$ may still be stationary. However, very situation-specific analysis may be required to establish this. For the EM algorithm of [88], such an analysis can be found, for example, in [13]. Of course, if the complete data vector is chosen so that $\mathcal{P}(Z|Y = y, \boldsymbol{\theta}) > 0$ for all $\boldsymbol{\theta}$, then such discontinuities are not present.

Note A.5 (The dimension of a generalized capture basin) Any generalized capture basin G must have the same dimension as Θ, in the sense that $\mathrm{aff}(G) = \mathrm{aff}(\Theta)$. Thus, for example, if $\Theta = \mathbb{R}^2$, no line segment inside Θ can constitute a generalized capture basin. We show this by contradiction.

Accordingly, assume instead that $\mathrm{aff}(G)$ is a strict subset of $\mathrm{aff}(\Theta)$. A consequence

of this is that $\Theta \setminus \text{aff}(G)$ is non-empty and contains a point that we denote $\boldsymbol{\xi}$. This must be true, since otherwise we would have $\Theta \subset \text{aff}(G)$, from which it would follow that $\text{aff}(\Theta) \subset \text{aff}(G)$, contradicting the assumption that $\text{aff}(G)$ is a strict subset of $\text{aff}(\Theta)$. Now fix any point $\boldsymbol{\psi} \in G$ and any $\alpha \in (0,\, 1)$. Define

$$\boldsymbol{\theta}_\alpha \stackrel{\triangle}{=} \alpha\boldsymbol{\psi} + (1-\alpha)\boldsymbol{\xi}. \tag{A.4}$$

Then $\boldsymbol{\theta}_\alpha \in \Theta$, due to the convexity of Θ, but $\boldsymbol{\theta}_\alpha \notin G$. If $\boldsymbol{\theta}_\alpha$ were an element of G, then rearranging (A.4) as follows,

$$\boldsymbol{\xi} = \frac{1}{1-\alpha}\boldsymbol{\theta}_\alpha - \frac{\alpha}{1-\alpha}\boldsymbol{\psi},$$

yields $\boldsymbol{\xi}$ as an affine combination of elements of G. However, this is impossible, since $\boldsymbol{\xi}$ was drawn from the set $\Theta \setminus \text{aff}(G)$. We conclude that $\boldsymbol{\theta}_\alpha \in \Theta \setminus G$ for any fixed $\alpha \in (0,\, 1)$.

Now, letting α tend to 1 causes $\boldsymbol{\theta}_\alpha$ to tend to $\boldsymbol{\psi}$ and establishes that $\boldsymbol{\psi} \in \text{cl}(\Theta \setminus G)$. Also, since $\boldsymbol{\psi}$ was drawn from G, we have that $\boldsymbol{\psi} \in \text{cl}(G)$. It follows that $\boldsymbol{\psi} \in \text{cl}(G) \cap \text{cl}(\Theta \setminus G)$ and, because $\boldsymbol{\psi}$ was arbitrary, that $G = \text{cl}(G) \cap \text{cl}(\Theta \setminus G)$. However, no generalized capture basin G can satisfy $G = \text{cl}(G) \cap \text{cl}(\Theta \setminus G)$ because (6.28) would be violated by taking any $\boldsymbol{\theta} \in G$ and letting $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta}$. This establishes a contradiction and proves that $\text{aff}(G) = \text{aff}(\Theta)$.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] S. Ahn and J. A. Fessler. Globally convergent image reconstruction for emission tomography using relaxed ordered subsets algorithms. *IEEE Trans. Med. Imag.*, 22(5):613–26, May 2003.

[2] S. Ahn, J. A. Fessler, D. Blatt, and A. O. Hero. Convergent incremental optimization transfer algorithms: Application to tomography. *IEEE Trans. Med. Imag.*, 25(3), March 2006. To appear.

[3] N. M. Alpert, J. F. Bradshaw, D. Kennedy, and J. A. Correia. The pincipal axes transformation – A method for image registration. *J. Nuc. Med.*, 31(10):1717–1722, 1990.

[4] E. Asma and R. M. Leahy. Theoretical comparison of uniform resolution penalized likelihood dynamic PET images with multiframe penalized likelihood static PET images. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 7, pages 4217–20, 2004.

[5] S. Basu and Y. Bresler. $O(N^3 \log N)$ backprojection algorithm for the 3-D Radon transform. *IEEE Trans. Med. Imag.*, 21(2):76–88, February 2002.

[6] A. Ben-Tal, T. Margalit, and A. Nemirovski. The ordered subsets mirror descent optimization method with applications to tomography. *SIAM J. Optim.*, 12(1):79–108, 2001.

[7] D. P. Bertsekas. *Nonlinear programming*. Athena Scientific, Belmont, 2 edition, 1999.

[8] A. Blake and A. Zisserman. *Visual reconstruction*. MIT Press, Cambridge, MA, 1987.

[9] F. Bookstein and W. D. K. Green. A feature space for edgels in images with landmarks. *J. Math. Im. Vision*, 3(3):231–61, September 1993.

[10] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge, UK, 2004.

[11] C. Broit. *Optimal registration of deformed images*. PhD thesis, University of Pennsylvania, 1981.

[12] J. A. Browne and A. R. De Pierro. A row-action alternative to the EM algorithm for maximizing likelihoods in emission tomography. *IEEE Trans. Med. Imag.*, 15(5):687–99, October 1996.

[13] C. Byrne. Likelihood maximization for list-mode emission tomographic image reconstruction. *IEEE Tr. Med. Im.*, 20(10):1084–92, October 2001.

[14] C. L. Byrne. Accelerating the EMML algorithm and related iterative algorithms by rescaled block-iterative methods. *IEEE Trans. Im. Proc.*, 7(1):100–9, January 1998.

[15] A. Chatziioannou, J. Qi, A. Annala, A. Moore, R. M. Leahy, and S. R. Cherry. Comparison of 3D MAP and FBP algorithms for image reconstruction in microPET. *IEEE Trans. Med. Imag.*, 19(5):507–12, May 2000.

[16] N. H. Clinthorne, J. A. Fessler, G. D. Hutchins, and W. L. Rogers. Joint maximum likelihood estimation of emission and attenuation densities in PET. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 3, pages 1927–32, 1991.

[17] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, and G. Marchal. Automated multimodality image registration using information theory. In Y. Bizais, C. Barillot, and R. D. Paola, editors, *Information Processing in Medical Im.*, pages 263–74. Kluwer, Dordrect, 1995.

[18] C. Comtat, P. E. Kinahan, J. A. Fessler, T. Beyer, D. W. Townsend, M. Defrise, and C. Michel. Clinically feasible reconstruction of 3d whole-body PET/CT data using blurred anatomical labels. *Phys. Med. Biol.*, 47(1):1–20, January 2002.

[19] B. M. Dawant. Non-rigid registration of medical images: Purpose and methods, a short survey. In *Proc. IEEE Intl. Symp. Biomed. Imag.*, pages 465–8, 2002.

[20] B. De Man and S. Basu. Distance-driven projection and backprojection. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 3, pages 1477–80, 2002.

[21] A R De Pierro. On the relation between the ISRA and the EM algorithm for positron emission tomography. *IEEE Tr. Med. Im.*, 12(2):328–33, June 1993.

[22] A R De Pierro. A modified expectation maximization algorithm for penalized likelihood estimation in emission tomography. *IEEE Tr. Med. Im.*, 14(1):132–137, March 1995.

[23] A R De Pierro. On the convergence of an EM-type algorithm for penalized likelihood estimation in emission tomography. *IEEE Tr. Med. Im.*, 14(4):762–5, December 1995.

[24] A. R. De Pierro and M. E. B. Yamagishi. Fast EM-like methods for maximum 'a posteriori' estimates in emission tomography. *IEEE Trans. Med. Imag.*, 20(4):280–8, April 2001.

[25] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Stat. Soc. Ser. B*, 39(1):1–38, 1977.

[26] M. L. Egger, C. Joseph, and C. Morel. Incremental beamwise backprojection using geometrical symmetries for 3D PET reconstruction in a cylindrical scanner geometry. *Phys. Med. Biol.*, 43(10):3009–24, October 1998.

[27] H. Erdoğan and J. A. Fessler. Monotonic algorithms for transmission tomography. *IEEE Tr. Med. Im.*, 18(9):801–14, September 1999.

[28] Hakan Erdoğan. *Statistical image reconstruction algorithms using paraboloidal surrogates for PET transmission scans*. PhD thesis, Univ. of Michigan, Ann Arbor, MI, 48109-2122, Ann Arbor, MI., July 1999.

[29] J. A. Fessler and H. Erdoğan. A paraboloidal surrogates algorithm for convergent penalized-likelihood emission image reconstruction. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 2, pages 1132–5, 1998.

[30] J. A. Fessler, N. H. Clinthorne, and W. L. Rogers. Regularized emission image reconstruction using imperfect side information. *IEEE Trans. Nuc. Sci.*, 39(5):1464–71, October 1992.

[31] J. A. Fessler, N. H. Clinthorne, and W. L. Rogers. On complete data spaces for PET reconstruction algorithms. *IEEE Trans. Nuc. Sci.*, 40(4):1055–61, August 1993.

[32] J. A. Fessler and A. O. Hero. Space-alternating generalized expectation-maximization algorithm. *IEEE Tr. Sig. Proc.*, 42(10):2664–77, October 1994.

[33] J. A. Fessler and A. O. Hero. Penalized maximum-likelihood image reconstruction using space-alternating generalized EM algorithms. *IEEE Tr. Im. Proc.*, 4(10):1417–29, October 1995.

[34] B. Fischer and J. Modersitzki. Fast inversion of matrices arising in image processing. *Numer. Algorithms*, 22(1):1–11, 1999.

[35] B. Fischer and J. Modersitzki. Curvature based image registration. *J. Math. Im. Vision*, 18(1):81–5, January 2003.

[36] R. R. Fulton and S. R. Meikle. Reconstruction of projection data corrupted by rigid or non-rigid motion. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, 2005.

[37] D. R. Gilland, B. A. Mair, J. E. Bowsher, and R. J. Jaszczak. Simultaneous reconstruction and motion estimation for gated cardiac ECT. *IEEE Tr. Nuc. Sci.*, 49(5):2344–49, October 2002.

[38] R. C. Gonzalez and R. C. Woods. *Digital image processing*. Addison-Wesley, Reading, Mass., 1992.

[39] P. J. Green. Bayesian reconstructions from emission tomography data using a modified EM algorithm. *IEEE Trans. Med. Imag.*, 9(1):84–93, March 1990.

[40] T. J. Hebert and S. S. Gopal. The GEM MAP algorithm with 3-D SPECT system response. *IEEE Trans. Med. Imag.*, 11(1):81–90, March 1992.

[41] T. J. Hebert and R. Leahy. Statistic-based MAP image reconstruction from Poisson data using Gibbs priors. *IEEE Trans. Sig. Proc.*, 40(9):2290–303, September 1992.

[42] W. J. Heiser. Convergent computation by iterative majorization: theory and applications in multidimensional data analysis. In W. J. Krzanowski, editor, *Recent Advances in Descriptive Multivariate Analysis*, Royal Statistical Society Lecture Note Series. Oxford University Press, New York, 1995.

[43] P. J. Huber. *Robust statistics*. Wiley, New York, 1981.

[44] H. M. Hudson and R. S. Larkin. Accelerated image reconstruction using ordered subsets of projection data. *IEEE Trans. Med. Imag.*, 13(4):601–9, December 1994.

[45] B. F. Hutton, A. Z. Kyme, Y. H. Lau, D. W. Skerrett, and R. R. Fulton. A hybrid 3-d reconstruction/registration algorithm for correction of head motion in emission tomography. *IEEE Trans. Nuc. Sci.*, 49(1-1):188–94, February 2002.

[46] M. Jacobson and J. A. Fessler. Simultaneous estimation of attenuation and activity images using optimization transfer. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 4, pages 2085–9, 2001.

[47] M. W. Jacobson and J. A. Fessler. Joint estimation of image and deformation parameters in motion-corrected PET. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 5, pages 3290–4, 2003.

[48] M. W. Jacobson and J. A. Fessler. Fast interpolation operations in non-rigid image registration. In *Proc. SPIE 5747, Medical Imaging 2005: Image Proc.*, pages 764–74, 2005.

[49] J. Kim. *Intensity based image registration using robust similarity measure and constrained optimization: applications for radiation therapy*. PhD thesis, Univ. of Michigan, Ann Arbor, MI, 48109-2122, Ann Arbor, MI, 2004.

[50] P. E. Kinahan and J. G. Rogers. Analytic 3D image reconstruction using all detected events. *IEEE Trans. Nuc. Sci.*, 36(1):964–8, February 1989.

[51] G. J. Klein, B. W. Reutter, and R. H. Huesman. Four-dimensional affine registration models for respiratory-gated PET. *IEEE Trans. Nuc. Sci.*, 48(3):756–60, June 2001.

[52] S. Klein, M. Staring, and J. P. Pluim. Comparison of gradient approximation techniques for optimisation of mutual information in nonrigid registration. In *Proc. SPIE 5747, Medical Imaging 2005: Image Proc.*, pages 192–203, 2005.

[53] J. Kybic, P. Thevenaz, A. Nirkko, and M. Unser. Unwarping of unidirectionally distorted EPI images. *IEEE Trans. Med. Imag.*, 19(2):80–93, February 2000.

[54] J. Kybic and M. Unser. Fast parametric elastic image registration. *IEEE Trans. Im. Proc.*, 12(11):1427–42, November 2003.

[55] D. S. Lalush and B. M. W. Tsui. Spatially-adaptive Gibbs priors for MAP-EM SPECT reconstruction. *J. Nuc. Med. (Abs. Book)*, 36(5):8p, May 1995.

[56] K. Lange. A gradient algorithm locally equivalent to the EM Algorithm. *J. Royal Stat. Soc. Ser. B*, 57(2):425–37, 1995.

[57] K. Lange and R. Carson. EM reconstruction algorithms for emission and transmission tomography. *J. Comp. Assisted Tomo.*, 8(2):306–16, April 1984.

[58] K. Lange, D. R. Hunter, and I. Yang. Optimization transfer using surrogate objective functions. *J. Computational and Graphical Stat.*, 9(1):1–20, March 2000.

[59] M. J. Ledesma-Carbayo, J. Kybic, M. Desco, A. Santos, M. Suhling, P. Hunziker, and M. Unser. Spatio-temporal nonrigid registration for ultrasound cardiac motion estimation. *IEEE Trans. Med. Imag.*, 24(9):1113–26, September 2005.

[60] C. H. Liu and Y. N. Wu. Parameter expansion scheme for data augmentation. *J. Am. Stat. Ass.*, 94(448):1264–74, December 1999.

[61] L. Livieratos, L. Stegger, P. M. Bloomfield, K. Schafers, D. L. Bailey, and P. G. Camici. Rigid-body transformation of list-mode projection data for respiratory motion correction in cardiac pet. *Phys. Med. Biol.*, 50(14):3313–22, July 2005.

[62] W. Lu and T. R. Mackie. Tomographic motion detection and correction directly in sinogram space. *Phys. Med. Biol.*, 47(8):1267–84, April 2002.

[63] A. E. Lujan, E. W. Larsen, J. M. Balter, and R. K. Ten Haken. A method for incorporating organ motion due to breathing into 3D dose calculations. *Med. Phys.*, 26(5):715–20, May 1999.

[64] Z. Q. Luo and P. Tseng. On the convergence of the coordinate descent method for convex differentiable minimization. *J. Optim. Theory Appl.*, 72(1):7–35, January 1992.

[65] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *IEEE Trans. Med. Imag.*, 16(2):187–98, April 1997.

[66] B. A. Mair, D. R. Gilland, and Zixiong Cao. Simultaneous motion estimation and image reconstruction from gated data. In *Proc. IEEE Intl. Symp. Biomedical Imaging*, pages 661–64, 2002.

[67] M. Á. Martín-Fernández and E. Mu noz Moreno. Articulated registration: elastic registration based on a wire-model. In *Proc. SPIE 5747, Medical Imaging 2005: Image Proc.*, pages 182–90, 2005.

[68] D. Mattes, D. R. Haynor, H. Vesselle, T. K. Lewellen, and W. Eubank. PET-CT image registration in the chest using free-form deformations. *IEEE Trans. Med. Imag.*, 22(1):120–8, January 2003.

[69] X. L. Meng and D. van Dyk. The EM algorithm - An old folk song sung to a fast new tune. *J. Royal Stat. Soc. Ser. B*, 59(3):511–67, 1997.

[70] J. Modersitzki. *Numerical methods for image registration*. Oxford, 2003.

[71] R. Narayanan, J. A. Fessler, H. Park, and C. R. Meyer. Diffeomorphic nonlinear transformations: A local parametric approach for image registration. In *Information Processing in Medical Im.*, pages 174–85, 2005.

[72] D. Nettleton. Convergence properties of the EM algorithm in constrained parameter spaces. *The Canadian Journal of Statistics*, 27(3):639–48, 1999.

[73] V. Noblet, C. Heinrich, F. Heitz, and J. Armspach. 3-D deformable image registration: a topology preservation scheme based on hierarchical deformation models and interval analysis optimization. *IEEE Tr. Med. Im.*, 14(5):553–65, May 2005.

[74] J. M. Ollinger and A. Goggin. Maximum likelihood reconstruction in fully 3D PET via the SAGE algorithm. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 3, pages 1594–8, 1996.

[75] J. M. Ortega and W. C. Rheinboldt. *Iterative solution of nonlinear equations in several variables*. Academic, New York, 1970.

[76] A. M. Ostrowski. *Solution of equations in Euclidian and Banach spaces*. Academic, New York, 1973.

[77] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. Mutual-information-based registration of medical images: a survey. *IEEE Trans. Med. Imag.*, 22(8):986–1004, August 2003.

[78] J. Qi and R. H. Huesman. Theoretical study of lesion detectability of MAP reconstruction using computer observers. *IEEE Tr. Med. Im.*, 20(8):815–22, August 2001.

[79] F. Qiao, T. Pan, J. W. Clark, and O. R. Mawlawi. Compensating respiratory motion in PET image reconstruction using 4D PET/CT. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, 2005.

[80] A. Rahmim, P. Bloomfield, S. Houle, M. Lenox, C. Michel, K. R. Buckley, T. J. Ruth, and V. Sossi. Motion compensation in histogram-mode and list-mode em reconstructions: beyond the event-driven approach. *IEEE Tr. Nuc. Sci.*, 51(5):2588–96, October 2004.

[81] G. K. Rohde, A. Aldroubi, and B. M. Dawant. The adaptive basis algorithm for nonrigid image registration. In *Proc. SPIE 4684, Medical Imaging 2002: Image Proc.*, volume 2, pages 933–43, 2002.

[82] G. K. Rohde, C. A. Berenstein, and D. M. Healy Jr. Measuring image similarity in the presence of noise. In *Proc. SPIE 5747, Medical Imaging 2005: Image Proc.*, pages 132–43, 2005.

[83] T. Rohlfing, C. R. Maurer, D. A. Bluemke, and M. A. Jacobs. Volume-preserving nonrigid registration of MR breast images using free-form deformation with an incompressibility constraint. *IEEE Trans. Med. Imag.*, 22(6):730–741, June 2003.

[84] K. Rohr, H. S. Stiehl, R. Sprengel, T. M. Buzug, J. Weese, and M. H. Kuhn. Landmark-based elastic registration using approximating thin-plate splines. *IEEE Trans. Med. Imag.*, 20(6):526–34, June 2001.

[85] D. Ruan, J. Balter, M. Roberson, J. Fessler, and M. Kessler. Non-rigid registration using regularization that accomodates local tissue rigidity. In *Great Lakes AAPM*, 2005.

[86] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes. Nonrigid registration using free-form deformations: application to breast MR images. *IEEE Trans. Med. Imag.*, 18(8):712–21, August 1999.

[87] Y. Seppenwoolde, H. Shirato, K. Kitamura, S. Shimizu, M. . Herk, J. V. Lebesque, and K. Miyasaka. Precise and real-time measurement of 3D tumor motion in lung due to breathing and heartbeat, measured during radiotherapy. *Int. J. Radiat. Oncol. Biol. Phys.*, 53(4):822–34, July 2002.

[88] L. A. Shepp and Y. Vardi. Maximum likelihood reconstruction for emission tomography. *IEEE Tr. Med. Im.*, 1(2):113–22, October 1982.

[89] R. L. Siddon. Fast calculation of the exact radiological path for a three-dimensional CT array. *Med. Phys.*, 12(2):252–5, March 1985.

[90] M. Staring, S. Klein, and J. P. Pluim. Nonrigid registration with adaptive, content-based filtering of the deformation field. In *Proc. SPIE 5747, Medical Imaging 2005: Image Proc.*, pages 212–21, 2005.

[91] A. Terstegge, S. Weber, H. Herzog, H. W. Muller-Gartner, and H. Halling. High resolution and better quantification by tube of response modelling in 3D PET reconstruction. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 3, pages 1603–7, 1996.

[92] P. Thevenaz, U. E. Ruttimann, and M. Unser. A pyramid approach to subpixel registration based on intensity. *IEEE Trans. Im. Proc.*, 7(1):27–41, January 1998.

[93] P. Thevenaz and M. Unser. Optimization of mutual information for multiresolution image registration. *IEEE Trans. Im. Proc.*, 9(12):2083–99, December 2000.

[94] K. Thielemans, M. W. Jacobson, and D. Belluzzo. On various approximations for the projectors in iterative reconstruction algorithms for 3D PET. In *Proc. of the 1999 Intl. Mtg. on Fully 3D Im. Recon. in Rad. Nuc. Med.*, pages 232–5, 1999.

[95] K. Thielemans, S. Mustafovic, and L. Schnorr. Image reconstruction of motion corrected sinograms. In *Proc. IEEE Nuc. Sci. Symp. Med. Im. Conf.*, volume 4, pages 2401–6, 2003.

[96] B. Thorndyke, E. Schreibmann, P. Maxim, B. Loo, A. Boyer, A. Koong, and L. Xing. Enhancing 4D PET through retrospective stacking. *Proc. Amer. Assoc. Phys. Med.*, 32(6):2096, June 2005.

[97] H. Ue, H. Haneishi, K. Suga, and H. Iwanaga. Motion correction for synthesis and analysis of respirat ory-gated lung spect image. In *Proc. SPIE 5747, Medical Imaging 2005: Image Proc.*, pages 1173–80, 2005.

[98] M. Unser. Splines: A perfect fit for signal and image processing. *spmag*, 16(6):22–38, November 1999.

[99] M. Unser, A. Aldroubi, and C. R. Gerfen. A multiresolution registration procedure using spline pyramids. In *Proc. SPIE 2034, Wavelet Appl. in Signal and Image Proc.*, pages 160–70, 1993.

[100] P. A. Viola. *Alignment by maximization of mutual information*. PhD thesis, Massachusetts Institute of Technology, 1995.

[101] M. P. Wachowiak and T. M. Peters. Combining global and local parallel optimization for medical image registration. In *Proc. SPIE 5747, Medical Imaging 2005: Image Proc.*, pages 1189–1200, 2005.

[102] C. F. J. Wu. On the convergence properties of the EM algorithm. *Ann. Stat.*, 11(1):95–103, March 1983.

[103] A. Yendiki and J. A. Fessler. A comparison of rotation- and blob-based system models for 3D SPECT with depth-dependent detector response. *Phys. Med. Biol.*, 49(11):2157–68, June 2004.

[104] D. F. Yu, J. A. Fessler, and E. P. Ficaro. Maximum likelihood transmission image reconstruction for overlapping transmission beams. *IEEE Tr. Med. Im.*, 19(11):1094–1105, November 2000.

[105] W. Zangwill. *Nonlinear programming, a unified approach*. Prentice-Hall, NJ, 1969.

[106] R. Zeng, J. A. Fessler, and J. M. Balter. 3D respiratory motion estimation from slowly rotating 2D X-ray projection views. In *Proc. Amer. Assoc. Phys. Med.*, page 2095, 2005. Abstract in Med. Phys 32(6) 2005.

[107] J. Zheng, S. Saquib, K. Sauer, and C. Bouman. Parallelizable Bayesian tomography algorithms with rapid, guaranteed convergence. *IEEE Trans. Im. Proc.*, 9(10):1745–59, October 2000.