QUANTIFYING UNCERTAINTIES IN IMAGING-BASED PRECISION MEDICINE

by

Nicholas Patrick Henscheid

Copyright © Nicholas Patrick Henscheid 2018

A Dissertation Submitted to the Faculty of the

GRADUATE INTERDISCIPLINARY PROGRAM IN APPLIED MATHEMATICS

In Partial Fulfillment of the Requirements

For the Degree of

DOCTOR OF PHILOSOPHY

In the Graduate College

THE UNIVERSITY OF ARIZONA

2018
THE UNIVERSITY OF ARIZONA
GRADUATE COLLEGE

As members of the Dissertation Committee, we certify that we have read the dissertation prepared by Nicholas Henscheid, titled Quantifying Uncertainties in Imaging-based Precision Medicine and recommend that it be accepted as fulfilling the dissertation requirement for the Degree of Doctor of Philosophy.

Harrison H. Barrett
Date: April 11, 2018
Joseph C. Watkins
Date: April 11, 2018
Leonid Kunyansky
Date: April 11, 2018

Final approval and acceptance of this dissertation is contingent upon the candidate's submission of the final copies of the dissertation to the Graduate College.

I hereby certify that I have read this dissertation prepared under my direction and recommend that it be accepted as fulfilling the dissertation requirement.

Dissertation Director: Harrison H. Barrett
Date: April 11, 2018
STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of the requirements for an advanced degree at the University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that an accurate acknowledgement of the source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: Nicholas Patrick Henscheid
Acknowledgements

It takes a village of intellectual, emotional, and financial support to produce a new scientist. None of this would not have been possible without the support of Alison, my family, and significant amounts of patience, encouragement and inspiration from decades worth of mentors and friends. I feel honored and privileged to have been able to pursue such an indulgent intellectual art.

First, I’d like to thank my professors at Western Washington University, who first showed me that mathematics was both deeply fascinating and endlessly entertaining: in particular, Tilmann Glimm and Edoh Amiran who helped encourage me to pursue a PhD.

The notion of becoming a scientist would not have entered my mind as a possibility without having been given the opportunity to spend a semester haphazardly experimenting with wireless power at the Jet Propulsion Laboratory, so I extend a thank you to John Choi and the rest of the Advanced Signal Processing Projects group there for the inspiration. I would also like to thank Allen Toreja at Lawrence Livermore for giving me the opportunity to get out of the desert for a summer and learn more than I ever thought possible about computational high-energy physics.

My time in the Program in Applied Mathematics has been endlessly intellectually stimulating, and I thank all my professors here at Arizona, both in the mathematics and the optical sciences departments, for vastly expanding my passion for and understanding of science and mathematics. In particular I would like to thank Harrison Barrett for subtly and patiently steering me towards a deep interest in precision medicine and molecular imaging.

Perhaps most essential to maintaining sanity during graduate school is a strong, supportive and entertaining group of friends. Endless thanks go to Luke Parsons, Jess Stahl, Kati Standefer, Kat Compton, Andrew Leach, Toby Shearman, Nick Kloehn, David Dolana, Diane Thompson, Nhan Nguyen, Malcolm Rupert and the dozens of other climbers and graduate students whom I am honored to call my friends. You’ve all literally held the other end of my rope, and for that I’m eternally grateful.

For the Santa Catalina mountains, and for the Dragoons, and for the Cactus and the Monsoon and for the Sun, I thank you for making Southern Arizona undeniably magical.
Dedication

To Alison, for her limitless patience and support.
# Contents

- **List of Figures** 8
- **List of Code Listings** 10

**Abstract** 11

1. **Introduction: the Role of Image Data in Precision Medicine** 12
   - 1.1 21st Century Medicine: Personalized, Predictive, Precise 12
   - 1.2 Uncertainties in M-PMED 20
   - 1.3 Molecular Imaging and Emission Computed Tomography 22
   - 1.4 The Role of Numerical Simulation 25
   - 1.5 Outline of the Dissertation 27

2. **Stochastic Modeling in Precision Medicine** 28
   - 2.1 Probability Theory 28
   - 2.2 Random Processes 49
   - 2.3 Object-Oriented Design and Parallel Computing 79
   - 2.4 Simulating Random Variables and Vectors 82
   - 2.5 Simulation of Random Processes 93
   - 2.6 A Survey of Random Process Models & their Simulation 96
   - 2.7 Review of Statistical Inference 109
List of Figures

1.1 System diagram relating inputs to quantities of interest. ........................................ 15
1.2 System diagram with measurement, null and control components. .......................... 16
1.3 System diagram with observed data and \textit{in silico} model. ..................................... 18

2.1 Relationship between spatial uncertainty and uncertainty in a biomarker of interest. 30
2.2 Comparison of different types of joint random variables. ....................................... 45
2.3 The ensemble interpretation of random processes. ................................................. 52
2.4 Parallel flowchart for the \texttt{GridEval} function. ...................................................... 82
2.5 Comparison of naive and improved accept-reject samplers. ...................................... 90
2.6 Demonstration of sampling efficiency of accept-reject sampler as dimension increases. 91
2.7 Simulated Gaussian random fields. ........................................................................... 101
2.8 Simulation of three Poisson point process with common PDF. .................................... 104
2.9 Three generalized lumpy background random fields. .................................................. 108

3.1 Support set, optical phase space and inflow/outflow boundaries. ............................... 132
3.2 Definition of forward and backward exit times. ......................................................... 136
3.3 Geometry for the adjoint propagation operator. ......................................................... 141
3.4 Effect of aperture on angular uncertainty. ................................................................. 144
3.5 Aperture cones for single- and multi-detector ECT systems. ...................................... 145
3.6 Illustration of a fixed-head SPECT imaging system. ................................................... 146
3.7 Illustration of a pixelated photon-counting detector. .................................................. 150
3.8 Maximum likelihood attribute estimation for ECT systems. ...................................... 155
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.9</td>
<td>Comparison of voxel-based and voxel-free approaches to computing ray transforms</td>
<td>171</td>
</tr>
<tr>
<td>3.10</td>
<td>Parallel RK4 method applied to logistic equation</td>
<td>175</td>
</tr>
<tr>
<td>3.11</td>
<td>Comparison of parallel and serial RK4 implementations</td>
<td>175</td>
</tr>
<tr>
<td>3.12</td>
<td>Illustration of the operator-integrator-helper object-oriented design</td>
<td>179</td>
</tr>
<tr>
<td>3.13</td>
<td>Illustration of geometry used to verify ballistic RTE code</td>
<td>181</td>
</tr>
<tr>
<td>3.14</td>
<td>Comparison of numerical ballistic RTE code to exact</td>
<td>183</td>
</tr>
<tr>
<td>3.15</td>
<td>Comparison of numerical ballistic RTE code to exact solution</td>
<td>183</td>
</tr>
<tr>
<td>3.16</td>
<td>System geometry used for the computation of Fourier crosstalk matrices</td>
<td>186</td>
</tr>
<tr>
<td>3.17</td>
<td>Description of detector parameters for photon-counting and photon-processing systems</td>
<td>187</td>
</tr>
<tr>
<td>3.18</td>
<td>Fourier crosstalk matrices for photon-counting and photon-processing systems</td>
<td>187</td>
</tr>
<tr>
<td>3.19</td>
<td>Estimated singular functions for photon-counting and photon-processing systems</td>
<td>188</td>
</tr>
<tr>
<td>3.20</td>
<td>Singular value decay rates for photon-counting and photon-processing systems</td>
<td>189</td>
</tr>
<tr>
<td>4.1</td>
<td>Conceptual framework for model-based precision medicine</td>
<td>191</td>
</tr>
<tr>
<td>4.2</td>
<td>Differential tumor growth curves</td>
<td>206</td>
</tr>
<tr>
<td>4.3</td>
<td>Tumor growth under uncertainty</td>
<td>209</td>
</tr>
<tr>
<td>4.4</td>
<td>Illustration of the TOC curve</td>
<td>213</td>
</tr>
<tr>
<td>4.5</td>
<td>Illustration of the drug concentration decomposition</td>
<td>215</td>
</tr>
</tbody>
</table>
List of Code Listings

2.1 Example of a simple Fourier mode object. ........................................ 80
2.2 Example computational script for performing a virtual clinical trial. .......... 84
3.1 CUDA code for parallel RK4 implementation .................................... 176
3.2 Example script for the RTE simulator ............................................. 179
3.3 Class definition of the RTE CUDA class .......................................... 180
Abstract

In this work, we present a rigorous mathematical framework for the usage of multiple patient-specific molecular images to enable model-based precision medicine, a paradigm of medical decision making defined by the employment of mathematical models of treatment efficacy to direct optimized treatment decisions for individual patients. We address the question of how to define and compute patient-specific probability of treatment success, using random field theory to define the notion of \textit{in silico virtual patient ensembles} and \textit{patient-specific virtual clinical trials}. We then provide a novel and rigorous deterministic and statistical analysis of photon-processing Emission Computed Tomography (ECT) data, highlighting the importance of null functions and Poisson statistics in defining the virtual patient ensemble and probability of treatment success. We discuss novel high-performance parallel numerical methods to simulate virtual patient ensembles and photon processing ECT systems; these simulations will advance our understanding of the uncertainties inherent in imaging-based precision medicine. Finally, we present a spatially resolved model for chemotherapy efficacy that employs ECT data, and demonstrate how our framework can be used to define, compute and optimize patient-specific probability of treatment success in this setting.
Chapter 1

Introduction: the Role of Image Data in Precision Medicine

1.1 21st Century Medicine: Personalized, Predictive, Precise

“It is far more important to know what person the disease has than what disease the person has.”

Hippocrates of Kos [111]

Traditionally, medicine has followed a pattern of understanding diseases as discrete bins into which patients are to be placed [220]. A patient is diagnosed with a particular disorder (placing them into a well-defined bin), and with this label the clinician can proceed with the selection of a validated treatment corresponding to that bin. If you have disease $D$, you receive treatment $T$, perhaps with a dosage modified according to some physiological trait such as age, height, weight and body surface area, and the probability that the treatment will be successful for you is defined in terms of a population of others who had $D$ and received $T$. This highly discrete view of medicine, while effective in many instances, is proving to be ineffective in more complex diseases such as cancer. An excellent example is the case of the chemotherapeutic drug gefitinib. Originally approved in 2003 for the treatment of Non-Small Cell Lung Cancer (NSCLC), it was discovered in 2004 that it was only effective in patients exhibiting a certain mutation in the EGFR gene [184] [223]. Hence the previously discrete bin of ‘non-small cell lung cancer’ was found to require (at least) two additional
sub-bins: those with the EGFR mutation and those without. Indeed, updating disease taxonomy is appreciated as one of the core challenges of modernizing and personalizing medicine [64].

Generally speaking, the goal of precision medicine is to grapple with the simple fact that every patient is different, and to develop medical treatments which are specific to each patient’s unique features. One could argue that most medical treatments – including most cancer treatments – have always been patient-specific to a certain extent, since data collected from the patient is used to select treatments and sometimes even modify the treatment parameters to optimize efficacy. For example, mass- or area-based dosing, where drugs are administered in units of (mass of drug)-per-(mass or surface area of patient), is certainly patient-specific. A more sophisticated example is External Beam RadioTherapy (EBRT) treatment of cancer. In radiotherapy, medical images such as X-Ray Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) are used to delineate tumor boundaries, allowing the radiation oncologist to design a treatment to maximize tumor exposure and minimize normal tissue complications [21] [34].

The grand challenge of personalized, precision medicine is to treat every medical scenario in a manner similar to EBRT: use all available patient-specific data to optimize the treatment choice for every individual [122] [62]. Due to the extremely large quantity and dimension of data that can be measured for each patient and the increasing complexity of treatment strategies for diseases such as cancer, it is unreasonable to expect that human doctors will be capable of efficiently processing this information to produce the most effective results: a more systematic method is called for.

A major strategy towards meeting the challenge of personalized medicine in a data-rich era is the use of so-called in silico predictive models to simulate disease progression and treatment response for an individual patient, hence improving the ability to plan treatments [303] [218] [247] [77]. An in silico model is a mathematical and/or computational model of a disease and treatment that combines fundamental physics, biology, and chemistry with empirical data to predict outcomes for an individual patient. Similar to the usage of in vitro models, which seek to mimic the real patient in a laboratory setting, we can think of in silico modeling as a way to build a ‘virtual’ patient that reflects known features of the real patient. Of course, in the in silico case, the virtual patient exists only in a mathematical or computational sense, instead of being a physical model. While
in silico modeling has advanced greatly, to our knowledge, a mathematically rigorous framework for in silico precision medicine which defines patient-specific treatment efficacy and optimization is largely lacking in the literature. In this dissertation, we provide several tools towards fulfilling this goal.

One of the primary challenges in the in silico paradigm is the definition of a figure of merit with which to optimize patient-specific treatment choices. We argue throughout this work and specifically in Chapter 4 that uncertainties inherent in the collection of patient-specific data imply that any treatment efficacy figure of merit must be probabilistic in nature. However, if the binned notion of disease is replaced with the notion that every patient is unique, it is initially unclear how to define patient-specific probability of treatment success, because no real population exists from which to estimate treatment efficacy. In this dissertation, we provide a rigorous mathematical methodology for computing such patient-specific probabilities via the theory of random processes, which leads to a definition of patient-specific treatment efficacy in terms of a virtual patient ensemble. Whereas treatment efficacy is classically evaluated via clinical trials on a real patient population, we provide a unique definition of patient-specific treatment efficacy in terms of a patient-specific virtual clinical trial performed on the virtual patient population. To our knowledge, the application of virtual clinical trials to precision medicine is entirely novel.

In this spirit, suppose that we wish to predict some quantitative patient characteristic which is parameterized by a scalar or vector $q$; for instance, $q$ could be a total number of tumor cells or spatially averaged tumor cell density in a patient post-chemotherapy. In other words, $q$ is a vector of biomarkers in the broad sense defined by the National Institutes of Health (NIH) and the World Heath Organization (WHO) [264]. That $q$ may consist of biomarkers which are not directly measurable, but may rather consist of quantities which are defined in terms of the patient’s anatomy and physiology but which must be predicted using available data and mathematical biology models, is what distinguishes our treatment from the current biomarker literature, which emphasizes biomarkers that are directly measurable. Our treatment is also unique from the so-called quantitative imaging biomarker literature [217], which again defines biomarkers exclusively in terms of patient imaging data: we will admit that a biomarker may require more than a single imaging
study to compute, perhaps relying on several imaging studies, and also perhaps an auxiliary math-
ematical or computational model, and perhaps additional parameters which may be unavailable for
the individual patient and must be estimated from other sources.

The goal of an in silico model would be to parameterize the patient by a vector $f$ – which
may consist of one or more functions of space and time and possibly additional parameters such as
genomic and proteomic data – then predict the quantities of interest $q$ from $f$. In an ideal world,
$f$ would contain enough information that there would exist an explicit map from the feature vector
$f$ to the true outcome $q$, i.e. we would have

$$q = \mathcal{T}(f)$$

for some function $\mathcal{T}$. This basic relationship is illustrated in Figure 1.1.

![Figure 1.1: A simplified system diagram relating inputs $f$ to biomarker quantities of
interest $q$. In the context of this dissertation, the ‘system’ is analogous to an individual
patient, the inputs are some anatomical and physiological processes which contribute
to disease state, progression and treatment effect, and the quantities of interest will be
biomarkers related to treatment success, for example.](image)

For example, suppose that we wish to predict the total number of tumor cells $N$ at some time
$t_1$. If we let $f \equiv n(r,t)$ parameterize the number density of tumor cells in a patient at position
$r \in V$ and time $t$, then $N$ can be expressed as a mapping

$$N = \mathcal{T}(n) = \int_V n(r,t_1) \, dr$$

Of course, we will almost never have complete access to the full $f$ which would be necessary
to compute $q$ using a map of the form (1.1.2), and we will most likely not know how to compute
$q$ from $f$ even if we did: the explicit function $\mathcal{T}$ in (1.1.1), if it exists at all, may be unknown.
For instance, we may only have access to \( n(r,t_0) \) but wish to compute \( N(t_1) \) with \( t_1 > t_0 \), and an explicit map \((1.1.1)\) that only depends on \( n(r,t_0) \) likely does not exist – we almost certainly require further information to predict \( N(t_1) \).

Rather than having access to the complete, idealized \( f \), what we will have access to is some measurements relating to \( f \), in our case noisy, indirect imaging data \( g \). This imaging data will allow us to estimate a certain component of \( f \) that we call \( f_m \), for ‘measurement’ component, while another component of \( f \), which we call \( f_n \) for ‘null’, is not estimable from \( g \). We will also allow the presence of a \textit{control} component \( f_c \), so that \((1.1.1)\) can be rewritten as

\[
q = T(f_m, f_n, f_c)
\]  

(1.1.3)

By allowing a generic null variable \( f_n \), the relationship \((1.1.3)\) is sufficiently general (in the sense that \((1.1.3)\) may be implicit, in the unfortunate case that \( f_n \) depends on \( q \)), and by including a control variable \( f_c \), we have given ourselves a method to potentially influence \( q \) by changing \( f_c \). The decomposition \((1.1.3)\) is inspired by the systems and control theory literature [250]. For example, in Chapter 4 we discuss chemotherapy: in this case, the drug choice, administered mass, and treatment schedule are all controls that the oncologist has at their command to try and influence \( q \). The modified relationship \((1.1.3)\) is shown in Figure 1.2.

![Figure 1.2: A system diagram showing how observable inputs \( f_m \), unobservable or null inputs \( f_n \), and controllable components \( f_c \) give rise to biomarker quantities of interest \( q \). The observable inputs depend on the available data collection methods (for instance, \textit{in vivo} imaging), while the control input \( f_c \) parameterizes the treatment. The vector \( f_n \) contains elements of the null space of the imaging systems and any other unobservable parameters which may affect the biomarkers \( q \).](image-url)
While (1.1.3) expresses a general relationship between measurable inputs, null components, controls and quantities of interest, we must also account for the fact that we will only be able to use indirect and noisy data to estimate \( f_m \), and the vector \( f_n \) will be wholly unknown and must be treated as random. The true map \( \mathbf{T} \) in (1.1.3) is also usually unknown, so we must employ a mathematical model \( \mathcal{M} \) that attempts to predict \( q \) from some other variables \( \tilde{f} \), which approximate or estimate \( f = (f_m, f_n, f_c) \) in some sense, to produce an estimate \( \tilde{q} \) of \( q \):

\[
\tilde{q} = \mathcal{M}(\tilde{f}) = \mathcal{M}(\tilde{f}_m, \tilde{f}_n, \tilde{f}_c)
\]  

(1.1.4)

Because of uncertainties – which we discuss briefly in the following section and throughout this dissertation – the input \( \tilde{f} \) must always be treated as random: it is a sample from a Virtual Patient Ensemble (VPE), a term we discuss further in Chapter 2.

The measurement component of the VPE, \( \tilde{f}_m \) is usually constructed from the (noisy) data \( g \) by the application of some statistical procedure, such as maximum likelihood estimation or a Bayesian method. The null component \( \tilde{f}_n \) can either be modeled (usually as a realization of some random process) or ignored entirely, while the control \( \tilde{f}_c \) is modeled according to the particular treatment plan. The tilde notation indicates that \( \tilde{f} \) is related to \( f \), but is not necessarily to be thought of as an estimator of \( f \) in the statistical sense: whereas \( f_m \) can be estimated from \( g \) with a point estimator \( \hat{f}_m \), \( f_n \) and \( f_c \) typically cannot; furthermore, we do not require that \( \tilde{f}_m \) be a point estimator – for instance, it could be a sample from a Bayesian posterior.

Because \( \mathcal{M}(\tilde{f}) \) is supposed to estimate some true \( q = \mathbf{T}(f) \), we can (if \( q \) is known, that is, which can be difficult) compute a residual (sometimes called a model discrepancy),

\[
\epsilon = \mathbf{T}(f) - \mathcal{M}(\tilde{f}) = q - \tilde{q}
\]  

(1.1.5)

so that the true \( q \) is expressed in terms of the model as

\[
q = \mathcal{M}(\tilde{f}) + \epsilon
\]  

(1.1.6)
The equation (1.1.6) can be thought of as the precision medicine equivalent to the venerable \( g = Hf + n \) from image science [26]; it expresses a fundamental relationship between an object, quantities of interest and uncertainties. The ideal (but usually unrealistic) scenario is when \( \epsilon \) can be treated as a random quantity with known, well-behaved statistics that do not depend on the unknown \( q \); for instance a mean-zero random vector with small variance would be ideal, because then \( \hat{q} = \mathcal{M}(\hat{f}) \) can be thought of as an unbiased estimator of \( q \), i.e. we will have \( \langle \hat{q} \rangle = q \), where the angle brackets denote a statistical expectation over all realizations of \( \hat{f} \). We will not address model discrepancy in this work, primarily because understanding \( \epsilon \) requires having access to a significant model validation dataset, which we do not currently possess. This modeling framework is displayed in Figure 1.3.

Figure 1.3: A system diagram showing how observed data will be used to approximate inputs to a model \( \mathcal{M} \) which predicts quantities of interest \( \hat{q} = q - \epsilon \). Observed data \( g \) is used to produce approximations \( \hat{f}_m \) of the observable input \( f_m \). The unobservable inputs \( f_n \) are approximated by random processes \( \hat{f}_n \), while the control parameters \( f_c \) are approximated by \( \hat{f}_c \).

The simple system diagrams and function notation employed in Figures [1.1 - 1.3] and Equations (1.1.1), (1.1.3) and (1.1.4) hide a potentially great deal of sophistication. As we have stated, the vector \( g \) is usually indirectly related to \( f_m \) through (say) an imaging system, so there may first be a reconstruction step to produce an estimate \( \hat{f}_m \) of the ‘true’ \( f_m \); we note that \( \hat{f}_m \) is always treated as random, regardless of whether \( f_m \) ought to be. Furthermore, due to the incredible complexity of the human body, any mechanistic model of the form (1.1.4) may be multi-scale, nonlinear,
and hybrid, meaning that it may combine multiple modeling tactics including continuum models, discrete models and perhaps data-driven methods [76]. Thus computing (1.1.4) for a given $\tilde{f}$ may involve the solution of multiple coupled ordinary and partial differential equations, usually via the application of numerical discretization methods. The true object $f$ is almost certainly infinite-dimensional (being comprised of spatiotemporal functions such as tumor cell density and drug concentration), while the measured data $g$ and the approximation $\tilde{f}$ will likely (though not always) be finite-dimensional in nature, in part because we may wish to simulate the model (1.1.4) on a computer; hence we must account for the loss of information in going from $f$ to $\tilde{f}$. Furthermore, there are uncertainties in (1.1.4) on multiple levels: for a fixed $f$, the data $g$ is always a realization of a generalized random vector $g|f$, and the null component $f_n$ may contribute significantly to $q$ but is not captured by $g$. Thus we must always treat $\tilde{f}$ and hence $\tilde{q}$ in (1.1.4) as random quantities, even though (for a fixed patient) the true $q$ is usually nonrandom. Perhaps worst of all, the model $M$ may be wholly incorrect in that it does not accurately predict $q$: the residual $e$ defined in (1.1.5) may be unacceptably large or have undesirable statistics.

It is easy to imagine models for which (1.1.4) would require supercomputing facilities to simulate: in numerical weather prediction, for instance, highly resolved models similar to (1.1.4) demand thousands of computing nodes every day on government clusters around the world. Now imagine that a model with this level of sophistication may be required to predict an individual patient’s disease progression! Hence the selection of $M$ must be met with a great deal of caution, balancing the effects of accuracy, uncertainty, and benefit to the patient. If a much simpler $M$ is capable of performing the task *effectively*, a more complicated $M$ may be unnecessary or even detrimental to patient outcomes. Of course, if it were possible to reliably predict the weather using simpler models, we would; the issues of model selection and reduction are complex and nuanced.

The use of any sort of predictive mathematical modeling, like that suggested by (1.1.4), in the oncology clinic is an extremely recent (as of 2018) development, and the field is ripe for further mathematical and computational development. Thankfully, decades of progress in medical image science and statistical inverse problems [26] [150] [266], computational science and engineering, uncertainty quantification, model validation and risk assessment [267] [257] [216] is available for
One of the fundamental aspects of $\mathcal{M}$-PMED that demands attention from the mathematical sciences is the precise probabilistic modeling of all relevant random quantities, in particular random quantities that are *spatiotemporal* in nature. This will lead us in Chapter 2 to consider *random fields* as a way to incorporate spatial uncertainties into (1.1.4), leading to a mathematically rigorous definition of a *virtual patient ensemble*, and a rigorous definition of patient-specific probability of treatment success. To our knowledge, this mathematical definition of patient-specific treatment efficacy does not exist in the literature.

The second aspect of $\mathcal{M}$-PMED that demands a more rigorous mathematical treatment is the use of noisy, incomplete imaging data to estimate model parameters. Imaging data demands sophisticated mathematical modeling in its own right, since the data is always collected in an indirect manner using noisy measurement devices. This leads us in Chapter 3 to consider a rigorous deterministic and statistical analysis of Emission Computed Tomography (ECT), a general class of functional molecular imaging system based on the emission of light from tagged drug concentrations. We present a novel mathematical analysis of both photon-counting and photon-processing molecular emission imaging systems which illustrates directly the necessity of defining a measurement and null component $f_m$ and $f_n$.

Throughout, we discuss general mathematical strategies that allow the usage of imaging data collected from nearly any device to be used in nearly any predictive model of the form (1.1.4), so the framework is more general than the examples provided. A fundamental feature of our strategy is that we treat the quantification of uncertainty as a core element of the study.

### 1.2 Uncertainties in $\mathcal{M}$-PMED

*“Medicine is a science of uncertainty and an art of probability”*

Sir William Osler [221, Pg. 125]

As we mentioned several times in the previous section, the successful implementation of $\mathcal{M}$-PMED will require a precise understanding of all relevant uncertainties. If we build a virtual patient $\tilde{f}$ using patient-specific data, then predict $\tilde{q} = \mathcal{M}(\tilde{f})$, how certain are we that the predicted value
of $\tilde{q}$ is correct? If there are errors that enter into the generation of $\tilde{f}$, there will almost certainly be errors in our prediction of $\tilde{q}$. The central goal of uncertainty quantification is to understand exactly the sources, magnitudes, and statistical behaviors of such errors.

The basic strategy in Uncertainty Quantification (UQ) is as follows. In the model input-output relationship shown in Equation (1.1.4) and Figure 1.3, the data $g$, the estimated input $\tilde{f}_m$, and the null component $\tilde{f}_n$ must all be treated as random quantities, and the control $\tilde{f}_c$ may or may not be random. Hence, because it is written as a function of a random input, the estimated quantity of interest $\tilde{q}$ is random as well.

By sampling different random inputs $\tilde{f} = (\tilde{f}_m, \tilde{f}_c, \tilde{f}_n)$ and applying the model $M$, one can infer the distribution of outcomes $\tilde{q}$; this is called the propagation of uncertainty. Knowing the distribution of outcomes, we can compute probabilistic quantities such as the average value $\langle \tilde{q} \rangle$, the variance/covariance matrix or operator, or any probability of the form

$$\text{Probability}(\tilde{q} \in E) \quad (1.2.1)$$

where $E$ is some set of outcomes. For example, if $\tilde{q}$ quantifies the outcome of a cancer treatment, we may be interested in the probability that $\tilde{q}$ is in a set of ‘desirable’ outcomes (such as ‘tumor control’ or ‘prolonged life’). The computation of such probabilistic quantities in the computer is the realm of Monte Carlo methods, which we discuss in Chapter 2.

It should be clear that - at least in principle - the uncertainty quantification framework is immediately applicable to the personalized, predictive medicine problem. In the in silico paradigm of precision medicine, we use mathematical models of disease progression and treatment efficacy to predict biomarkers. Such models depend on patient-specific parameters that are uncertain before measurements are taken. Even after measurements are taken, there is still a degree of uncertainty owing to the fact that the measurements are incomplete, indirect and noisy. Thus model parameters, including spatiotemporally varying ones, should be considered as random, and model prediction of treatment outcomes should be given a measurement of uncertainty; the goal of $\mathcal{M}$-PMED should be to compute probabilities of the form (1.2.1). To reduce uncertainty, patient-specific measurements can be designed and performed, but the selection and optimization of such
measurements must analyzed via a risk-benefit analysis: in the clinic, there is only a finite amount of time and resources, and these must be employed to maximal benefit to the patient. In this work, we employ the mathematically rigorous tools and techniques of UQ to define the notions of patient-specific treatment efficacy in terms of virtual clinical trials, which can be seen as the application of a Monte Carlo method to estimate a probability of the form [1.2.1]. To our knowledge, this application of UQ techniques to patient-specific treatment optimization is entirely novel.

1.3 Molecular Imaging and Emission Computed Tomography

“Science begins with counting. To understand a phenomenon, a scientist must first describe it; to describe it objectively, he must first measure it. If cancer medicine was to be transformed into a rigorous science, then cancer would need to be counted somehow - measured in some reliable, reproducible way.”

Siddhartha Mukherjee, in reference to Sidney Farber’s approach to studying leukemia [207]

One of the crucial technologies that allows us to consider $\mathcal{M}$-PMED as a plausible clinical decision making strategy is medical imaging. As we discussed above, if a patient is parameterized by a vector $\mathbf{f}$ which consists of spatiotemporal functions (such as drug concentration $c(r,t)$ and tumor cell density $n(r,t)$), and we require knowledge of $\mathbf{f}$ in order to predict $\mathbf{q}$, without patient-specific data that relates to $\mathbf{f}$ we can do no better than to assume that $\mathbf{f}$ is sampled from an ensemble of functions, that is, a random process, and hence the variability in the predicted $\mathbf{q}$ may be large. The way to reduce this uncertainty is by collecting data that relates specifically to $\mathbf{f}$, and in the clinic, this means medical imaging data.

There are many different types of medical imaging systems, each emphasizing a different range of anatomical or physiological traits. In essence, any imaging system uses some form of energy to extract information from an object: the most common medical imaging systems use energy in the form of visible light, X-Ray and gamma ray radiation, magnetic resonance, sound, and heat. Many systems are now combining several such techniques, and the range of physiological processes that can be investigated is increasingly immense. One of the most versatile classes of imaging system uses ‘tagged’ drugs that target specific biochemical receptors. The tagging process makes the
drug either emit some form of energy (usually light) or be sensitive to a particular form of energy (providing contrast). These techniques are called, broadly speaking, molecular imaging modalities \[291\][292][144]. A particular subset of molecular imaging is Emission Computed Tomography or ECT, which is a broad class of modalities whereby a drug of interest is tagged with a molecule that is capable of emitting light, usually gamma radiation or visible fluorescent light. The imaging system is then designed to collect and measure this light from multiple angles in three dimensions and over time, generating a dataset \(g\) of spatiotemporal photointeractions. Because the amount of light emitted is very low, the quantum nature of photodetection becomes essential, leading to the conclusion that \(g\) is a random process even for a fixed realization of the object \(f\). In ECT, however, there is an extremely precise, well-validated set of mathematical models that connect the object with both the average (deterministic) and the stochastic properties of the data \(g\).

To describe these models briefly, consider the particular case when the object \(f\) consists of a single drug concentration \(c(r, t)\). With an ECT system, \(c \equiv c(r, t)\) is radioactive, and the system will collect a random photointeraction dataset \(g\); because \(g\) depends on \(c\), we will write the data for this object as \(g|c\). In Chapter 3, we will derive an explicit operator \(\mathcal{H}\) which maps \(c(r, t)\) to the statistical average of \(g|c\), i.e. we will have

\[
\mathcal{H}c = \langle g|c \rangle \tag{1.3.1}
\]

where the angle brackets indicate a statistical average over all realizations of the imaging data \(g|c\). The operator \(\mathcal{H}\) in (1.3.1) will turn out to be linear in the case of ECT, which is a distinct advantage for analysis. The physics of photodetection will then imply that \(g|c\) is (depending on the system design) either a Poisson point process or a Poisson random vector with mean \(\mathcal{H}c\); in either case, we will write

\[
g|c \sim \text{Poi}(\mathcal{H}c) \tag{1.3.2}
\]

Because we wish to draw conclusions about \(c\) – for instance, by computing a model output \(\tilde{q} = \mathcal{M}(\tilde{c})\) – it is necessary to ‘invert’ the relationship (1.3.2) by computing a statistical model \(\tilde{c}\) for
we call such a statistical model a patient-specific virtual ensemble, because it depends on the data collected for our patient \((g|c)\), because it is virtual (it is a mathematical or computational model of \(c\), not an ensemble of ‘real’ \(cs\), and because it is random (\(\tilde{c}\) is not a single concentration, but a collection of realizations of drug concentrations). As mentioned previously, \(\tilde{c}\) may be constructed using a combination of statistical point estimation and Bayesian techniques.

In Chapter 3, we will discuss two common strategies for forming the patient-specific virtual ensemble. The first is to compute a point estimator of the measurement component of \(c\), using for instance the Maximum Likelihood (ML) procedure, which returns \(\hat{c}_m\); this estimator is a random quantity, owing to the randomness in \(g|c\). We can then employ asymptotic statistics or the bootstrap procedure to compute the ‘sampling’ distribution of \(\hat{c}_m\). If implemented correctly, this strategy has several appealing properties that we discuss in Chapter 3 but also presents some practical challenges that we will discuss as well, namely that the null component \(c_n\) is a nonestimable parameter. The second method we discuss employs a statistical model of \(c\) in order to make use of Bayes’ theorem to invert the relationship \((1.3.2)\) ‘directly’, arriving at the virtual ensemble \(c|g\); this strategy also has several appealing properties, but presents several practical and philosophical challenges as well, namely the selection of a prior.

In either case, (MLE or Bayes), the procedure produces a virtual ensemble \(\tilde{c}\) that is intended to mimic the ‘true’ \(c\) as much as possible, while also accounting for uncertainties inherent in the imaging system. While Poisson noise is one source of uncertainty, another source of uncertainty arises when imaging data is used, and which is dependent on the deterministic properties of the operator \(\mathcal{H}\). In particular, the operator \(\mathcal{H}\) may destroy information in such a way that even if noise-free data \(\mathcal{H}c\) were available, an exact estimate of \(c\) would still be impossible, and hence uncertainty remains. Without going into the details here, the operator \(\mathcal{H}\) may possess a subspace of null functions, which are functions \(c_n\) for which \(\mathcal{H}c_n \equiv 0\) (meaning no data will ever be produced by such functions) and the operator \(\mathcal{H}\) may have a sequence of singular values which decays rapidly towards zero, meaning that certain components of \(c\) may be severely attenuated. We will employ the Singular Value Decomposition (SVD) and a relative of it, the Fourier Crosstalk Matrix (FCM), to analyze the deterministic properties of ECT imaging systems. These deterministic properties
will suggest a decomposition of the object $f$ into components $f_m$ and $f_n$, which again indicate a ‘measurable component’ (a component whose properties are estimable from the data $g$) and a ‘null component’ $f_n$ whose properties are not realistically estimable from the data $g$. The virtual ensemble $\tilde{f}$ should also reflect this decomposition, with $\tilde{f}_m$ being related to $f_m$ and $\tilde{f}_n$ being related to $f_n$. It is perhaps obvious that if $f_n$ is not estimable from data, but it is desired to make use of it anyway (for instance, in $\tilde{q} = \mathcal{M}(\tilde{f})$), some form of prior information will be necessary to model $f_n$.

1.4 The Role of Numerical Simulation

We take a special emphasis in this work on the use of computational routines for the simulation of objects, imaging systems and the decisions made with imaging data. The purpose of such simulations is to evaluate, in a highly controlled setting, the performance of proposed systems and algorithms. It is not necessarily the goal in this work to develop the highest-performance algorithms and methods for each individual system component, but because of the nature of statistical computing (the same simulation may need to be repeated many times), it is always necessary to write code that performs well enough to obtain results in a reasonable amount of time. In some cases we sacrifice computational sophistication for tractability; we will strive to mention where improvements to software can be made in the future.

The goal of performing simulations in science and engineering has historically been to help understand, design and build real systems and devices in order to test hypotheses and perform tasks, and the same is true in the biomedical sciences. However, the use computational modeling and simulation as a strictly preclinical tool is changing with the onset of $\mathcal{M}$-PMED. As we have emphasized, many illnesses are complex interacting systems phenomena, and each individual patient can now generate very high-dimensional datasets, so the application of in silico models will become a practical necessity in the modern clinic [77].

This paradigm shift brings simulation from the basement laboratory and theoreticians’ office to the forefront of modern medicine, and in our view, the application of the same rigorous scientific computing methodologies that have been developed in the engineering community [216] to ensure
that computational codes produce correct, reliable results should also be applied in the medical community. We also emphasize that good computational science is not simply ‘data science’: to be considered a third pillar of science, computation must act as an integral part of the feedback loop along with theory and experiment, and purely data-driven methods rarely seek this level of companionship with classical science [219]. Because most of the code that was written for this dissertation was intended for personal research use only, we cannot claim to have met this level of standard, but we will strive to at least discuss where such verification and validation should be performed in the future.

As has been well documented elsewhere, the present and future of high performance computing is necessarily parallel: Moore’s law, which ambitiously stated that the number of transistors on a microchip would double every 18 months, has met fundamental physical limits [252]. The solution is to make devices and algorithms operate in parallel, performing many operations simultaneously to increase performance. In this work, we make extensive use of General Purpose Graphics Processing Units (GPGPUs), which are massively parallel: a single NVidia Tesla P100 contains 56 multiprocessors, each with 64 cores, totalling 3584 processing cores; each multiprocessor supports 2048 threads, meaning that (ignoring some technicalities about how the device organizes thread execution) 114688 simultaneous threads can be initiated on a single device – a small city’s worth of workers are at our fingertips.

General purpose graphics devices such as those produced by NVidia were originally designed for high performance computer graphics, computer gaming and console gaming systems such as the Sony Playstation and Microsoft XBox, but by the early to mid 2000’s they became the target of general purpose and scientific computing applications [271] [87]. NVidia picked up on this trend and developed the language CUDA [1], which exists as an extension to both the C and Fortran languages; other GPGPU-specific languages have followed, including OpenCL [208]. The recent swarm of so-called deep learning methods [172] has perhaps only been made possible because of the massively parallel image processing capabilities of GPGPUs.

\[^{1}\text{CUDA stands for Compute Unified Device Architecture}\]
1.5 Outline of the Dissertation

1. In Chapter 2, we discuss the theory and simulation of Physiological Random Processes (PRPs), which are the central mathematical devices that facilitate our definition of the Virtual Patient Ensemble (VPE) and patient-specific probability of treatment success. We take a particular perspective where random processes are defined as generalized random vectors, and provide a wide range of mathematical and computational tools for working with PRPs and hence VPEs. The application of random processes to M-PMED is emphasized throughout.

2. In Chapter 3, we discuss the theory and simulation of molecular Emission Computed Tomography (ECT) imaging systems, emphasizing both the deterministic and stochastic mathematical modeling aspects of real ECT systems. In particular, we provide a rigorous analysis of a class of ECT systems based on the principle of \textit{photon processing}, relating such systems to theoretical results in inverse transport theory. We also discuss parallel simulation strategies for ECT systems, presenting fast and accurate deterministic methods for simulating the scatter-free Radiative Transfer Equation (RTE) and computing the Fourier crosstalk matrix and singular value decomposition, which are practical deterministic tools for understanding uncertainties inherent to imaging data. We also discuss statistical image reconstruction, providing a rigorous definition of patient-specific VPEs in the case when multiple ECT images are collected for an individual patient.

3. In Chapter 4, we provide a novel application of the theory and simulation strategies developed during the previous chapters to the problem of predicting treatment outcomes in precision chemotherapy using ECT data collected for an individual patient. We discuss a spatially resolved mathematical model for chemotherapy treatment effect that requires measuring two ECT images, and we discuss how to compute patient-specific probability of treatment success using this model.

In Chapter 5, we highlight the specific contributions of this dissertation and provide directions for future work. In the Appendix, we provide several additional discussions that were too technical or obscure to retain in the main text.
Chapter 2

Stochastic Modeling in Precision Medicine

2.1 Probability Theory

As discussed in the introduction, the main objective of this work is to discuss uncertainties in Model-based Precision Medicine (abbreviated M-PMED) that arise due to the use of in vivo ECT imaging data. Recall that the goal of M-PMED is to compute a model output

$$\tilde{q} = M(\tilde{f}).$$ (2.1.1)

The input $\tilde{f} = (\tilde{f}_m, \tilde{f}_n, \tilde{f}_c)$ is a mathematical parameterization of a patient and treatment that we call the virtual patient, the output $\tilde{q}$ quantifies the outcome of a treatment (say), and the model $M$ uses either mechanistic biophysical models and/or data-driven methods to predict $\tilde{q}$ from $\tilde{f}$. As we discussed in Section 1.2, $\tilde{f}$ and hence $\tilde{q}$ must be treated as random because of uncertainties due to incomplete and noisy data; we say that $\tilde{f}$ is a sample from a Virtual Patient Ensemble or VPE. The goal of probability and random process theory is to provide a mathematically rigorous foundation for VPE modeling.

One of the basic types of question that we will discuss in Chapter 4 is the following. Suppose that the variable $\tilde{q}$ in (2.1.1) measures the predicted effect of a cancer treatment for a particular
patient in our clinic. Because we treat \( \mathbf{q} \) as random, we would like to be able to answer questions such as ‘what is the probability that \( \mathbf{q} \) assumes a value in a desired set of outcomes?’ Or, written mathematically, we would like to be able to compute a number such as

\[
\text{Probability}(\mathbf{q} \in E_{\text{desired}}) \tag{2.1.2}
\]

where \( E_{\text{desired}} \) is a set of outcomes corresponding to the ‘desired effect’ - for instance in cancer treatment, the desired effect is usually (at least short-term) tumor control and acceptable normal tissue exposure, which we assume can be expressed as \( \mathbf{q} \in E \) for some \( \mathbf{q} \) and some \( E \) (we suggest an explicit example in Chapter 4). A probability such as (2.1.2) is sometimes called a certification probability in the engineering community, because it represents the problem of ‘certifying’ that a certain engineered system (such as a bridge or dam) will meet some design criteria (such as withstanding a certain wind speed or water level) with high probability. The variable \( \mathbf{q} \) measures the predicted response of the system, while the set \( E_{\text{desired}} \) quantifies the design criteria [267, 182].

In the language of this dissertation, (2.1.2) will take the form of a patient-specific treatment success probability.

As we emphasized in Section 1.1, the problem (2.1.2) looks innocuous, but the difficulty lies in the fact that \( \mathbf{q} \) may be a very complicated functional of its inputs, involving for instance the solution of ordinary or partial differential equations. Furthermore, the input \( \mathbf{f} \) in (2.1.1) is usually an approximation of a random process, i.e. a random vector in an infinite-dimensional space. An example of a simple biological quantity of interest is displayed in Figure 2.1.

The goal of this chapter is to use probability and random process theory to rigorously define patient-specific treatment efficacy probability (2.1.2). Throughout this chapter, we introduce concepts and notation, but mostly avoid rigorous definitions; some essential ones are available in Appendix 6.1.

General references for the mathematical foundations of probability include Billingsley [30] and Dudley [84]; the engineer’s reference is Papoulis and Pillai [224]. Barrett and Myers [26] review many of the key concepts in Appendix C and Chapter 8; I will use notation similar to [26] with some exceptions, namely probability laws (measures) will be denoted \( \mathbb{P} \) while probability densities will be
Figure 2.1: A demonstration of how spatial uncertainty relates to uncertainty in a quantity of interest. Here, we compute a random scalar quantity of interest \( Q = \mathcal{M}(\tilde{f}) = (\tilde{f}, h)_{L^2} \), where \( h = h(r) \) is the template function shown in the upper right and \( \tilde{f} \) is a (lognormal) spatial random field (ref. Section 2.6). Three realizations of \( \tilde{f} \) are shown, corresponding to different values of \( q \) (indicated with arrows); a histogram probability density estimate of \( p_Q(q) \), generated from 1024 realizations of \( \tilde{f} \), is also displayed.

denoted with \( p(x) \) instead of \( \text{pr}(x) \). For random variables, an unbolded capital letter indicates the random variable while a lowercase letter is a realization, e.g. \( X \) is a random variable with realization \( x = X(\omega) \). All vector quantities, including functions, will be denoted with a bold lowercase Latin letter, e.g. \( f, g, u, v, x, y, \) etc. If a vector or function is random, we choose not to indicate this explicitly with different notation unless necessary. For instance, if \( f \) is a random process, \( f_j \) or \( f_\omega \) will be a particular realization of the process. If the arguments of a scalar-valued function or vector are needed, we unbold, so that \( f \equiv f(r, t) \) indicates a scalar-valued spatiotemporal random
process, while \( \mathbf{u} \equiv u(i) \) is a finite-dimensional vector. If a function is vector-valued, we retain the bold, i.e. \( \mathbf{f} \equiv f(r, t) \) is a vector-valued spatiotemporal random process. Matrices will be bold uppercase Latin e.g. \( \mathbf{A}, \mathbf{B}, \mathbf{K} \), with arguments indicated with either subscript or function notation e.g. \( A_{ij} \) or \( A(i, j) \). Operators will be bold, uppercase and calligraphic e.g. \( \mathbf{A}, \mathcal{H} \) or \( \mathcal{M} \). Abstract vector spaces will be given script letters such as \( \mathcal{X} \) or \( \mathcal{H} \), except well-known spaces such as \( L^2 \). Sets will typically be unbolded, uppercase Greek or unbolded script Latin letters e.g. \( \Omega, \mathcal{I}, \mathcal{U}, \mathcal{V} \). Deviations from this notation will be mentioned explicitly.

2.1.1 Probability Spaces

The basic abstract notion in probability theory is the probability space, which consists of a sample space, denoted generically \( \Omega \), a collection of event sets denoted by \( \mathcal{F} \), and a probability function, denoted by \( \mathbb{P} \), which is a ‘set function’ mapping event sets \( E \in \mathcal{F} \) to real numbers \( \mathbb{P}(E) \in [0, 1] \).

While \( \Omega \) can be an arbitrary set, the following requirements are placed on \( \mathcal{F} \) and \( \mathbb{P} \) in order to produce a theory that is both internally consistent and consistent with intuition:

1. The events \( \mathcal{F} \) must form a sigma algebra (see Appendix 6.1);

2. The probability function \( \mathbb{P} \) must be a probability measure on \( \mathcal{F} \) (see Appendix 6.1).

These definitions were solidified from a set of axioms put forth by Kolmogorov [162] [253], and one can combine the definitions of sigma algebra and the probability function to derive the entire calculus of probability. We assume a ‘pragmatic’ philosophy of probability, whereby both frequentist and Bayesian notions of probability are found to be useful in their appropriate context; see [26], [225] and Appendix 6.3.

We will nearly always make the assumption that the sigma algebra of events is generated by some underlying topology, that is, it is the Borel sigma algebra (see Appendix 6.1). This sigma algebra makes it possible to measure the probability of events such as intervals of real numbers \( (a, b) \subseteq \mathbb{R} \), as well as naturally define probability measures on infinite-dimensional spaces of functions such as the Hilbert space \( L^2(V) \). Thus, unless otherwise specified, the word ‘event’ can be replaced with ‘Borel set’ in this work. \(^1\)

\(^1\)We will typically avoid explicit discussions of the class of event sets in this work to avoid ‘loosing the forest for
2.1.2 Random Variables

While the probability space is the basic abstract mathematical object that defines probability of events, the central object of study in probability is not probability spaces per se, but random variables and their distributions [260, §1.1.1].

For our purposes, the concept of a random variable is central because it represents a quantity of interest $x \in \mathbb{R}$ whose value – for whatever reason – either fluctuates between repeated observations or is otherwise uncertain. The rigorous definition of a real random variable is that it is a so-called measurable function $X$ (see Appendix 6.1) from an abstract probability space $(\Omega, \mathcal{F}, \mathbb{P})$ to the real numbers $\mathbb{R}$, i.e.

$$X : \Omega \to \mathbb{R}, \quad x = X(\omega)$$

While this definition is correct, the ‘underlying’ probability space $\Omega$ is not always directly accessible in the sense that we may not know all the physical reasons why a variable fluctuates, and hence cannot always make a definitive statement about what $(\Omega, \mathcal{F}, \mathbb{P})$ is or what the explicit mapping from $\omega \in \Omega$ to $x = X(\omega) \in \mathbb{R}$ is. Realistically, if $x$ is some physically observable quantity such as a length or mass, we will only have available some set of repeated observations, say $x_1, x_2, \ldots$; no explicit knowledge of $\Omega$ or $X$ is necessary to estimate the distribution of $x$ and probabilities of the form (2.1.2). Of course, it is also frequently the case that $x$ represents some uncertain quantity that is not directly observable, but is instead predictive in nature: for example the weather tomorrow or the effect of a proposed treatment on a particular patient are quantities we may wish to predict, and must treat as random in the sense that various sources of uncertainty may prevent our ability to perform an exact prediction. In this case, we would still like to be able to compute probabilities like (2.1.2), and thus one of main the goals of stochastic modeling and computation is to effectively model $\Omega$ and $X$ in such a way that the ‘correct’ distribution of values is obtained, meaning that probabilities like (2.1.2) can be computed.

\footnote{the trees’, but the author is certainly aware that one should be careful, particularly when treating probability in infinite dimensions where other sigma algebras, such as the cylinder sets [100], frequently appear.}
The distribution of $X$ is the probability function $\mathbb{P}_X$ which acts on events $E \subset \mathbb{R}$ via

$$
\mathbb{P}_X(E) = \mathbb{P}(X^{-1}(E)) = \mathbb{P}(\{\omega \in \Omega : X(\omega) \in E\})
$$

If $X$ has distribution $\mathbb{P}_X$, we write $X \sim \mathbb{P}_X$. Again, this definition is correct, but difficult to use if we do not know $\Omega$, $\mathbb{P}$, and the explicit mapping $\omega \mapsto X(\omega)$. Instead, we hope to have either an explicit form for $\mathbb{P}_X$ – usually in the form of a probability density, which we discuss in Section 2.1.4 – or a collection of samples of $X$, from which we can infer a form for $\mathbb{P}_X$; this is the realm of statistics, and a topic that we will return to in 2.7.

Alternately, as mentioned above, we can and will model the set $\Omega$ and the mapping $X$ – that is, we can select a concrete set $\Omega$ (such as the interval $[0, 1]$ or a class of functions) and a concrete probability measure $\mathbb{P}$ that is easy to simulate (such as the uniform measure on $[0, 1]$ or a known random process), then construct an explicit mapping $X : \Omega \to \mathbb{R}$ that gives rise (at least approximately) to the distribution of values that we require or expect. This type of modeling is one of the central goals of this dissertation: we will be interested in a random quantity $\tilde{q}$, which we will express as a function of another random quantity $\tilde{f}$ through the mapping $\tilde{q} = \mathcal{M}(\tilde{f})$. Our model of $\tilde{f}$ will come from a statistical modeling of physiological objects and ECT images, while the mapping $\mathcal{M}$ will arise from a mathematical model of the problem at hand, namely the prediction of treatment outcomes.

We make a brief comment about a technical mathematical condition before moving on. In this work, all functions are assumed to be measurable, in the sense discussed in Appendix 6.1, with respect to the appropriate sigma algebras of events. We also quickly define a complex random variable as a (measurable) function $Z : \Omega \to \mathbb{C}$; the real and imaginary parts of a complex random variable must be ordinary real random variables.

2.1.3 Random Vectors

A random variable represents a single, real- or complex-valued uncertain quantity $X$. It is also useful to consider multiple random quantities $X_1, X_2, \ldots, X_n$, whose values may or may not be statistically related. For example, suppose we measure the concentration $c(r, t)$ of a drug in a
patient at 10 spatiotemporal locations \((r_1, t_1), \ldots, (r_{10}, t_{10})\); this forms a 10-dimensional random quantity, and these values are likely related if the sample locations are close together, i.e. if \(c(r_1, t_1)\) is large, this might indicate that \(c(r_2, t_2)\) is probably large as well. Random vectors in image science can be very high-dimensional: for example, a 10 megapixel camera effectively produces a sample from a 10-million-dimensional random vector, though, this dimension is frequently reduced through compression. In Chapter 3 we discuss an ECT imaging system design which produces random data that is best modeled as a sample from an infinite-dimensional random vector, that is, a random process. We discuss random processes in Section 2.2.

As in the definition of a random variable, we fix an abstract probability space \((\Omega, \mathcal{F}, \mathbb{P})\). We will define a (real) random vector \(\mathbf{u}\) as a function from \(\Omega\) to \(\mathbb{R}^n\); a complex random vector as a function from \(\Omega\) to \(\mathbb{C}^n\). As before, the distribution of \(\mathbf{u}\) is defined abstractly for event sets \(E\) as

\[
\mathbb{P}_{\mathbf{u}}(E) = \mathbb{P}(\mathbf{u}^{-1}(E)) = \mathbb{P}\left(\{\omega \in \Omega : \mathbf{u}(\omega) \in E\}\right),
\]

and as before this definition is purely formal unless an explicit form for the function \(\mathbf{u}(\omega)\) is known – in many cases we know \(\mathbb{P}_{\mathbf{u}}\) directly (for instance, we know the PDF of \(\mathbf{u}\)), or we model it by selecting the appropriate \((\Omega, \mathcal{F}, \mathbb{P})\) and mapping \(\mathbf{u}\). Though \(\mathbf{u}\) is technically a function, we frequently abuse notation and write \(\mathbf{u} \in \mathbb{R}^n\) to indicate that \(\mathbf{u} : \Omega \to \mathbb{R}^n\). We also say that two random vectors \(\mathbf{u}\) and \(\mathbf{v}\) are equal in distribution if \(\mathbb{P}_{\mathbf{u}}(E) = \mathbb{P}_{\mathbf{v}}(E)\) for all events \(E\); hence random vectors that are equal in distribution can be treated as the same in a statistical sense.

Given a random vector \(\mathbf{u}(\omega)\), a variety of random variables can be extracted by transformations. Suppose that \(\mathbf{u} \in \mathbb{R}^n\) and let \(M : \mathbb{R}^n \to \mathbb{R}\) be a function; then, \(q = M(\mathbf{u})\) is a random variable, because it can be written as a mapping from \(\Omega\) to \(\mathbb{R}\). For example, given a (nonrandom) vector \(\mathbf{y} \in \mathbb{R}^n\), the dot product \(q(\omega) = \mathbf{y} \cdot \mathbf{u}(\omega)\) is a random variable, so that in particular, each component of \(\mathbf{u}\) is a random variable. We discuss more general transformations in Section 2.1.5.

We will indicate the components of a random vector as follows. Writing \(\mathbf{u} \in \mathbb{R}^n\) as a column vector, we have (using \(\dagger\) to denote the matrix transpose here and for the remainder of the work)

\[
\mathbf{u} = [U_1, \ldots, U_n]^\dagger,
\]
where each $U_i$ is a component random variable.

More generally, one could define a random vector as a mapping from $\Omega$ to an arbitrary vector space $\mathcal{X}$; we will tackle this more general issue in Section 2.2.6 in the context of random processes.

One of the main issues we face when dealing with random quantities is the description of their statistical behavior. We have already given the probability distribution (2.1.3), which specifies how to compute the probability that a variable assumes a value in a set $E$, but we seek other, more easily manipulated descriptions. The first such description is provided by the Probability Density Function, or PDF; later, we discuss the Characteristic Function (CF) and the moments, which provide alternate statistical descriptions of random quantities.

### 2.1.4 Probability Densities

Suppose that we have a function $p(x)$, defined for $x \in \mathbb{R}^n$, such that

$$p(x) \geq 0 \quad \text{and} \quad \int_{\mathbb{R}^n} p(x) \, dx = 1$$

Then, we can define a probability function $\mathbb{P}(E)$ for events $E \subset \mathbb{R}^n$ via

$$\mathbb{P}(E) = \int_E p(x) \, dx.$$ (2.1.4)

Proving that (2.1.4) defines a probability measure is an elementary exercise in integration theory. The obvious question then arises: given a random vector $u$ with distribution $\mathbb{P}_u$, does there necessarily exist a function $p_u(x)$ such that (2.1.4) holds for all events $E$? If we demand that $p_u(x)$ be a classical (as opposed to generalized) function, the answer is no: consider for example a random variable $X$ which always assumes the value 0. The probability law $\mathbb{P}_X$ is then a Dirac measure, that is, $\mathbb{P}_X(E) = 1$ if $0 \in E$ and zero otherwise, and there does not exist a classical function having the property that $\int_E p(x) \, dx = 1$ for all $E$ containing zero and $\int_E p(x) \, dx = 0$ otherwise. We say that this random variable is discrete because it concentrates its behavior on a discrete set; a probability function $\mathbb{P}$ for which a classical function $p(x)$ does exist with the property (2.1.4) is called absolutely continuous (with respect to Lebesgue measure). A necessary and sufficient condition for
the existence of a classical function $p(x)$ which satisfies (2.1.4) is that $P(E) = 0$ for all sets such that $\int_E dx = 0$; note that the Dirac measure does not satisfy this property, since $P(\{0\}) = 1$ while $\int_{\{0\}} dx = 0$.

A function $p(x)$ which satisfies (2.1.4) is called a Probability Density Function or PDF. We will make the assumption in this work that all finite-dimensional random vectors $u$ possess – at least in principle – a generalized PDF of the form

$$p_u(x) = p_u^c(x) + p_u^d(x) = p_u^c(x) + \sum_{j=1}^{\infty} P_j \delta(x - x_j) \quad (2.1.5)$$

where $p_u^c(x)$ is the absolutely continuous component and $p_u^d(x)$ is the discrete component. We require that $p_u^c(x) \geq 0$ for all $x$, $P_j \geq 0$ for all $j$ and

$$\int_{\mathbb{R}^n} p_u(x) \, dx = \int_{\mathbb{R}^n} p_u^c(x) \, dx + \sum_{j=1}^{\infty} P_j = 1$$

Note that (2.1.5) can also easily be extended to complex random vectors. If the PDF of a random vector $u$ is known explicitly, the problem of computing probabilities $P(u \in E)$ is reduced to a problem of integration:

$$P_u(E) = \int_E p_u(x) \, dx \quad (2.1.6)$$

While we assume that finite-dimensional random vectors posses a generalized PDF, it may be the case that an explicit form of this PDF is not known; this is frequently the case when $\tilde{q} = \mathcal{M}(\tilde{f})$ is a random vector defined in terms of a model $\mathcal{M}$ and random process $\tilde{f}$. In this case, we may need to resort to Monte Carlo methods, which we discuss in Section 2.1.6.

In many situations, it is also useful to define a function called the Cumulative Distribution Function or CDF. For a random vector $u$, this is the function $F_u(x)$ defined via

$$F_u(x) = P_u \left( (-\infty, x_1] \times \cdots \times (-\infty, x_n] \right)$$

If $u$ is absolutely continuous, then $F_u$ is piecewise smooth (i.e. continuous and differentiable almost
everywhere) and we have the following relationships between the CDF and the PDF:

\[ p_u(x) = \frac{\partial^n}{\partial x_1 \cdots \partial x_n} F_u(x), \quad F_u(x) = \int_{-\infty}^{x_n} \cdots \int_{-\infty}^{x_1} p_u(x') \, dx' \]  \hspace{1cm} (2.1.7)

If \( u \) has discrete component, the CDF will have jump discontinuities, and (2.1.7) still holds if both sides are interpreted in the sense of generalized functions.

### 2.1.5 Transformations

As we mentioned briefly in Section (2.1.3), we can derive many random variables from a given random vector by mapping \( \mathbb{R}^n \) to \( \mathbb{R} \). More generally, it is always the case that functions of random quantities are also random quantities. In the following, real quantities can be replaced by complex quantities \( \text{mutatis mutandis} \). So, suppose that \( u \in \mathbb{R}^n \) is a random vector, and let \( M: \mathbb{R}^n \to \mathbb{R}^m \). Then, \( v = M(u) \) is an \( \mathbb{R}^m \)-valued random vector. The distribution of \( v \) is defined abstractly through (2.1.3), i.e. for any event \( E \) we have

\[ P_v(E) = P_u(M^{-1}(E)) = P_u(\{x \in \mathbb{R}^n : M(x) \in E\}) \]  \hspace{1cm} (2.1.8)

This is the abstract change of variables theorem. In terms of the PDFs of \( u \) and \( v \), we have

\[ \int_E p_v(y) \, dy = \int_{M^{-1}(E)} p_u(x) \, dx \]  \hspace{1cm} (2.1.9)

While an explicit formula exists for \( p_v \) in some cases, the abstract definitions (2.1.8) and (2.1.9) are sometimes the only available method to compute probabilities of the sort \( P_v(E) \). It is sometimes possible to derive the CDF for \( v \), because (2.1.9) implies that

\[ \int_{-\infty}^{y_n} \cdots \int_{-\infty}^{y_1} p_v(y') \, dy' = \int_{\{x : M(x) \leq y\}} p_u(x) \, dx \]

where the inequality \( M(x) \leq y \) is interpreted componentwise. So, if the sublevel set \( \{x : M(x) \leq y\} \) is readily computable, the CDF for \( v \) is obtained, and hence the PDF by differentiation as

\(^2\text{meaning ‘changing what needs changed’}\)
in (2.1.7). If \( M : \mathbb{R}^n \to \mathbb{R}^n \) is a smooth bijection and \( G = M^{-1} \) is also smooth, we have the differential change-of-variables formula:

\[
p_v(y) = p_u(G(y)) |J_G(y)|
\]

where \( |J_G(y)| \) is the determinant of the Jacobian matrix of the inverse map evaluated at \( y \) [26].

Transformations are essential in this work because we are concerned with the statistical properties of \( \tilde{q} = M(\tilde{f}) \); we will derive – explicitly in some cases – the statistical properties of the input \( \tilde{f} \) (typically a random process), from which the statistical properties of \( \tilde{q} \) can be derived either via transformation laws similar to those given here or by Monte Carlo simulation.

### 2.1.6 Expectation and Monte Carlo Simulation

Given a random vector \( u \in \mathbb{R}^n \) with distribution \( P_u \) and PDF \( p_u(x) \) of the form (2.1.5), the expected value of \( u \) is the (non-random) vector \( \bar{u} = \langle u \rangle \in \mathbb{R}^n \) defined, if it exists, via

\[
\bar{u} = \langle u \rangle = \int_{\Omega} u(\omega) \ dP(\omega) = \int_{\mathbb{R}^n} x \ dp_u(x) = \int_{\mathbb{R}^n} x p_u(x) \ dx.
\]  

(2.1.10)

Note that we interpret the integral of a vector component-wise, that is, as a vector of \( n \) one-dimensional integrals. The first of the three integrals in (2.1.10) is, again, defined in terms of the background space. The second and third are more readily computable: we have assumed that a PDF of the form (2.1.5) is available for all finite-dimensional random vectors, so (2.1.10) is defined in terms of a classical (Lebesgue) integral. If necessary, we will indicate the random quantity which is being averaged over with a subscript, i.e. \( \langle u \rangle_u \).

Note that because it is a generalized integral, the expectation operation is linear in its argument, i.e. \( \langle \alpha u + \beta v \rangle = \alpha \langle u \rangle + \beta \langle v \rangle \) for any random vectors \( u, v \) and scalars \( \alpha, \beta \).

We now discuss how expectations of the form (2.1.10) can be computed as the limiting value of a sequence of finite averages. Suppose that we can generate Independent, Identically Distributed (I.I.D.) copies of \( u \) (we define I.I.D. in Section 2.1.9), say \( u_1, u_2, \ldots \) where each \( u_j \sim P_u \); then, if
\( \hat{u} < \infty, \)

\[ \hat{u} = \lim_{N \to \infty} \frac{1}{N} \sum_{j=1}^{N} u_j. \]  

The result (2.1.11) is the Law of Large Numbers or LLN. The meaning of the limit requires some care, because the quantity on the left is deterministic while the quantity on the right is random: the strong LLN states that the equality is almost sure, i.e. the set of sequences for which equality fails has probability zero.

One can also consider expectations of transformed variables. So, let \( v = M(u) \) as before. The expected value of \( v \) can be written as

\[ \bar{v} = \langle v \rangle_v = \int_{\mathbb{R}^m} y p_v(y) \, dy = \int_{\mathbb{R}^n} M(x)p_u(x) \, dx = \langle M(u) \rangle_u \]  

The line of equalities (2.1.12) is sometimes called the law of the unconscious statistician [287].

Equations (2.1.12) and (2.1.11) suggest a simulation technique for computing \( \bar{v} \): draw \( N \) independent samples from the random vector \( u \), resulting in a sequence \( u_1, \ldots, u_N \), then compute \( v_j = M(u_j) \) and apply (2.1.11):

\[ \bar{v} \approx \frac{1}{N} \sum_{j=1}^{N} v_j = \frac{1}{N} \sum_{j=1}^{N} M(u_j) \]  

This is called the Direct Monte Carlo strategy for estimating (2.1.12). By removing the assumption that the samples are independent, one can obtain different Monte Carlo schemes, the most popular being the class of Markov Chain Monte Carlo (MCMC) methods, where the sequence \( u_1, u_2, \ldots \) forms a Markov Chain [236] [178] [26]. Convergence results similar to the LLN for MCMC go by the name ergodic theorems, and are usually more subtle to prove than the classical LLN.

As a particular case of (2.1.13), suppose we wish to compute a treatment efficacy probability of the sort \( p_{\bar{q}}(E) \), where \( \bar{q} = M(\hat{f}) \), as in (2.1.2). By noting that \( p_{\bar{q}}(E) = \langle \chi_E(\bar{q}) \rangle \) where \( \chi_E(\bar{q}) = 1 \)
if \( \bar{q} \in E \) and zero otherwise, we have

\[
P_{\bar{q}}(E) = \lim_{N \to \infty} \sum_{j=1}^{N} \chi_{E}(\bar{q}_j) = \lim_{N \to \infty} \frac{M(N)}{N} \approx \frac{\#(\{\bar{q}_1, \ldots, \bar{q}_N\} \cap E)}{N} \tag{2.1.14}
\]

where \( M(N) = \#(\{\bar{q}_1, \ldots, \bar{q}_N\} \cap E) \) is the number of times \( \bar{q} \in E \) out of \( N \) trials. Returning to the idea of a Virtual Clinical Trial (VCT) (discussed in Section 1.1), imagine that we assemble a virtual cohort of \( N \) ‘patients’, \( \tilde{f}_1, \ldots, \tilde{f}_N \), sampled from a common random vector \( \tilde{f} \) (we may assume \( \tilde{f} \) is finite-dimensional for now). Then, if \( \bar{q} = \mathcal{M}(\tilde{f}) \) computes some quantity of interest - for instance predicted treatment effect - we can estimate the probability that the treatment effect is a desired one by computing \( \bar{q}_j = \mathcal{M}(\tilde{f}_j) \), and counting the proportion of \( \bar{q}_j \)'s that satisfy the desirability criterion \( \bar{q}_j \in E \).

In essence, (2.1.14) is how the weather is predicted: let \( \mathcal{M}(u) \) be a numerical simulation of the weather tomorrow, where \( u \) is a random quantity which parameterizes the state of the atmosphere today (usually, \( u \) is conditional on some measured data). By drawing many samples of \( u \), we can compute the number of times the simulation results in rain (say), and use (2.1.14) to compute the probability of rain tomorrow; such a prediction is called an ensemble forecast, where the sampled collection of \( u_j \)s are what is meant by the ensemble [286].

### 2.1.7 Moments

A special case of the expected value (2.1.12) is when we consider powers of a random vector \( u \in \mathbb{R}^n \).

Let \( \alpha \in \mathbb{Z}_{\geq 0}^n = \{ (\alpha_1, \ldots, \alpha_n) : \alpha_j \in \{0, 1, 2, \ldots\} \} \) be a multiindex. We define the \( \alpha \)th moment of \( u \) as the expected value (if it exists)

\[
\langle u^{\alpha} \rangle = \int_{\mathbb{R}^n} x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n} p_u(x) \, dx
\]

(2.1.15)

For example, if \( |\alpha| = \sum_{j=1}^{n} \alpha_j = 1 \) (\( \alpha \) consists of a single one, in the \( j \)th slot), we have the component means:

\[
\bar{U}_j = \langle u^{(0, \ldots, 1, \ldots, 0)} \rangle = \int_{\mathbb{R}^n} x_j p_u(x) \, dx = \int_{-\infty}^{\infty} x_j p_{u_j}(x_j) \, dx_j
\]

40
We can also shift \( \mathbf{u} \) by its mean value first, then consider the moments of the resulting vector, which results in the definition of the \( \alpha \)th centered moment of \( \mathbf{u} \) as

\[
\left\langle (\mathbf{u} - \bar{\mathbf{u}})^\alpha \right\rangle = \int_{\mathbb{R}^n} (x_1 - \bar{U}_1)^{\alpha_1} \cdots (x_n - \bar{U}_n)^{\alpha_n} p_u(x) \, dx
\]

(2.1.16)

These moments measure the degree of deviation from the mean value, and thus describe how ‘dispersed’ the random quantity is.

The collection of all \( n^2 \) centered moments of order two (that is, for all \( \alpha \) such that \( |\alpha| = 2 \)) can be assembled into a matrix called the covariance matrix of \( \mathbf{u} \). We write the covariance matrix as \( \mathbf{K}_u \equiv \mathbf{K}_u(i,j) \), with components

\[
K_u(i,j) = \left\langle (U_i - \bar{U}_i)(U_j - \bar{U}_j) \right\rangle
\]

(2.1.17)

Note that if \( i = j \), we have the variances of the component variables:

\[
K_u(i,i) = \left\langle (U_i - \bar{U}_i)^2 \right\rangle = \text{Var}(U_i).
\]

The off-diagonal elements are called the covariances between elements of \( \mathbf{u} \):

\[
K_u(i,j) = \text{Cov}(U_i, U_j) := \left\langle (U_i - \bar{U}_i)(U_j - \bar{U}_j) \right\rangle.
\]

Note that we can also write \( \mathbf{K}_u \) as the expected value of the random matrix \( (\mathbf{u} - \bar{\mathbf{u}})(\mathbf{u} - \bar{\mathbf{u}})^\top \), where \( \top \) indicates the matrix transpose:

\[
\mathbf{K}_u = \left\langle (\mathbf{u} - \bar{\mathbf{u}})(\mathbf{u} - \bar{\mathbf{u}})^\top \right\rangle.
\]

(2.1.18)

If \( \mathbf{u} \in \mathbb{C}^n \) is a complex random vector, we modify the definitions \( \text{(2.1.17)} \) and \( \text{(2.1.18)} \) to read

\[
K_u(i,j) = \left\langle (U_i - \bar{U}_i)(U_j - \bar{U}_j)^* \right\rangle
\]

(2.1.19)

\[
\mathbf{K}_u = \left\langle (\mathbf{u} - \bar{\mathbf{u}})(\mathbf{u} - \bar{\mathbf{u}})^\dagger \right\rangle
\]

(2.1.20)
where † indicates the conjugate transpose and * indicates the complex conjugate. The covariance matrix is Hermitian symmetric ($K_u = K_u^*$) and positive semidefinite ($z^\dagger K_u z \geq 0$ for all $z$).

Frequently, only the lower-order moments of a random vector are known (and then usually only approximately) from experimental data: for instance, we may have estimates of the mean and covariance matrix of $u$, but nothing else. The question then arises whether such knowledge adequately constrains the statistics of $u$. The answer is unfortunately no, in general: it is easy to construct two random vectors for which all moments of order $|\alpha| \leq 2$ match, but for which $P_{u_1} \neq P_{u_2}$. In fact, not even the collection of all moments of a random vector is guaranteed to fully specify it [183]; an example is the lognormal random variable, which has finite moments of all orders but for which its distribution is not uniquely specified by these moments (they grow too quickly); see [42] for a discussion. If a small number of moments are known and a particular distribution is desired, the maximum entropy approach provides an appealing method to construct one; see Section 2.4.5. For example, if only the mean and covariance of a random vector $u \in \mathbb{R}^n$ are known, the maximum entropy distribution is the multivariate Gaussian, which is completely specified by its mean and covariance.

2.1.8 The Characteristic Function

As we have discussed, there are so far four ways to specify the statistics of a random vector $u$: we can provide the probability distribution $P_u$, which is a measure; or we can provide a PDF $p_u(x)$, which is a (generalized) function on $\mathbb{R}^n$; or we can provide the CDF $F_u(x)$. One can also provide moments of $u$, but as mentioned, the moments may not fully constrain $P_u$. An alternative complete description is provided by the characteristic function.

The characteristic function of a random vector $u \in \mathbb{R}^n$ is defined as the Fourier transform of the PDF of $u$ (using the convention of Barrett and Myers [26]):

$$\psi_u(\xi) = \left\langle \exp(-2\pi i \xi^\dagger u) \right\rangle_u = \int_{\mathbb{R}^n} \exp(-2\pi i \xi^\dagger u) p_u(x) \, dx \quad (2.1.21)$$

For any random variable, the characteristic function is a uniformly continuous, positive-definite function of $\xi$ such that $\psi(\xi) \leq \psi(0) = 1$ for all $\xi \in \mathbb{R}^n$. Positive definiteness means that any matrix
of the form $A(i, j) = \psi(\xi_i - \xi_j)$ is positive (semi)definite. Furthermore, the characteristic function is a complete statistical characterization of a random vector: a generalization of the Fourier inversion theorem due to Bochner [31] [100] [183] states that any continuous, positive-definite function $\psi(\xi)$ with $\psi(0) = 1$ is the characteristic function of some unique (in distribution) random vector $\mathbf{u}$ in $\mathbb{R}^n$. The Fourier inversion formula provides the PDF of $\mathbf{u}$ given the CF of $\mathbf{u}$:

$$p_u(x) = \int_{\mathbb{R}^n} \exp(2\pi i \xi^\top u) \psi_u(\xi) \, d\xi$$

The above formula must be interpreted in the sense of generalized functions because $p_u(x)$ may contain discrete component (ref. (2.2.23)); if $\psi_u$ decays sufficiently rapidly, then $\mathbf{u}$ will be absolutely continuous (i.e. contain no discrete part).

Given a characteristic function $\psi_u(\xi)$, one can also derive all the moments (2.1.15) by taking derivatives at $\xi = 0$. Furthermore, the $\alpha$th moment exists if and only if the $\alpha$th derivative of $\psi_u(\xi)$ exists at zero. The general formula is

$$\langle u^\alpha \rangle = \frac{1}{(-2\pi i)^{|\alpha|}} \left. \frac{\partial^{|\alpha|} \psi_u}{\partial \xi^\alpha} \right|_{\xi = 0}, \quad |\alpha| = \sum_j \alpha_j.$$

In Section 2.2.8, we will see that an infinite-dimensional generalization of the characteristic function is useful for providing a complete statistical characterization of a random process.

### 2.1.9 Joint and conditional probability, independence

In the following discussion, all real quantities can be replaced with complex quantities *mutatis mutandis*.

Given two random vectors $\mathbf{u} \in \mathbb{R}^n$ and $\mathbf{v} \in \mathbb{R}^m$ with PDFs $p_u(x)$ and $p_v(y)$, respectively, we can treat them as a larger random vector $\mathbf{w} = [\mathbf{u}^\top, \mathbf{v}^\top]^\top \in \mathbb{R}^{n+m}$ if we provide a joint PDF. A joint
PDF is a PDF $p_{u,v}(x, y) = p_w(z)$ such that the marginal densities are $p_u(x)$ and $p_v(y)$:

\[
\int_{\mathbb{R}^n} p_w(x, y) \, dy = p_u(x) \tag{2.1.22}
\]
\[
\int_{\mathbb{R}^n} p_w(x, y) \, dx = p_v(y) \tag{2.1.23}
\]

For any given $p_u(x)$ and $p_v(y)$, there are indeed infinitely many joint PDFs $p_w(x, y)$ that satisfy (2.1.22) and (2.1.23). There are two special extreme cases of interest: in the special case when $p_{u,v}(x, y) = p_u(x)p_v(y)$, we say that $u$ and $v$ are independent. In the special case where $p_{u,v}(x, y)$ is concentrated on the graph of a function $y = M(x)$, i.e.

\[p_w(x, y) = \delta(y - M(x)),\]

we say that $u$ and $v$ are deterministically coupled because $v$ can be expressed as a (deterministic) transformation of $u$. Statistically speaking, the pair $(u, v)$ has been reduced to a pair $(u, M(u))$ that displays the same statistical behavior. By drawing samples of the joint vector $w$, one can determine (at least approximately) if $u$ and $v$ are independent, deterministically coupled, or somewhere in between. This is illustrated for $n = m = 1$ in Figure 2.2.

In general, any collection of random vectors $\{u_j\}_{j=1}^n$ is said to be independent if for the random vector $v = [u_1^\top, \ldots, u_n^\top]^\top$, we have

\[p_v(x) = \prod_{j=1}^n p_{u_j}(x_j)\]

If furthermore $p_{u_j} \equiv p_u$ for some common PDF $p_u(x)$, then the collection $\{u_j\}$ is said to be Independent and Identically Distributed or IID.

Given two real or complex random vectors $u$ and $v$, we define the cross-covariance of $u$ and $v$ by the matrix expected value

\[K_{u,v} = \text{Cov}(u, v) = \langle (u - \bar{u})(v - \bar{v})^\dagger \rangle.\]
The components of this matrix are the covariances between the components of the vectors:

\[
K_{u,v}(i,j) = \text{Cov}(U_i, V_j) = \left< (U_i - \bar{U}_i)(V_j - \bar{V}_j)^\ast \right>.
\]

Defining \( w = [u^\top, v^\top]^\top \), we can write the covariance matrix of \( w \) in terms of the covariances of \( u \) and \( v \) and the cross-covariance as follows:

\[
K_w = \begin{bmatrix}
K_u & K_{u,v} \\
K_{u,v}^\ast & K_v
\end{bmatrix}
\]  \hspace{1cm} (2.1.24)

If \( K_{u,v} \) is the zero matrix, we say that \( u \) and \( v \) are uncorrelated (which does not imply independence, in general, and also does not imply that the individual components of \( u \) or of \( v \) are uncorrelated; the matrix (2.1.24) is block diagonal, but not necessarily diagonal).

In the following discussion, we make a more careful distinction between a random vector and a realization of it by attaching function notation when necessary to the random quantity, i.e. \( u(\omega) \) is random while \( x \) is non-random.

Given two real random vectors \( u(\omega) \in \mathbb{R}^n \) and \( v(\omega) \in \mathbb{R}^m \), let \( w(\omega) = [u(\omega)^\top, v(\omega)^\top]^\top \in \mathbb{R}^{n+m} \)
\( \mathbb{R}^{n+m} \) have joint PDF \( p_w(x, y) \) and marginal PDFs \( p_u(x), p_v(y) \). Then, suppose that we observe \( v(\omega) = y \). We then define the conditional random vector \( u(\omega) | (v(\omega) = y) \) or simply \( u|y \) as the \( n \)-dimensional random vector with PDF

\[
p_{u|y}(x) = \frac{p_w(x, y)}{p_v(y)}
\]

(2.1.25)

The PDF of \( u|y \) is thus proportional to the joint PDF, where we fix \( y \) at the realized value of \( v \).

With the definition (2.1.25), we can define the conditional expectation of \( u \) given \( y \) as

\[
\langle u|y \rangle = \int_{\mathbb{R}^n} x p_{u|y}(x) \, dx
\]

(2.1.26)

As \( y \) is a realization of \( v \), we can also write \( \langle u|y \rangle = \langle u|v(\omega) \rangle \), and by allowing \( \omega \) to vary, we define a random vector \( \langle u|v(\omega) \rangle \) (the randomness is now due to \( v \); we have ‘averaged out’ the randomness in \( u \)). Thus with our local notation convention, \( \langle u|y \rangle \) is non-random, while \( \langle u|v \rangle \) is random; the notation requires care here.

Directly from the definition (2.1.25), the joint PDF, conditional PDFs and marginal PDFs are all connected via Bayes’ theorem:

\[
p_{u|y}(x) p_v(y) = p_w(x, y) = p_{v|x}(y) p_{u|x}(x)
\]

(2.1.27)

The densities in (2.1.27) can be rearranged in various ways to obtain useful relationships. For instance, the most common usage of (2.1.27) is when \( y \) is an observation of \( v \), and we then write

\[
p_{u|y}(x) \propto p_{v|x}(y) p_u(x).
\]

(2.1.28)

The conditional density on the left is called the posterior, while the densities on the right are respectively the likelihood and the prior. Note that as \( y \) is a fixed observation, the likelihood \( p_{v|x} \) is a function of \( x \), not \( y \). The proportionality/normalization factor is the inverse of the marginal PDF \( p_v(y) \), a constant which does not depend on \( x \). The mnemonic of (2.1.28) is that ‘posterior is proportional to likelihood times prior’.
Lastly, we can also connect the marginal PDFs and the conditional PDFs via Bayes rule:

\[ p_u(x) = \int_{\mathbb{R}^m} p_w(x, y) \, dy = \int_{\mathbb{R}^m} p_{u|y}(x)p_v(y) \, dy. \]  

(2.1.29)

with a similar formula holding for \( p_v(y) \). If we use this form of \( p_u(x) \) to compute the expected value of \( M(u) \in \mathbb{R}^k \), we arrive at the law of total expectation:

\[ \langle M(u) \rangle = \int_{\mathbb{R}^n} M(x)p_u(x) \, dx = \int_{\mathbb{R}^n} \int_{\mathbb{R}^m} M(x)p_{u|y}(x)p_v(y) \, dy \, dx = \left\langle \langle M(u) \rangle_{u|v} \right\rangle_v. \]  

(2.1.30)

The formula (2.1.30) is one of the most useful formulas in image science: it allows us to break up complicated averages into nested conditional averages. The formula (2.1.30) shows two levels of nesting, but we can go deeper. For instance, if \( w = (u_1, u_2, u_3) \) are any three joint random vectors and we wish to compute \( \langle M(u_3) \rangle \), we can write

\[ \langle M(u_3) \rangle = \left\langle \left\langle \langle M(u_3) \rangle_{u_3|u_2, u_1} \right\rangle_{u_2|u_1} \right\rangle_{u_1}. \]

This is particularly useful if \( (u_1, u_2, u_3) \) form a Markov chain, so that \( u_3 \) is independent of \( u_1 \) conditional on \( u_2 \), i.e. \( p_{u_3|(u_2, u_1)}(z) = p_{u_3|u_2}(z) \), and hence

\[ \langle M(u_3) \rangle = \left\langle \left\langle \langle M(u_3) \rangle_{u_3|u_2} \right\rangle_{u_2|u_1} \right\rangle_{u_1}. \]

This nesting can continue indefinitely, if desired.

### 2.1.10 Hilbert Spaces of Random Variables and Vectors

A useful abstraction is to consider the set of all (real or complex) \( n \)-dimensional random vectors \( u \equiv u(\omega) \), defined on a common probability space \((\Omega, \mathcal{F}, \mathbb{P})\), such that their second absolute moment is finite:

\[ \langle |u|^2 \rangle = \int_{\Omega} |u(\omega)|^2 \, d\mathbb{P}(\omega) = \int_{\mathbb{R}^n} |x|^2 p_u(x) \, dx < \infty, \]
where \(|x|^2 = \sum_{j=1}^{n} |x_j|^2\). We will call this set \(L^2(\Omega)\), or \(L^2(\Omega; \mathbb{R}^n)\) or \(L^2(\Omega; \mathbb{C}^n)\) if the distinction needs to be made. For any \(u, v \in L^2(\Omega)\), we define an inner product and norm as follows:

\[
(u, v)_{L^2(\Omega)} = \sum_{j=1}^{n} \langle U_j V_j^* \rangle, \quad \|u\|_{L^2(\Omega)} = \sqrt{(u, u)_{L^2(\Omega)}}
\]  

(2.1.31)

where * indicates complex conjugate, if necessary. To see that \((\cdot, \cdot)_{L^2(\Omega)}\) is an inner product, note that \((u, u)_{L^2(\Omega)} = \sum_{j=1}^{n} \langle |U_j|^2 \rangle \geq 0\), with equality if and only if \(u \equiv 0\) almost surely. By linearity of expectation, \((\alpha u + \beta v, w)_{L^2(\Omega)} = \alpha (u, w)_{L^2(\Omega)} + \beta (v, w)_{L^2(\Omega)}\), and finally \((u, v)_{L^2(\Omega)} = (v, u)^*_{L^2(\Omega)}\).

With this inner product, \(L^2(\Omega)\) is complete \([84, \text{Ch. 5}]\), and is hence a Hilbert space. More generally, we define \(L^p(\Omega)\) (explicitly, \(L^p(\Omega; \mathbb{R}^n)\) and \(L^p(\Omega; \mathbb{C}^n)\)) to be the set of real or complex random vectors \(u\) with finite absolute \(p\)th moment, i.e.

\[\langle |u|^p \rangle < \infty\]

where \(|u|\) is the \(n\)-dimensional Euclidean norm. With the norm \(\|g\|_{L^p(\Omega)} = (\langle |g|^p \rangle)^{1/p}\), these spaces are complete and hence form Banach spaces \([84]\); for the most part we will only make use of the case \(p = 2\). Because the probability space \((\Omega, \mathcal{F}, \mathbb{P})\) has total measure 1, we also have the inclusion \(L^p(\Omega) \subset L^q(\Omega)\) if \(p \leq q\); thus, for example, if a random variable has finite moments of order higher than two, it will also have finite moments of order one and two and will hence be an element of \(L^1(\Omega)\) and \(L^2(\Omega)\) \([84]\).

Thinking of random variables and vectors as elements of a Hilbert space is a fruitful abstraction, because it allows one to think in terms of geometry: uncorrelated random variables are 'orthogonal', for instance. The norm \(\|\cdot\|_{L^2(\Omega)}\) gives rise to a form of probabilistic convergence called \textit{mean square convergence}. The Hilbert space structure also permits us to consider orthonormal basis expansions for random variables, leading to the class of so-called \textit{polynomial chaos expansions} \([181, 102]\), a topic that we unfortunately do not pursue further in this work. When we arrive at the description of random processes, we will extend the idea of \(L^2(\Omega; \mathbb{R}^n)\) to consider the Hilbert space of \textit{Hilbert-space-valued} random vectors, for instance \(L^2(\Omega; L^2(I))\), where \(I\) is a spatial or spatiotemporal domain. This will provide ways of handling difficult concepts in the theory of random functions by
retreating to the well understood theory of Hilbert spaces.

Another rich topic in the theory of Hilbert spaces related to random variables, and which space
does not allow us to pursue, is the class of reproducing kernel Hilbert spaces [26] [28] [7].

2.2 Random Processes

In the previous section, we discussed finite-dimensional random vectors, and provided mathemati-
cally rigorous descriptions of their statistical properties using the distribution $P_u$, the PDF $p_u$, the
CF $\psi_u$, and expected values $\langle M(u) \rangle$. There are two fundamental scientific facts that motivate the
generalization of these concepts to random processes:

1. Physical and biological processes are spatiotemporally heterogeneous (i.e. non-constant), and
2. Incomplete information leads to uncertainty.

The first fact leads us to consider quantities which vary in space and time; the second fact leads
us to assume that these functions are random. It is a basic mathematical modeling assumption
throughout this work that all anatomical and physiological processes, and all image data can be
described using random processes. We will also see that clinically relevant probabilistic quantities
which arise in $\mathcal{M}$-PMED, such as

$$ P(\tilde{q} \in E), $$

will require the application of random process theory, since $\tilde{q}$ is defined in terms of a model:
$\tilde{q} = \mathcal{M}(\tilde{f})$, and the input $\tilde{f}$ is best modeled as a random process.

Random processes have an extensive history in both pure mathematics and the applied sciences.
They have been used to model behavior in statistical mechanics [190], [309], optics [26], [186], [110],
quantum mechanics [90], [106], [96], fluid mechanics [205], [206], biology and chemistry [278], [35],
psychology [272], earth sciences [189], [201], [57], hydrology [240], physiology [156], and finance
[153]; they play a central role in systems science and control theory [250], [262], communications
and information [65], and countless other topics. Very recently, random processes have been em-
ployed to quantify uncertainties in clinical oncology applications [303], [127], [63], but the theory of spatiotemporal uncertainty is less well developed in mathematical oncology as in the other fields just named.

Like probability, the mathematical theory of random processes is vast. Our goal is to introduce only the minimal set of concepts necessary to define and simulate uncertainties in $\mathcal{M}$-PMED that arise due to the use of in vivo ECT data. Because physiological and biomedical imaging systems are inherently spatiotemporal, we will require concepts from the theory of multidimensional random processes, sometimes called random fields. We generally take an ‘applied’ approach which seeks to define and manipulate only those mathematical quantities which are readily computable and observable, such as mean and correlation functions and finite-dimensional quantities of interest. We will also commonly make the assumption that random processes can be modeled as generalized random vectors, for instance taking values either in a Hilbert space of functions such as $L^2(\mathcal{I})$ or in a space of generalized functions $\mathcal{F}'$. While we briefly discuss ways to mathematically justify the assumption that a process has $L^2(\mathcal{I})$ realizations, we also feel that such an assumption is frequently justified on physically realistic grounds: real processes will have finite energy. While this assumption occasionally needs to be generalized or specialized, we have found that in every case, treating random processes as generalized random vectors whose realizations are elements of some vector space seems to be a powerful strategy, and we will take this approach. The theory of ‘generalized’ random processes, i.e. those having $\mathcal{F}'$ realizations, was developed extensively by the Gel’fand group [100]. A crucial tool in this study is the characteristic functional, a generalization of the finite-dimensional characteristic function that we discuss in Section 2.2.8.

We begin with a discussion of classical second-order processes, which are random processes for which the realizations are proper functions of the variable $\tau \in \mathcal{I}$ (as opposed to generalized functions, which are functionals on a space of test functions), and such that each evaluation point $\tau$ gives rise to an ordinary real or complex random variable with finite variance, i.e. an element of the Hilbert space $L^2(\Omega)$ presented in Section 2.1.10. A standard mathematical reference for second-order processes is Ash and Gardner [10]; see also Gikhman and Skorokhod [104] and Lord et al. [181]. The engineer’s reference is again Papoulis and Pillai [224]; Barrett and Myers [26] also
discuss much of this content in Chapter 8.

### 2.2.1 Classical Second-Order Random Processes

As in the sections on random variables and vectors, defining a random process requires fixing a background probability space \((\Omega, \mathcal{F}, P)\). As before, this space plays only a formal role as being the abstract ‘generator of randomness’. To define a random process, we also require an index set, denoted generically by \(\mathcal{I}\). We will always assume that \(\mathcal{I}\) is a subset of \(\mathbb{R}^d\), with \(d\) typically being in the range \(d = 1\) (e.g. functions of a time variable \(t\)) to \(d = 4\) (e.g. functions of a 3D spatial coordinate \(r\) and time \(t\)); \(d > 4\) will also be relevant in our discussion of photon-processing imaging systems in Chapter 3. An element of \(\mathcal{I}\) will be denoted with a bold \(\tau\) unless otherwise specified.

A classical random process \(f\) is defined to be a real- or complex-valued function of the two variables \(\omega \in \Omega\) and \(\tau \in \mathcal{I}\):

\[
f : \mathcal{I} \times \Omega \to S,
\]

where \(S \subset \mathbb{R}\) or \(\mathbb{C}\). Straight away, there are two reasonable interpretations of a random process:

1. For each fixed \(\omega \in \Omega\), \(f_\omega \equiv f(\omega) \equiv f(\cdot, \omega)\) is a standard function from \(\mathcal{I}\) to \(S\) called a ‘realization’ of the process.

2. For each fixed \(\tau \in \mathcal{I}\), \(X_\tau \equiv f(\tau, \cdot)\) is a random variable from \(\Omega\) to \(S\). More generally, for any collection \(\{\tau_1, \ldots, \tau_n\} \subset \mathcal{I}\), \(u_{\tau_1, \ldots, \tau_n} = [X_{\tau_1}, \ldots, X_{\tau_n}]^\top\) is an \(n\)-dimensional random vector.

As with random vectors, we occasionally choose to not distinguish the notation between a random process and its realizations unless necessary, such as when conditioning on a particular realization. If there is a danger of confusion, we will typically use a subscript e.g. \(f_j\) or \(f_\omega\) to indicate a realization of the process \(f\), so that ‘let \(f = f_j\’\) will mean explicitly, ‘let \(f_j\) be a realization of \(f\’\).

The first interpretation of a random process given above can also be thought of as the ensemble interpretation. The set \(\{f(\omega) : \omega \in \Omega\}\) is a collection of realizations of the process, which could be called the ensemble of realizations; refer to Figure 2.3. Imagine, for example, that \(\Omega\) parameterizes all possible patients who might present with a particular condition (e.g. a cough). Then, if
every patient $\omega$ gives rise to a realization $f(\omega)$ (say for example $f(t, \omega) \equiv f(\mathbf{r}, t, \omega)$ is their X-Ray attenuation coefficient, measured in Hounsfield units), then the ensemble $\{f(\omega) : \omega \in \Omega\}$ is a representation of ‘all possible X-Ray attenuation coefficient maps arising from patients with a cough’. Note that the population need not be ‘real’ in any sense: it can be a virtual, abstract population of patients that present with a cough. This Virtual Patient Ensemble, or VPE, can either model a true population as just described, or can be defined for a particular patient by conditioning on available patient-specific data; we discuss such patient-specific ensembles in Chapters 3 and 4.

Figure 2.3: An illustration of the ensemble interpretation of a two-dimensional random process $f(\omega) = f(\mathbf{r}, \omega) = f(x, y, \omega)$, and the joint PDFs of two-point finite-dimensional sample vectors $\mathbf{u}_{r_1, r_j}$. Note that the PDF for $\mathbf{u}_{r_1, r_2}$ reflects the fact that $f(r_1)$ and $f(r_2)$ are highly correlated ($r_1$ is close to $r_2$), while the PDF of $\mathbf{u}_{r_1, r_3}$ reflects the fact that $f(r_1)$ and $f(r_3)$ are likely independent ($r_1$ and $r_3$ are further apart than the correlation length). Adapted from [57].

To illustrate the concept further, suppose that $\mathcal{I} = [a, b]$ is an interval of real numbers, $S = \mathbb{R}$ and $t \in \mathcal{I}$ represents time; then $f(t, \omega)$ represents a random real-valued function of time. If $\mathcal{I} = V \subset \mathbb{R}^3$, $S = \mathbb{R}$, and $\tau = \mathbf{r} \in V$ represents a spatial coordinate, then $f(\mathbf{r}, \omega)$ represents a random real-valued scalar field such as temperature or concentration. If $\mathcal{I} = V \subset \mathbb{R}^3 \times [a, b]$ and $\tau = (\mathbf{r}, t)$ represents a spatiotemporal coordinate, then $f(\mathbf{r}, t, \omega)$ represents a time-varying spatial
random scalar field; later, we will allow \( S \subseteq \mathbb{R}^m \), to allow \( f \equiv f(r, t, \omega) \) to represent a time-varying random vector field, for instance a velocity or electric field, or some other collection of multiple random scalar fields (such as temperature, pressure and concentration). More general choices of \( I \) will arise in Chapter 3 and while more general state spaces \( S \) are certainly possible, they are not necessary for our purposes. If a random process \( f \) models a real physiological process such as drug susceptibility, we have coined the term *Physiological Random Process* or *PRP* to indicate this context.

We will assume that all classical random processes are at least *second-order*, which means that for each fixed \( \tau \in I \), the random variable \( X_\tau = f(\tau, \cdot) \) has finite second moment, i.e. \( \langle |X_\tau|^2 \rangle < \infty \), and is thus an element of the Hilbert space \( L^2(\Omega) \) (see Section 2.1.10). Note that this does not immediately imply that the *realizations* are square-integrable as functions of \( \tau \): a random process which assumes the constant 1 for all \( \tau \in \mathbb{R}^n \) and for all \( \omega \in \Omega \) is second-order, but certainly not square-integrable on \( \mathbb{R}^n \). It is also not the case that all interesting random processes are classical second-order processes. For example, an extremely useful class of random processes are the *point* processes, whose realizations are sums of Dirac deltas, and thus the point-sample variables \( X_\tau \) are not even well-defined. We treat such *generalized* random processes in Section 2.2.6.

We will frequently make the modeling assumption that realizations are square-integrable functions of \( \tau \): see Sections 2.2.4 and 2.2.6. We will make the distinction by reserving the label ‘second-order’ for processes with finite second moment for all \( \tau \), and ‘square-integrable’ for processes whose realizations are (almost surely) square-integrable functions of \( \tau \). The second-order assumption is what allows the definition of the mean and covariance functions of a process, which we define in Section 2.2.3.

One of the central issues in the employment of random processes in practical applications is the need to specify the statistical properties of the process. As with random variables and random vectors, it is almost never the case that we know the underlying \( \Omega \), nor the explicit function mapping \( (\tau, \omega) \) to \( f(\tau, \omega) \); rather, we are usually only able to describe the joint statistics of the process. With a finite-dimensional random vector \( u \), say \( u \in \mathbb{R}^n \), we were able to describe its statistics using either the distribution \( P_u \), the PDF \( p_u(x) \), or the CF \( \psi_u(\xi) \). We also discussed
moments of random vectors, though we gave the example of the lognormal to demonstrate that
knowledge of the moments of every order is not sufficient to fully constrain a random vector’s
statistics. Note that both the PDF and the characteristic function are functions of \( n \) variables for
an \( n \)-dimensional random vector. We would like to extend the notions of \( P_u, p_u(x), \psi_u(\xi) \) and
the moments to random processes, but this will require some care: if \( \mathcal{I} \) is a continuum set (e.g. an
open subset of \( \mathbb{R}^n \)), the realizations of a stochastic process \( f \) are now ostensibly elements of some
infinite-dimensional space, and thus any PDF or CF, if one were to exist, would necessarily be a
function of an infinite-dimensional variable, i.e. a functional. If we wish to consider a probability
measure analogous to \( P_u \), it would necessarily be defined on subsets of some infinite-dimensional
space. Lastly, if we wish to consider moments by generalizing (2.1.15) or (2.1.16), the index will
necessarily also become continuous (instead of a covariance matrix, we will have a covariance
function of two variables). We will see that such descriptions of a random process are all possible
in principle, but somewhat delicate, and some (namely moments and the generalized CF) tend to
be much easier to manipulate than the generalizations of \( P_u \) and \( p_u(x) \). We begin by reducing the
problem of specifying statistics to one that we already understand by extracting finite-dimensional
random vectors from a random process.

2.2.2 Finite-Dimensional Densities

Given a random process \( f(\tau, \omega) \), we have that for each fixed \( \tau \in \mathcal{I} \), \( X_\tau \) is a standard real-valued
random variable. If we select multiple sample points, say \( \tau_1, \ldots, \tau_n \), we furthermore have that
\( u_{\tau_1:\tau_n} \equiv [X_{\tau_1}, \ldots, X_{\tau_n}]^T \) is a random vector in \( \mathbb{R}^n \) (or \( \mathbb{C}^n \)). The PDF of this random vector is
denoted \( p_{\tau_1:\tau_n}(x) \), and is called a finite-dimensional density of \( f \). For example, when \( n = 1 \), we
have a collection of one-point PDFs \( p_\tau(x) \), i.e. a one-dimensional PDF for each sample point \( \tau \in \mathcal{I} \).
For \( n = 2 \), we have the collection of two-point PDFs, \( p_{\tau_1,\tau_2}(x, y) \), i.e. a two-dimensional PDF for
each pair \( \{\tau_1, \tau_2\} \subset \mathcal{I} \).

The finite-dimensional densities must be defined in a consistent manner: re-arrangement of
indices should not affect the resulting distribution, and marginalization over any collection of vari-
ables should return the same result as if the remaining variables were chosen at the start [10].
If a collection \( \{ p_{\tau_1, \tau_n} \} \) is known for all \( n \in \{1, \ldots \} \) and for all collections \( \{ \tau_1, \ldots, \tau_n \} \subset I \), and they satisfy these consistency conditions, the process is said to be fully characterized, and a stochastic process does indeed exist that has these as finite-dimensional densities; this is the Daniell-Kolmogorov theorem \([30]\) \([10]\) \([181]\). The first difficulty with this construction of a stochastic process is the immense quantity of information required: if \( I \) is an open subset of \( \mathbb{R}^d \) (say, an interval or rectangular domain), then we require uncountably many functions of one variable, i.e. a function \( p_\tau(x) \) for each \( \tau \in I \); uncountably many functions of two variables, i.e. a function \( p_{\tau_1, \tau_2}(x, y) \) for each pair of \( \tau_1, \tau_2 \in I \), uncountably many of three variables, and so on. Despite the consistency conditions, only in very special cases is this exercise tractable. The second difficulty with the finite-dimensional distributions is that the process given by the Daniell-Kolmogorov theorem is not guaranteed to have any desirable realization properties - it may need to be modified in order to be (for example) mean square continuous \([104]\). We must, therefore, seek more practical ways to describe how random processes behave; one such description is through moment functions.

### 2.2.3 Moment Functions and Realization Properties

Suppose that \( f : I \times \Omega \to S \) is a real- or complex-valued, second-order stochastic process, where \( I \subset \mathbb{R}^d \). Then, we have that \( X_\tau = f(\tau, \cdot) \in L^2(\Omega) \) for every \( \tau \in I \), and so the following three functions are well-defined:

\[
\mu_f(\tau) = \langle X_\tau \rangle, \tag{2.2.1}
\]
\[
k_f(\tau, \tau') = \langle (X_\tau - \mu_f(\tau))(X_{\tau'} - \mu_f(\tau'))^* \rangle, \text{ and} \tag{2.2.2}
\]
\[
r_f(\tau, \tau') = \langle X_\tau X_{\tau'}^* \rangle, \tag{2.2.3}
\]

where \( * \) denotes complex conjugate, when necessary. They are called, respectively, the \textit{mean function}, the \textit{covariance function}, and the \textit{correlation function} of the process. Note that we have the identity

\[
k_f(\tau, \tau') = r_f(\tau, \tau') - \mu_f(\tau)\mu_f^*(\tau')
\]
Each of $\mu_f$, $k_f$ and $r_f$ is a real- or complex-valued function, depending on whether the process is real- or complex-valued.

The mean and covariance function are defined so as to give the mean vector and covariance matrix of the finite-dimensional sample vector $u_{\tau_1:\tau_n}$:

$$u_{\tau_1:\tau_n} = [\mu_f(\tau_1), \ldots, \mu_f(\tau_n)]^T$$

(2.2.4)

$$K_{\tau_1:\tau_n}(i,j) = \text{Cov}(X_{\tau_i}, X_{\tau_j}) = k_f(\tau_i, \tau_j)$$

(2.2.5)

The mean and covariance function provide all statistical information about how a process behaves on average and how it correlates with itself when evaluated at any two points. Thus, given the covariance function, one can start to make quantitative, probabilistic statements about the behavior of $f(\tau_2)$ given information about $f(\tau_1)$. Because the mean and covariance do not fully specify the densities $p_{\tau_1,\tau_2}(x,y)$, nor do they specify any statistics involving three or more sample points, the mean and correlation function cannot, in general, give the full probabilistic description of the process (which again, in theory, would require the joint distribution of every finite-dimensional sample vector $u_{\tau_1:\tau_n}$). There are notable exceptions to this: for instance, just as in the finite-dimensional case, Gaussian processes are completely determined by their mean and covariance functions, and Poisson point processes, despite being generalized processes, are fully determined by their mean function. We discuss both Gaussian processes and Poisson point processes in more detail in Section 2.6.

Like the covariance matrix, the covariance function is symmetric and non-negative definite. In fact, this follows again directly from the definition: for any collection $\{\tau_1, \ldots, \tau_n\} \subset \mathcal{I}$ and for any $z \in \mathbb{C}^n$ (or $\mathbb{R}^n$) we have that

$$\sum_{j,k=1}^{n} z_j^* k_f(\tau_j, \tau_k) z_k = z^* K_{\tau_1:\tau_n} z \geq 0$$

where $K_{\tau_1:\tau_n}$ is the covariance matrix defined in (2.2.5). Note that the mean function depends on a single input, $\tau$, while the covariance and correlation functions depend on two inputs, $\tau$ and $\tau'$; higher-order moment functions, which depend on more evaluation points, can also be defined. If
$X_\tau \in L^p(\Omega)$ for every $\tau$, then the following function is well-defined for any collection $\tau_1, \ldots, \tau_k \in I$ with $k \leq p$ \footnote{humans, animals and imaging systems are bounded in space and time, after all.}: 

$$M_f^{(k)}(\tau_1, \ldots, \tau_k) = \langle X_{\tau_1} \cdots X_{\tau_k} \rangle. \quad (2.2.6)$$

Centered versions of \footnote{humans, animals and imaging systems are bounded in space and time, after all.} can also be defined.

The moment functions can be used to make (probabilistic) statements about the behavior of the realizations of a process $f$. For example, if the mean and covariance functions are both continuous, the realizations of the process are \textit{mean square} continuous, which means that $X_{\tau_j} \to X_\tau$ if $\tau_j \to \tau$, the convergence being in $L^2(\Omega)$. (see e.g. \cite{10}). In this spirit, we show briefly that if the covariance function is integrable on the diagonal, the realizations of the process are almost surely (which means with probability 1) square-integrable with respect to the parameter $\tau$: thus a property of the moments of $f$ (integrability on the diagonal) translates to a property of the realizations of $f$ (square integrability). So, let $f$ be a second-order process, and assume (without loss of generality) that $\mu_f \equiv 0$. Suppose further that

$$\int_I k_f(\tau, \tau)d\tau < \infty \quad (2.2.7)$$

Then, we have by the general Tonelli-Fubini theorem \footnote{humans, animals and imaging systems are bounded in space and time, after all.} that

$$\langle \|f(\tau, \omega)\|^2_{L^2(I)} \rangle = \int_I \int_\Omega |f(\tau, \omega)|^2 d\tau d\mathbb{P}(\omega)$$

$$= \int_I \int_\Omega |f(\tau, \omega)|^2 d\mathbb{P}(\omega) d\tau$$

$$= \int_I \langle f(\tau, \omega) f(\tau, \omega)^* \rangle d\tau$$

$$= \int_I k_f(\tau, \tau) d\tau$$

By assumption, the last quantity is finite; hence $\|f(\tau, \omega)\|^2_{L^2(I)}$ must be finite for all $\omega \in \Omega$ outside a set of probability zero. The condition \footnote{humans, animals and imaging systems are bounded in space and time, after all.} is realistic for our purposes, because it is essentially always the case in physiology and image science that $I$ is a compact subset of $\mathbb{R}^d$. Thus if we
simply assume that $k_f(\tau, \tau')$ is continuous (or bounded, even), we’ll certainly have (2.2.7).

This proposition justifies the statement that any second-order process with covariance satisfying (2.2.7) can be treated as a Hilbert-space-valued random vector, i.e. we can say $f : \Omega \to L^2(\mathcal{I})$. This provides a theoretical justification to the natural assumption that the realizations of a random process which models a physical system should have finite energy, as discussed in the introduction to this section.

We provide two more definitions before moving on. A process $f$ is said to be wide-sense stationary or WSS if $\mathcal{I} = \mathbb{R}^d$, the mean function is constant i.e. $\mu_f(\tau) \equiv \mu_f$, and the covariance function is shift-invariant i.e. (abusing notation) $k_f(\tau, \tau') \equiv k_f(\tau - \tau')$. A WSS process $f$ is said to be ergodic in mean if we have

$$\lim_{T \to \infty} \frac{1}{(2T)^d} \int_{B_T} f(\tau, \omega) \, d\tau = \mu_f$$

where $B_T = [-T, T]^d$ is a box of side length $2T$ and $(2T)^d$ is the volume of this box. Because the quantity on the left is random (it depends on $\omega$) while the quantity on the right is not, we must specify the sense of convergence; convergence in $L^2(\Omega)$ is typical, but can also be almost sure or in probability. Note that perhaps the most unrealistic requirement of stationarity (and hence ergodicity) is that $\mathcal{I} = \mathbb{R}^d$; all physical objects and imaging systems are bounded, hence no physical object can truly be a realization of a stationary process. We say that $f$ is locally wide-sense stationary if on some set $\mathcal{I}' \subset \mathcal{I}$, we have that $\mu_f(\tau)$ is constant on $\mathcal{I}'$ and some $\tilde{k}(\tau)$ exists such that $k_f(\tau, \tau') = \tilde{k}(\tau - \tau')$ for all $\tau, \tau' \in \mathcal{I}'$.

### 2.2.4 Hilbert-space-valued Random Processes

In light of the discussion in the above section, we will assume in this work that any classical second-order random process can be considered as a Hilbert-space-valued random vector, i.e. a random element taking values in the range space $L^2(\mathcal{I})$. In other words we are assuming that the process $f$, a-priori defined only as a measurable function $f : \mathcal{I} \times \Omega \to S$, can instead be thought of as a
Borel measurable function

\[ f : \Omega \rightarrow L^2(\mathcal{I}), \quad \omega \mapsto f(\tau, \omega) \]

We will furthermore assume, in light of (2.2.7) and the subsequent proposition, that

\[ \langle \| f(\tau, \omega) \|^2_{L^2(\mathcal{I})} \rangle < \infty \]

In other words, not only are realizations square-integrable functions of the parameter \( \tau \), but the process is also second order in this generalized sense. As in Section 2.1.10 the space of all such random vectors forms a Hilbert space, which is denoted \( \mathcal{H} = L^2(\Omega; L^2(\mathcal{I})) \) \[181\] \[104\]. Unless otherwise specified, all classical random processes are assumed to be elements of this \( \mathcal{H} \) for an appropriate index set \( \mathcal{I} \).

The space \( \mathcal{H} \) has an inner product defined by nesting expectation and the usual inner product:

\[ (f, g)_{\mathcal{H}} = \left( \langle f, g \rangle_{L^2(\mathcal{I})} \right) = \left( \int_{\mathcal{I}} f(\tau, \omega) g(\tau, \omega)^* \, d\tau \right)_\omega. \quad (2.2.8) \]

This inner product also gives rise to a norm, which could be called the Root Mean Square Error or RMSE norm:

\[ \| f \|_{\mathcal{H}} = \sqrt{(f, f)_{\mathcal{H}}} = \left( \int_{\mathcal{I}} |f(\tau, \omega)|^2 \, d\tau \right)_\omega^{1/2} = \sqrt{\left( \langle \| \cdot \|_{L^2(\mathcal{I})}^2 \rangle \right)_\omega} \quad (2.2.9) \]

The norm \( (2.2.9) \) has a simple interpretation: it computes the squared \( L^2(\mathcal{I}) \) norm of each realization of the process, then averages the result over the ensemble (followed by a square root).

Two immediate consequences of the assumption that \( f \in \mathcal{H} \) arise. First, note that if we fix any ‘test function’ \( \phi \in L^2(\mathcal{I}) \), a process \( f \in \mathcal{H} \) gives rise to a random variable \( X_\phi \in L^2(\Omega) \) via the inner product:

\[ X_\phi = (f, \phi)_{L^2(\mathcal{I})}. \quad (2.2.10) \]
Thus, we can alternately think of $f$ as an operator
\[ f : L^2(\mathcal{I}) \to L^2(\Omega), \quad \phi \mapsto X_\phi. \]

Second, if $f \in \mathcal{H}$, the covariance function $k_f$ gives rise to a positive-definite, self-adjoint covariance operator $C_f$ which is well-defined on $L^2(\mathcal{I})$; in fact, it is also trace-class (also known as nuclear \[100\] \[181\] \[121\] \[104\]). The operator $C_f$ is defined for $\phi \in L^2(\mathcal{I})$ via
\[
(C_f \phi)(\tau) = \int_\mathcal{I} k_f(\tau, \tau') \phi(\tau') d\tau'
\] (2.2.11)

Note that the covariance operator can be used to compute the covariance between the random variables $X_\phi$ and $X_\psi$ defined in (2.2.10) (assume that $f$ is real-valued and $\bar{X}_\phi = \bar{X}_\psi = 0$ without loss of generality):

\[
\text{Cov}(X_\phi, X_\psi) = \langle (f, \phi)_{L^2(\mathcal{I})}, (f, \psi)_{L^2(\mathcal{I})} \rangle = \left\langle \int_\mathcal{I} \int_\mathcal{I} f(\tau) \phi(\tau) f(\tau') \psi(\tau') d\tau d\tau' \rightangle
= \int_\mathcal{I} \int_\mathcal{I} \langle f(\tau) f(\tau') \rangle \phi(\tau) \psi(\tau') d\tau d\tau'
= \int_\mathcal{I} \int_\mathcal{I} k_f(\tau, \tau') \phi(\tau) \psi(\tau') d\tau d\tau'
= (C_f \phi, \psi)
\]

This calculation can also be used to define the covariance operator $C_f$ directly. One defines the bilinear operator
\[
B_f(\phi, \psi) = \text{Cov}(X_\phi, X_\psi),
\]
then an application of the Schwartz kernel theorem (a generalization of the Riesz representation theorem) proves that there exists an operator $C_f$ such that $B_f(\phi, \psi) = (C_f \phi, \psi)$, and this operator is easily shown to be a valid covariance operator \[100\].

That $C_f$ is positive definite means that $(C_f \phi, \phi)_{L^2(\mathcal{I})} \geq 0$ for all $\phi \in L^2(\mathcal{I})$, and the proof that $C_f$ is positive definite is given in the line of calculations above: working backwards, $(C_f \phi, \phi) =$
\[ \text{Cov}(X_\phi, X_\phi) = \text{Var}(X_\phi) \geq 0. \] To see why \( C_f \) is self-adjoint, it suffices to recall that \( k_f \) is, by definition (2.2.2), Hermitian symmetric, that is, \( k_f(\tau', \tau) = k_f(\tau, \tau') \). The proof that \( C_f \) is trace class is found in [181] and [104]. Note that if \( f \) were WSS, the covariance operator (2.2.11) would be a convolution operator, because then \( k_f(\tau, \tau') \equiv k_f(\tau - \tau') \). However, (nonzero) WSS processes cannot give rise to square-integrable realizations, and are thus not elements of \( \mathcal{H} \).

Since trace-class operators are compact [232, Theorem VI.21], the spectral theorem of compact self-adjoint operators implies that \( C_f \) has a complete orthonormal system of eigenfunctions with real eigenvalues. In other words, there exists a countably infinite family \( \{e_j\}_{j=1}^{\infty} \subset L^2(\mathcal{I}) \) and \( \{\lambda_j\}_{j=1}^{\infty} \subset \mathbb{R} \) such that

\[ C_f e_j = \lambda_j e_j \quad \text{and} \quad (e_j, e_k)_{L^2(\mathcal{I})} = \delta_{jk}. \] (2.2.12)

That \( C_f \) is positive-definite implies that the corresponding eigenvalues are nonnegative, i.e. \( \lambda_j \geq 0 \); that \( C_f \) is trace class implies the sequence of eigenvalues is summable i.e.

\[ \text{tr}(C_f) = \sum_{j=1}^{\infty} \lambda_j < \infty \] (2.2.13)

It can be shown that \( \text{tr}(C_f) \) is also equal to the integral (2.2.7).

We will use the basis (2.2.12) in the next section to compute a useful orthonormal expansion for realizations of a process \( f \in \mathcal{H} \).

### 2.2.5 Series Expansions of Processes and Karhunen-Loève Analysis

As we discussed at the end of Section 2.2.1, one of the primary challenges we face in working with random processes is having an efficient manner in which to represent the statistics and realizations of a process. As we have mentioned, it can be essentially impossible to define all the many-point density functions \( p_{\tau_1:\tau_n}(x) \). However, owing to the assumption that the realizations of a process \( f \in \mathcal{H} \) are \( L^2(\mathcal{I}) \) functions, we can, regardless of the statistical properties of the process, expand realizations in either a basis or frame \( \{\phi_j\}_{j=1}^{\infty} \) for \( L^2(\mathcal{I}) \). To this end, we briefly review some notation and concepts from basis and frame theory; see also [26] and [58].
Given any sequence $\{\phi_j\}_{j=1}^\infty \subset L^2(I)$, we define two operators associated with it:

$$
\Phi : L^2(I) \to \ell^2(\mathbb{N}), \quad (\Phi f)_j = (f, \phi_j)_{L^2(I)}
$$

$$
\Phi^\dagger : \ell^2(\mathbb{N}) \to L^2(I), \quad (\Phi^\dagger z)_j(\tau) = \sum_{j=1}^\infty z_j \phi_j(\tau)
$$

The operators $\Phi$ and $\Phi^\dagger$ are, respectively, the analysis and synthesis operators of the sequence. If $\{\phi_j\}_{j=1}^\infty$ is an orthonormal basis, then $\Phi$ is unitary and hence $\Phi^\dagger \Phi = \Phi \Phi^\dagger = I$, the identity operator. In terms of a series expansion, we would have for every $u \in L^2(I)$,

$$
u = \Phi^\dagger \Phi u = \sum_{j=1}^\infty (u, \phi_j)_{L^2(I)} \phi_j
$$

If $\{\phi_j\}_{j=1}^\infty$ is not an orthonormal basis, but is either a Riesz basis or frame, then there exists a dual (or analysis) sequence $\{\tilde{\phi}_j\}_{j=1}^\infty$ such that $\Phi^\dagger \tilde{\Phi} = \tilde{\Phi} \Phi^\dagger = I$. In terms of a series expansion, we would have

$$
u = \Phi^\dagger \tilde{\Phi} u = \sum_{j=1}^\infty (u, \tilde{\phi}_j)_{L^2(I)} \phi_j
$$

The dual sequence is given explicitly in terms of the expansion sequence via $\tilde{\phi}_j = (\Phi^\dagger \Phi)^{-1} \phi_j$.[58] Perhaps the most well-known non-orthonormal expansions for $L^2(I)$ arise from the definition of multiresolution-analysis-based wavelet frames [26] [185] [305].

Regardless of the analysis and synthesis sequence, if $f \in \mathcal{H}$, is a square-integrable random process, expanding the realizations in a series of the form (2.2.14) gives rise to a random series expansion of the form

$$
f(\tau, \omega) = \sum_{j=1}^\infty Z_j(\omega) \phi_j(\tau), \quad Z_j(\omega) = (f(\omega), \tilde{\phi}_j)_{L^2(I)}
$$

[58] we assume the sequence is a Bessel sequence [58] so that the analysis and synthesis operators are bounded
Or, in other words, we can express the process as

\[ f = \Phi^\dagger z \]

where \( \Phi^\dagger \) is the synthesis operator for the chosen expansion sequence and \( z \) is a random \( \ell^2(N) \) sequence. Note that the expansion functions in (2.2.15) are non-random: the statistical properties of \( f \) are now fully contained in the random sequence \( z = [Z_1, Z_2, \ldots] \in \ell^2(N) \). In principle, describing the joint statistics of \( z \) is a simpler endeavor than describing the full joint statistics of \( f \), however we must still potentially consider infinitely many PDFs, i.e. we must describe \( p_{Z_j}(x) \) for all \( j \), and \( p_{Z_j, Z_k}(x, y) \) for all \( j, k \), etc., but at least now we have only countably many PDFs to describe. We will now see that by using eigenanalysis of the covariance operator \( C_f \), we can de-correlate the sequence \( z \), so that \( \langle Z_j Z_k \rangle = \sigma^2_j \delta_{jk} \); if the process is Gaussian, this is sufficient to guarantee that the entire sequence is independent, i.e. any collection of \( Z_j \)'s are independent.

Suppose that \( f \in \mathcal{H} \) is a random process with \( \mu_f \equiv 0 \). Then, as discussed in Section 2.2.4, the covariance operator \( C_f \) is self-adjoint and trace class (and therefore compact), and thus there exists an orthonormal basis \( \{e_j\}_{j=1}^\infty \) of \( L^2(\mathcal{I}) \) consisting of eigenfunctions of \( C_f \), and the eigenvalues \( \{\lambda_j\}_{j=1}^\infty \) are nonnegative. Thus, every realization of \( f \) can be written as

\[
f(\tau, \omega) = \sum_{j=1}^\infty Z_j(\omega)e_j(\tau), \quad Z_j(\omega) = (f(\omega), e_j)_{L^2(\mathcal{I})} = \int_\mathcal{I} f(\tau, \omega)e_j(\tau) \, d\tau. \tag{2.2.16}
\]

For now, the convergence is in the Hilbert space \( \mathcal{H} \) [181]. This expansion is called the Karhunen-Loève Expansion or KLE [26] [154] [180] [164], and it has several useful properties that distinguish it from other expansions of the sort (2.2.15):

1. The random variables \( Z_j \) defined in (2.2.16) are mean-zero and pairwise uncorrelated, and their variances are given by the eigenvalues of the operator \( C_f \), that is,

\[
\langle Z_j \rangle = 0, \quad \langle Z_j, Z_k \rangle_{L^2(\Omega)} = \langle Z_j Z_k^* \rangle = \lambda_j \delta_{jk}.
\]

Note that, as usual, this does not imply that the \( Z_j \) are pairwise independent unless the
process is Gaussian. This is why the KLE is called biorthogonal: both the KL basis functions \( \{e_j\} \) are orthogonal (in \( L^2(I) \)) and the random variables \( Z_j \) are orthogonal (in \( L^2(\Omega) \)).

2. If the covariance function is continuous and \( I \) is compact, the series in (2.2.16) actually converges in a much stronger sense: for each fixed \( \tau \in I \), it converges in \( L^2(\Omega) \) (i.e. in mean square), and the rate of convergence is uniform in \( \tau \) \( [181] \).

3. The KLE gives an optimal finite-dimensional linear approximation to the process \( f \) in the following sense. For any \( n > 0 \), the optimal approximation (in terms of the \( H \) norm) to \( f \) of the form

\[
f_n = \Phi_n^\dagger z = \sum_{j=1}^n Z_j(\omega)\phi_j
\]

is given by projection of \( f \) onto the span of the first \( n \) KL basis functions, i.e. we should choose \( \phi_j = e_j \) and \( Z_j = (f, e_j)_{L^2(I)} \), where \( e_j \) are the KL basis functions, ordered according to their corresponding eigenvalues; see e.g. [185].

Note that if the mean of \( f \) is nonzero, one simply sets \( f' = f - \mu_f \) and applies the KLE to \( f' \); we can add back the mean to obtain

\[
f(\tau, \omega) = \mu_f(\tau) + \sum_{j=1}^\infty Z_j(\omega)e_j(\tau) = \mu_f(\tau) + \left( \mathcal{E}^\dagger z(\omega) \right)(\tau) \tag{2.2.17}
\]

Since \( \langle Z_j Z_k^* \rangle = \lambda_j \delta_{jk} \), another way to write (2.2.17) is

\[
f(\tau, \omega) = \mu_f(\tau) + \sum_{j=1}^\infty \sqrt{\lambda_j} \xi_j(\omega)e_j(\tau) \tag{2.2.18}
\]

where \( \langle \xi_j \xi_k^* \rangle = \delta_{jk} \) are uncorrelated random variables with variance one.

The KLE, written in the form (2.2.18), facilitates a very straightforward method to quickly construct and simulate random processes \( f \in \mathcal{H} \). First, we choose an orthonormal basis for \( L^2(I) \), for instance a Fourier basis, orthogonal polynomial basis, or wavelet basis. Then, we choose a summable sequence \( \lambda = (\lambda_1, \lambda_2, \ldots) \), with \( \lambda_j \geq 0 \) for all \( j \). Next, fix an integer \( m > 0 \), a mean
function $\mu_f(\tau)$, and define a random vector $Z \in \mathbb{C}^m$ which has mean zero and covariance matrix $K = I_{m \times m}$. Then, by setting

$$f(\tau, \omega) = \mu_f(\tau) + \sum_{j=1}^{m} \sqrt{\lambda_j} Z_j(\omega) \phi_j(\tau),$$

we have a rapid method to draw approximate samples from $f$. Note that if the covariance operator $C_f$ is finite rank, an expansion of the form (2.2.19) will actually be exact, since then only finitely many eigenvalues of $C_f$ will be nonzero. We will use expansions of the sort (2.2.19) to simulate random processes in Section 2.6.

If we wish to compute a KLE, but start out only knowing the covariance function $k_f$ or covariance operator $C_f$, we must first solve the eigenvalue problems $C_f \phi_j = \lambda_j \phi_j$; in some cases these eigenvalue problems can be solved explicitly (for instance, if $C_f$ is a convolution operator on a finite domain with periodic boundary, the eigenfunctions of $C_f$ are Fourier modes), but this is not typical: it may be necessary to compute numerical approximations to the eigenvalues and eigenfunctions of $C_f$. We do not address this problem here; see e.g. [181] [117]. Note also that if only $\mu_f$ and $k_f$ are known, and the process is not known to be Gaussian, the random variables $Z_j$ in the KLE may have higher-order correlations that cannot be described. While the KL basis functions and the decay sequence $\lambda$ can be ascertained, the full joint statistics of $Z_j$ require more information than is provided by $\mu_f$ and $k_f$ alone: in principle we require the full distribution of $f$ to describe the joint statistics of $Z_j$.

While the KLE is certainly elegant and indeed useful in many situations, there are other situations where it is perhaps suboptimal. As we just mentioned, if higher-order statistics of $f$ are unknown, the full joint statistics of the $Z_j$ will also be unknown, and treating this sequence as independent may not reflect the true behavior of $f$. Furthermore, while we showed above that the KLE gives rise to optimal finite-dimensional linear approximations of a random process, for processes which are non-Gaussian and/or non-stationary, it is possible to show that the KLE representation is, in a sense, suboptimal. Indeed, one of the flagship results of wavelet-type analysis is that nonlinear approximations – for instance, those based on thresholding of expansion coefficients, the components of $Z$ need only be uncorrelated, not necessarily independent.
as opposed to truncation of series – of non-stationary random processes can vastly outperform the KLE in terms of efficient representation: sparsity replaces linearity as a paradigm. These results are discussed for instance in Mallat [185] and Unser [274]; we unfortunately cannot discuss the rigorous aspects of this issue any further due to space considerations. For our purposes, the convenience of fixing a particular basis – which may or may not be the KLE, and may or may not be optimally efficient for some particular process – will outweigh the potential benefits gained by using a more exotic basis that is fine-tuned for the representation of some particular process.

2.2.6 Generalized Random Vectors and Generalized Processes

We have already discussed an abstract way of thinking about square-integrable random processes: the Hilbert space $\mathcal{H}$, introduced in Section 2.2.4, consists of generalized random vectors i.e. functions $f : \Omega \rightarrow L^2(I)$. The choice of $L^2(I)$ was convenient, but not the only choice of ‘target’ vector space that we can make. In general, given any topological vector space $\mathcal{X}$ (see [212], [100]), one can define the Borel sigma algebra $\mathcal{B}$, and hence consider probability measures on $\mathcal{X}$. So, given a probability space $(\Omega, \mathcal{F}, P)$, we define a generalized random vector as a (measurable) function

$$f : \Omega \rightarrow \mathcal{X} \quad (2.2.20)$$

We have already given the example where $\mathcal{X} = L^2(I)$, the standard Hilbert space of functions $f(\tau)$ on some domain $I \subset \mathbb{R}^d$. Another classic example where $\mathcal{X}$ is not a Hilbert space is the so-called Wiener path space, where $\mathcal{X} = C([0,1])$, the space of continuous functions on the interval $[0,1]$, with norm given by $\| \cdot \|_{\infty} = \sup_{t \in [0,1]} |f(t)|$. This space is used in the study of Brownian motion – intuitively, it is of physical interest to show that the mathematical construction of Brownian motion (which was first given by Einstein [85] and generalized by Smoluchowski [258]) leads to physically realistic (in this case meaning continuous) paths.

In many instances, the vector space $\mathcal{X}$ is actually a dual space of some other vector space, i.e. we would write $\mathcal{X} = \mathcal{T}'$ where $\mathcal{T}$ is a space of test functions. For example, generalized functions (in the sense of Schwartz) are defined as continuous linear functionals on a vector space of test functions, usually taken to be the set $\mathcal{D}$ of smooth, compactly supported functions [212], [242],
A generalized function \( f \equiv f(\phi) \) is an element of \( \mathcal{F}' \), the linear dual space of \( \mathcal{F} \); this is also a vector space, and so \((2.2.20)\) still makes sense.

Let \( \mathcal{F} \) be a space of test functions, for instance the Schwartz class \( \mathcal{S} \) (sometimes called the good functions \([26][177]\)), which consists of smooth functions \( f \) which together with all their derivatives decay rapidly, or the space \( \mathcal{D} = C^\infty_0 \) of smooth, compactly supported test functions. Each test function \( \phi \in \mathcal{F} \) may be either real-valued, complex-valued, or vector-valued (to allow for coupled processes, as we discuss in Section 2.2.9). A continuous linear functional on \( \mathcal{F} \) is called a generalized function. When \( \mathcal{F} = \mathcal{D} \), the standard terminology is to call \( \mathcal{D}' \) the space of distributions \([263]\), whereas if \( \mathcal{F} = \mathcal{S} \), elements of \( \mathcal{S}' \) are called tempered distributions \([263][26]\). In other words, a generalized function is an operator \( f : \mathcal{F} \to S \) where \( S \) is the state space (usually \( \mathbb{R}, \mathbb{C} \) or \( \mathbb{R}^n \)), such that \( f \) is linear:

\[
  f(\alpha \phi + \beta \psi) = \alpha f(\phi) + \beta f(\psi),
\]

and \( f \) is continuous:

\[
  f(\phi_n) \to f(\phi) \quad \forall (\phi_n) \xrightarrow{\mathcal{F}} \phi.
\]

It is common, though occasionally misleading, to use standard function notation for generalized functions, i.e. we write \( f \equiv f(x) \) for a generalized function in the variable \( x \), knowing that this means that \( f \) is only well-defined through the test function \( \phi \equiv \phi(x) \); the action of \( f \) on \( \phi \) can also be written as

\[
  f(\phi) = (f, \phi)_{\mathcal{F}', \mathcal{F}},
\]

or simply \((f, \phi)\) if the space of test functions is clear. This notation is intended to be reminiscent of the standard \( L^2 \) inner product, but since \( f \) is a generalized function, it is better to think in terms of \( f \) being a linear functional on \( \mathcal{F} \); see \([263]\) for a more extensive discussion.

A second-order generalized random process \( f \) can now be defined in one of two ways, similar to
the two interpretations of a classical random process. Either, we treat \( f \) as a continuous operator 
\[ f : \mathcal{T} \to \mathcal{L}^2(\Omega; S), \]
that is, a mapping which carries a test function \( \phi \) into an \( S \)-valued random variable \( X_\phi \), where \( S \subseteq \mathbb{R} \) or \( \mathbb{C} \), or alternatively, we can treat \( f \) as a mapping from \((\Omega, \mathcal{F}, \mathbb{P})\) into \((\mathcal{T}', \mathcal{B})\), that is, as a generalized random vector taking values in the topological vector space \( \mathcal{T}' \). From the perspective of a measurement device, the first interpretation is perhaps the most useful. If the test function \( \phi \) represents a sensor, then \( X_\phi \) represents the random variable of observable measurement values. The space of all generalized random processes for a fixed test function space will be denoted \( \mathcal{G} \) (if the test function space is clear).

Given a collection of \( n \) test functions \( \{\phi_1, \ldots, \phi_n\} \subset \mathcal{T} \), we can define the random vector

\[ u_{\phi_1: \phi_n} = [X_{\phi_1}, \ldots, X_{\phi_n}]^\top \]  

(2.2.21)

The PDF of this random vector will be denoted \( p_{\phi_1: \phi_n}(x) \), while the CF of this random vector will be denoted \( \psi_{\phi_1: \phi_n}(\xi) \). Note that (2.2.21) can also be expressed in terms of a ‘continuous-to-discrete operator’ [26]: any collection of \( n \) test functions \( \{\phi_1, \ldots, \phi_n\} \subset \mathcal{T} \) defines a bounded operator (written for complex-valued test functions, but \( \mathbb{C} \) can be replaced by \( \mathbb{R} \) if necessary):

\[ \Phi_n : \mathcal{T}' \to \mathbb{C}^n, \quad (\Phi_n f)_m = (f, \phi_m)_{\mathcal{T}', \mathcal{T}} \]

Thus \( u_{\phi_1: \phi_n} \) is simply \( \Phi_n f \).

Note that if \( f \in \mathcal{H} \) is a classical square-integrable random process, it can also be considered a generalized random process, i.e. \( f \in \mathcal{G} \), because in this case the action of an arbitrary realization \( f \) on a test function \( \phi \) is well-defined as a standard integral:

\[ X_\phi = (f, \phi)_{\mathcal{L}^2(\mathcal{I})} = \int_{\mathcal{I}} f(\tau)\phi(\tau) \, d\tau \]

This is because the standard test function spaces can be considered as subspaces of \( \mathcal{L}^2(\mathcal{I}) \).

We now discuss the extension of the probability distribution \( \mathbb{P} \) to generalized random vectors and the possibility of defining PDFs in infinite dimensions.
2.2.7 Probability Distributions and PDFs in Infinite Dimensions

Just as in the finite-dimensional case, every generalized random vector \( f : \Omega \rightarrow \mathcal{X} \) gives rise to a probability measure on \((\mathcal{X}, \mathcal{B})\), denoted \( P_f \). As usual, the abstract definition is given in terms of the underlying probability space \((\Omega, \mathcal{F}, P)\):

\[
P_f(E) = P(f^{-1}(E)) = P(\{\omega \in \Omega : f(\omega) \in E\}) \tag{2.2.22}
\]

The measure \( P_f \) is also frequently called the pushforward of \( P \) under the map \( f \). As we have already discussed in section 2.1.2, definitions such as (2.2.22) that make explicit use of the ‘background’ probability space \((\Omega, \mathcal{F}, P)\) are purely abstract: we typically only have access to realizations of \( f \), not the underlying space \((\Omega, \mathcal{F}, P)\). We frequently make use of the notation \( f \sim P_f \) to indicate that \( f \) is a random process with distribution \( P_f \), again noting that this is simply an abstract shorthand; the properties of \( f \) are usually specified more explicitly, for instance by specifying the mean and correlation function; we give many examples in Section 2.6.

While the distribution of \( f \) always exists abstractly, the problem of constructing a probability density for a generalized random vector \( f : \Omega \rightarrow \mathcal{X} \) where \( \mathcal{X} \) is an infinite-dimensional vector space is much more subtle than in the finite-dimensional case, requiring a more rigorous treatment of absolute continuity, culminating in the statement of the Radon-Nikodym theorem. Very briefly, recall that in finite dimensions we said \( p(x) \) was a PDF of the random vector \( u \) if

\[
P_u(E) = \int_E p_u(x) \, dx \tag{2.2.23}
\]

for all events \( E \). We could write, abstractly, that

\[
p_u(x) = \frac{dP_u}{dx}, \quad \text{or} \quad dP_u = p_u(x) \, dx \tag{2.2.24}
\]

where the derivative is the so-called Radon-Nikodym derivative, and \( dx \) indicates the choice of the Lebesgue measure as the reference measure. As we discussed in Section 2.1, a PDF is guaranteed to exist if \( P_u(E) = 0 \) for all sets \( E \) such that \( \int_E dx = 0 \); we gave the example of a Dirac measure.
to illustrate why this condition is necessary. Given any two measures $P_0$ and $P_1$, we would say that $P_1$ is absolutely continuous with respect to $P_0$ if $P_0(E) = 0$ implies that $P_1(E) = 0$, and this would be written $P_1 \ll P_0$. In this case, the Radon-Nikodym theorem states that a density exists, and we would write

$$p(x) = \frac{dP_1}{dP_0} \quad (2.2.25)$$

The choice of $dx$ as the reference measure on $\mathbb{R}^n$ is merely a convenient convention, one could, for instance, choose the standard Gaussian measure $P_0 = \mathcal{N}(0, 1)$ as the reference measure.

We would like to extend (2.2.24) to the case of a generalized random vector with distribution $P_f$, but the issue is that in infinite dimensions, there does not exist an appropriate analogue of the Lebesgue measure $dx$. Thus the choice of a reference measure against which to test absolute continuity is not at all standard. One choice is to let the reference measure $P_0$ be a Gaussian measure $\mathcal{N}(\mu, \mathcal{C})$ (discussed in Section 2.6), but this choice is also not necessarily clear: a theorem of Feldman and Hájek, for instance, states that any two Gaussian measures on a normed vector space are either mutually singular (so that no PDF can exist), or equivalent (each has a density with respect to the other), and the conditions for the latter case are very strict. Thus, if working exclusively with Gaussian processes, PDFs may be useful, but will require care. Furthermore, if $P_f$ is non-Gaussian, there is in general no reason to expect that $P_f$ should be absolutely continuous with respect to any fixed Gaussian measure. So, while it is possible to define PDFs in infinite dimensions, it is not always convenient to do so.

We have several ways around this problem. First, it turns out that the characteristic functional (which we discuss in the next section) is well defined for fairly arbitrary generalized random vectors, and it provides a complete description of a random process which can be manipulated. Second, as we have discussed many times, we will almost always seek to reduce a generalized random vector $\tilde{f}$ to a finite-dimensional quantity of interest $\tilde{q} = \mathcal{M}(\tilde{f})$, for which a regular finite-dimensional probability density of the form (2.2.23) can be derived, or at least simulated by drawing approximate samples.

---

7 sigma finite, to be precise
8 rigorously, the only translation-invariant sigma finite measure on an infinite-dimensional space is the zero measure [267, 33]
of \( \tilde{f} \) and computing \( \tilde{q} \). Third, Poisson point processes present an interesting case where the Radon-Nikodym theorem is easily applied in infinite dimensions, and can be used to define a likelihood function in this setting; we discuss this in Section 2.6.2.

### 2.2.8 The Characteristic Functional

Recall again that we provided three ways to describe the complete statistics of a finite-dimensional random vector: the distribution \( \mathbb{P}_u \), the PDF \( p_u \), and the CF \( \psi_u \). The moments are a useful, but in general incomplete description of \( u \). We have already seen that a generalization of \( \mathbb{P}_u \), denoted \( \mathbb{P}_f \), can always be defined for a generalized random vector \( f : \Omega \to \mathcal{X} \), and we then explained why \( p_u \) can be inconvenient to generalize to infinite dimensions. Moments can be generalized to regular and generalized random processes, but again they suffer from the problem of incompleteness. We will now define a version of the CF \( \psi_u \) for the case of a generalized random vector \( f \); as it turns out, this provides a complete statistical description of \( f \) and proves to be easier to manipulate than other descriptions in many cases.

When we defined a second-order generalized random process \( f \in \mathcal{G} \), we stated that each test function \( \phi \in \mathcal{T} \) gives rise to a random variable \( X_\phi \in L^2(\Omega) \) via \( X_\phi(\omega) \equiv f(\phi, \omega) = (f, \phi) \), where the parenthesis are the notation for the pairing \( (\cdot, \cdot)_{\mathcal{T}', \mathcal{T}} \). Thus, we can compute the characteristic function of \( X_\phi \):

\[
\psi_{X_\phi}(\xi) = \langle \exp(-2\pi i \xi X_\phi) \rangle = \langle \exp[-2\pi i \xi (f, \phi)] \rangle \quad (2.2.26)
\]

On one hand, the formula (2.2.26) defines a function of \( \xi \in \mathbb{R} \); on the other hand, for a fixed \( \xi \in \mathbb{R} \), one can think of it as a functional from \( \mathcal{T} \) to \( \mathbb{C} \), where \( \phi \) maps to \( \psi_{X_\phi}(\xi) \). This motivates the definition of the characteristic functional as follows. Suppose that \( f \in \mathcal{G} \) is a generalized random process. Then, for any \( \phi \in \mathcal{T} \), we can define the random variable \( X_\phi \), and hence we define the characteristic functional, or CF, of \( f \) as (using the convention of [26])

\[
\Psi_f(\phi) = \langle \exp[-2\pi i X_\phi] \rangle = \langle \exp[-2\pi i (f, \phi)] \rangle_f. \quad (2.2.27)
\]
The characteristic functional was first considered by Kolmogorov in 1935 [163], and is discussed extensively in the classic book on generalized random processes by Gel’fand and Vilenkin [100].

When \( f \in \mathcal{H} \) is a classical square-integrable process, we can allow \( \phi \in L^2(I) \) in the definition of \( \Psi_f \), hence (2.2.27) returns, for every test function \( \phi \in L^2(I) \), the number \( \Psi_f[\phi] = \langle \exp[-2\pi i X_{\phi}] \rangle \).

If \( f \) has realizations in a space of generalized functions \( \mathcal{T}' \), we have that for every test function \( \phi \in \mathcal{T} \), the characteristic functional returns the number \( \Psi_f[\phi] = \langle \exp[-2\pi i X_{\phi}] \rangle \); the only difference is in the definition of \( X_{\phi} \), which in the classical case is defined as \( X_{\phi} = (f, \phi)_{L^2(I)} \), while in the generalized case is defined as \( X_{\phi} = f(\phi) \).

In (2.2.27), the test function is a function of the index parameter \( \tau \in I \). In the special case where the index set \( I \) is a spatiotemporal domain, i.e. of the form \( I = V \times T \), with \( V \subset \mathbb{R}^3 \) and \( T \subset \mathbb{R} \), we define a slightly less general version of the characteristic functional that only considers testing against a spatial function \( \phi = \phi(r) \). The result is a time-varying functional:

\[
\Psi_f[\phi, t] = \left\langle \exp \left[ -2\pi i (f(r, t), \phi(r)) \right] \right\rangle_f \tag{2.2.28}
\]

where we have written explicitly that \( f \) depends on both \( (r, t) \) while \( \phi \) depends on only \( r \). The definition (2.2.28) allows for the statistics of the process to vary in time, but will not allow the computation of joint statistics across multiple time points. So, whereas (2.2.27) can be used to compute (e.g.) the joint PDF of \( X_{(r_1, t_1)} \) and \( X_{(r_2, t_2)} \) for any \( (r_1, t_1) \in V \times T \), the functional (2.2.27) can only be used to compute the joint PDF of variables with the same time point. These PDFs can evolve in time, however, and (2.2.28) can be used to study this time evolution [206] [128].

In a certain sense, the characteristic functional arises from the finite-dimensional characteristic function of \( u_{\phi_1: \phi_n} \) by taking \( n \to \infty \); it is thus occasionally referred to as a Fourier transform in infinite dimensions. We will give many explicit examples of (2.2.27) and (2.2.28) in Section 2.6; for the remainder of this section, we discuss some general mathematical properties of the functional (2.2.27). The first property we address is to show the characteristic functional gives a complete statistical description of \( f \), thus answering the question of how to efficiently describe the statistical properties of a random process.

72
A Generalization of Bochner’s Theorem

Recall that for a finite-dimensional random vector $u$, Bochner’s theorem stated that the CF $\psi_u(\xi)$ was a complete description of the statistics of $u$, in the sense that if $\psi(\xi)$ is any continuous, positive-definite function on $\mathbb{R}^n$ with $\psi(0) = 1$, then there exists a (unique, in distribution) random vector $u$ such that $\psi(\xi) = \psi_u(\xi)$. A generalization of this theorem to CFIs was given by Minlos [200], [100], [274], and it states that if $\Psi[\phi]$ is a continuous, positive-definite functional on the space $\mathcal{S}$ with $\Psi[0] = 1$, then there exists a (unique, in distribution) generalized random process $f \in \mathcal{G}$ such that $\Psi_f = \Psi$. Thus, the CFI is a complete specification of $f \in \mathcal{G}$, and we can with impunity say that a generalized random process if drawn from $\Psi_f$ in the same way that we say $f$ is drawn from $\mathbb{P}_f$:

$$f \sim \mathbb{P}_f \iff f \sim \Psi_f$$

The left hand side is the ‘distribution’ picture while the right hand side is the ‘CFI’ picture.

Finite-Dimensional Distributions from the CFI

Suppose that $\phi_1, \ldots, \phi_n$ is a finite collection of test functions and $u_{\phi_1:\phi_n}$ is the corresponding random vector as defined in [2.2.21]. To obtain the finite dimensional PDF $p_{\phi_1:\phi_n}(x)$ for $u_{\phi_1:\phi_n}$, we evaluate the CFI at an arbitrary linear combination of $\phi_1, \ldots, \phi_n$:

$$\Psi_f \left[ \sum_{j=1}^{n} \xi_j \phi_j \right] = \left\langle \exp \left( -2\pi i \left( f, \sum_{j=1}^{n} \xi_j \phi_j \right) \right) \right\rangle$$

$$= \left\langle \exp \left( -2\pi i \sum_{j=1}^{n} \xi_j (f, \phi_j) \right) \right\rangle$$

$$= \left\langle \exp \left( -2\pi i \xi u_{\phi_1:\phi_n} \right) \right\rangle$$

$$= \psi_{\phi_1:\phi_n}(\xi)$$
Thus by evaluating $\Psi_f$ at a particular test function, the characteristic function of $u_{\phi_1:\phi_n}$ is obtained; to gain its probability density $p_{\phi_1:\phi_n}(x)$, one must evaluate the inverse Fourier transform

$$p_{\phi_1:\phi_n}(x) = \int_{\mathbb{R}^n} \psi_{\phi_1:\phi_n}(\xi) \exp(2\pi i \xi^\top x) d\xi = \int_{\mathbb{R}^n} \Psi_f \left[ \sum_{j=1}^n \xi_j \phi_j \right] \exp(2\pi i \xi^\top x) d\xi \quad (2.2.29)$$

The formula (2.2.29) exposes a possible challenge in the use of characteristic functionals to derive statistical properties of $f$: if the inverse Fourier transform (2.2.29) cannot be performed analytically, the PDF $p_{\phi_1:\phi_n}(x)$ is only defined in terms of a potentially high-dimensional integral which may be difficult to estimate (note that the dimension of the integral in (2.2.29) is equal to the number of test functions one wishes to compute the joint statistics for).

Now suppose that $f$ is equivalent to a classical process, in other words, $(f, \delta(\tau - \tau'))$ is well-defined for every $\tau'$ (it need not be). To obtain the PDF of the finite-dimensional vector of sampled values $u_{\tau_1:\tau_n}$, it is necessary that $\Psi_f[\phi]$ be well-defined for delta function inputs $\phi_j(\tau) = \delta(\tau - \tau_j)$ (again, it need not be – consider the case of Gaussian white noise given in (2.6.6)). In this case one has

$$\Psi_f \left[ \sum_{j=1}^n \xi_j \delta(\tau - \tau_j) \right] = \psi_{\tau_1:\tau_n}(\xi)$$

and hence by computing an inverse Fourier transform analogous to (2.2.29), one obtains $p_{\tau_1:\tau_n}(x)$.

**Moment Functions via Functional Derivatives of the CFl**

Like the moments of a random variable or random vector can be recovered from the characteristic function, the moment functions of a random process can be recovered from the characteristic functional by taking certain derivatives, though in the case of the characteristic functional we require functional derivatives. Given a characteristic functional $\Psi_f[\phi]$ and test function $\eta$, we consider the limit

$$\lim_{\epsilon \to 0} \frac{\Psi_f[\phi + \epsilon \eta] - \Psi_f[\phi]}{\epsilon} = \frac{d}{d\epsilon} \Psi_f[\phi + \epsilon \eta] \bigg|_{\epsilon=0} \quad (2.2.30)$$
If this derivative exists, we call it the variation of $\Psi_f$ in the direction $\eta$, and write it as

$$\tag{2.2.30} \delta \Psi[\phi; \eta]$$

If upon computing (2.2.30) we see that it takes the form $\Psi'_\phi[\eta]$ for some linear functional $\Psi'_\phi[\cdot]$ (i.e. (2.2.30) is linear in $\eta$ for each fixed $\phi \in \mathcal{F}$), then we call $\Psi'_\phi$ the functional derivative of $\Psi$ at $\phi$, written as

$$\Psi'_\phi = \frac{\delta \Psi_f[\phi]}{\delta \phi}$$

For the characteristic functional, the functional derivative is relatively easy to compute:

$$\delta \Psi_f[\phi; \eta] = \left. \frac{d}{d\epsilon} \Psi_f[\phi + \epsilon \eta] \right|_{\epsilon=0} = \left. \frac{d}{d\epsilon} \left\langle \exp \left[ -2\pi i (f, \phi + \epsilon \eta) \right] \right\rangle \right|_{\epsilon=0}$$

$$= \left\langle -2\pi i (f, \eta) \exp \left[ -2\pi i (f, \phi) \right] \right\rangle$$

$$= \left( \left\langle -2\pi i f \exp \left[ -2\pi i (f, \phi) \right] \right\rangle, \eta \right)$$

Noting that this is in the form of a linear functional, we have the (relatively) simple formula

$$\Psi'_{\phi, t} = \left\langle -2\pi i f \exp \left[ -2\pi i (f, \phi) \right] \right\rangle \tag{2.2.31}$$

Using this, it is now possible to see that if we evaluate (2.2.31) at $\phi = 0$, we obtain the following convenient formula for the mean [60] [205] [274]:

$$\bar{f} = \langle f \rangle = \frac{i}{2\pi} \Psi'_0 = \frac{i}{2\pi} \frac{\delta \Psi_f}{\delta \phi} \bigg|_{\phi=0}.$$
Higher-order moment functions can also be derived by computing higher-order functional derivatives of $\Psi_f$ but we omit their formulae here; see e.g. [205, 274].

Transformation of the CFl Under a Linear Operator

One of the key features of the characteristic functional is that it behaves very favorably under linear transformation of realizations. Suppose that $\mathcal{A} : \mathcal{X} \to \mathcal{Y}$ is a bounded linear operator with adjoint $\mathcal{A}^\dagger$. Then, if we consider the process $g(\tau) = (\mathcal{A}f)(\tau)$, it is easy to see from the definition of $\Psi_f$ that

$$\Psi_g[\phi] = \langle \exp[-2\pi i (\mathcal{A}f, \phi)] \rangle = \langle \exp[-2\pi i (f, \mathcal{A}^\dagger \phi)] \rangle = \Psi_f[\mathcal{A}^\dagger \phi]$$

As a (nontrivial) example of (2.2.32), suppose that $\mathcal{D}$ is a Fourier multiplier with nonvanishing symbol, i.e. an operator which is expressed in terms of a Fourier transform as

$$(\mathcal{D}\phi)(x) = \mathcal{F}^{-1} \left[ p(\xi)\hat{\phi}(\xi) \right] = \int_{\mathbb{R}^n} \int_{\mathbb{R}^n} p(\xi)\phi(y) \exp(2\pi i \xi^\top(x - y)) \, dy \, d\xi$$

for a symbol $p(\xi)$ which has no zeros on $\mathbb{R}^n$. The differential operator $I - \nabla^2 = 1 - \sum_{j=1}^n \frac{\partial^2}{\partial x_j^2}$ is such an example, because in this case we would have

$$p(\xi) = 1 - \sum_{j=1}^n (-2\pi i \xi_j)^2 = 1 + 4\pi^2 |\xi|^2$$

and this is a nonvanishing function. Now, suppose we wish to solve $\mathcal{D}v = w$ where $w \in \mathcal{G}$ is a generalized random process with CFl $\Psi_w$. Because $p(\xi)$ is nonvanishing, we can consider the inverse operator, which is also a Fourier multiplier, but with symbol $1/p(\xi)$:

$$(\mathcal{D}^{-1}w)(x) = \mathcal{F}^{-1} \left[ \frac{\hat{w}(\xi)}{p(\xi)} \right] = \int_{\mathbb{R}^n} \int_{\mathbb{R}^n} \frac{w(y)}{p(\xi)} \exp(2\pi i \xi^\top(x - y)) \, dy \, d\xi$$

If $p(\xi)$ is real-valued, this operator is self-adjoint, and so $(\mathcal{D}^{-1})^\dagger = \mathcal{D}^{-1}$. We would then have that
the characteristic functional of $\mathbf{v} = \mathbf{D}^{-1}\mathbf{w}$ is

$$
\Psi_{\mathbf{v}}[\phi] = \Psi_{\mathbf{w}}[\mathbf{D}^{-1}\phi]
$$

This type of example is used to construct a wide array of sparse generalized random processes by Unser et al. [274]. For example, if $\mathbf{D} = I - \nabla^2$ as above and $\mathbf{w}$ is a Gaussian white noise, then $\mathbf{v} = \mathbf{D}^{-1}\mathbf{w}$ is a mean zero isotropic Gaussian random process with exponential covariance operator

$$
k(x, x') \propto \exp(-\ell \|x - x'\|) \quad [57], \quad [274].
$$

Other differential (or non-differential) operators lead to different covariance operators, while non-Gaussian choices of noise $\mathbf{w}$ - for instance a Poisson point process - lead to interesting non-Gaussian processes $\mathbf{v}$. We discuss examples like this further in Section 2.6.

### 2.2.9 Multiple Random Processes

Suppose that we have two second-order classical random processes defined on a common probability space and index set, i.e. $\mathbf{f}^{(1)} : \mathcal{I} \times \Omega \to S_1$ and $\mathbf{f}^{(2)} : \mathcal{I} \times \Omega \to S_2$. As in Section 2.1.9, we can consider the vector-valued process $[\mathbf{f}^{(1)}, \mathbf{f}^{(2)}]^\top$ as a joint random process by considering a vector-valued process $\mathbf{f} : \mathcal{I} \times \Omega \to S_1 \times S_2$. As in 2.1.9, there are different possibilities for statistical relationships between $\mathbf{f}^{(1)}$ and $\mathbf{f}^{(2)}$, the most general case being that there is a joint distribution $\mathbb{P}_{\mathbf{f}^{(1)}\mathbf{f}^{(2)}}$ which has the distributions $\mathbb{P}_{\mathbf{f}^{(1)}}$ and $\mathbb{P}_{\mathbf{f}^{(2)}}$ as marginals. Note that in 2.1.9 we defined marginal and joint PDFs, not distributions, but because the notion of a PDF is not natural for processes, we resort to this abstract definition for now. The two extreme cases are again similar, where we say that $\mathbf{f}^{(1)}$ and $\mathbf{f}^{(2)}$ are independent if $\mathbb{P}_{\mathbf{f}^{(1)}\mathbf{f}^{(2)}} = \mathbb{P}_{\mathbf{f}^{(1)}}\mathbb{P}_{\mathbf{f}^{(2)}}$, and deterministically coupled if there exists an operator $\mathcal{M} : \mathcal{X} \to \mathcal{X}$ such that $\mathbf{f}^{(2)} \overset{d}{=} \mathcal{M}(\mathbf{f}^{(1)})$, where the superscript $d$ means ‘in distribution’, i.e. $\mathbb{P}_{\mathbf{f}^{(2)}} = \mathbb{P}_{\mathcal{M}(\mathbf{f}^{(1)})}$.

Because the characteristic functional is slightly more natural than the distribution (being a functional instead of a measure on a function space), we can also define joint and marginal processes using an upgraded version of $\Psi_{\mathbf{f}}$. So, suppose that $\mathbf{f} = [\mathbf{f}^{(1)}, \ldots, \mathbf{f}^{(n)}]^\top$ is a vector-valued generalized random process, i.e. the realizations of $\mathbf{f}$ are elements of $\mathcal{T}'$ where $\mathcal{T}$ is a space of
vector-valued test functions. Then, for any \( \phi = [\phi^{(1)}, \ldots, \phi^{(n)}] \top \in \mathcal{S} \), we define

\[
(f, \phi) = \sum_{j=1}^{n} (f^{(j)}, \phi^{(j)})
\]

where the parentheses are interpreted either as \( L^2(\mathcal{I}) \) inner products or as generalized function pairings. We then define the joint characteristic functional as

\[
\Psi_f[\phi] = \left\langle \exp \left[ -2\pi i (f, \phi) \right] \right\rangle = \left\langle \exp \left[ -2\pi i \sum_{j=1}^{n} (f^{(j)}, \phi^{(j)}) \right] \right\rangle = \left\langle \prod_{j=1}^{n} \exp \left[ -2\pi i (f^{(j)}, \phi^{(j)}) \right] \right\rangle
\]

We can then say that if \( f^{(1)} \) and \( f^{(2)} \) are independent if

\[
\Psi_f[\phi] = \Psi_{f^{(1)}}[\phi^{(1)}] \Psi_{f^{(2)}}[\phi^{(2)}]
\]

If \( f \) is a second-order random process, each fixed \( \tau \in \mathcal{I} \) gives rise to a random vector \( f(\tau, \cdot) \equiv \mathbf{u}_\tau \in L^2(\Omega; \mathbb{R}^m) \) for all \( t \in T \), and so the following two functions exist:

\[
\mu_f(\tau) = \left\langle g_\tau \right\rangle
\]

\[
k_f(\tau, \tau') = \text{Cov}(g_\tau, g_{\tau'}) = \left\langle (g_\tau - \mu_g(\tau))(g_{\tau'} - \mu_g(\tau')) \right\rangle
\]

Note that the mean function \( \mu_f \) is a (non-random) vector-valued function, while the covariance function is a (non-random) matrix-valued function. Writing them in components would look like:

\[
\mu_f(\tau) = \begin{bmatrix}
\mu_f^{(1)}(\tau) \\
\vdots \\
\mu_f^{(m)}(\tau)
\end{bmatrix} \in \mathbb{R}^m
\]

\[
k_f(\tau, \tau') = \begin{bmatrix}
k_f^{(1,1)}(\tau, \tau') & \cdots & k_f^{(1,m)}(\tau, \tau') \\
\vdots & \ddots & \vdots \\
k_f^{(m,1)}(\tau, \tau') & \cdots & k_f^{(m,m)}(\tau, \tau')
\end{bmatrix} \in \mathbb{R}^{m \times m}.
\]
where each $k^{(i,j)}_{\tau, \tau'}$ is a cross-covariance function of the $i$th component against the $j$th component, i.e.

$$k^{(i,j)}_{\tau, \tau'} = \left\langle (f^i(\tau) - \mu^i(\tau))(f^j(\tau') - \mu^j(\tau'))^* \right\rangle$$

More explicitly,

$$k_{\tau, \tau'} = \begin{pmatrix}
\left\langle (f^1(\tau) - \mu^1(\tau))(f^1(\tau') - \mu^1(\tau'))^* \right\rangle & \cdots & \left\langle (f^1(\tau) - \mu^1(\tau))(f^n(\tau') - \mu^n(\tau'))^* \right\rangle \\
\vdots & \ddots & \vdots \\
\left\langle (f^n(\tau) - \mu^n(\tau))(f^1(\tau') - \mu^1(\tau'))^* \right\rangle & \cdots & \left\langle (f^n(\tau) - \mu^n(\tau))(f^n(\tau') - \mu^n(\tau'))^* \right\rangle
\end{pmatrix}$$

### 2.3 Object-Oriented Design and Parallel Computing

In the following sections, we describe several methods for simulating random variables, vectors, and processes. The goal of any computational scheme is to choose a system of representation for the abstract mathematical object under consideration, then design algorithms that work with this representation to produce a desired output which approximates some underlying operation. In this work, there are several mathematical objects that we wish to simulate: physiological random processes, represented by some generalized random vector $f$; imaging data, represented by some generalized random vector $g|f$, whose distribution depends on $f$ through a system operator $H$; virtual patient-specific ensembles represented by a generalized random vector $\tilde{f}$, and quantity of interest models $\tilde{q} = M(\tilde{f})$. Each of these entities must be represented in the computer in order to estimate the desired probabilities and produce usable output.

In the hopes of making this dichotomy between representation of objects in hardware and software and the mathematical objects themselves more clear, we shift gear slightly and introduce some basic concepts from object oriented design. The goal is not to discuss this subject at length; we recommend the books [265] and [194]. We also do not present code in any particular language, but rather use a pseudocode that mimics both C++ and Matlab, without trying to maintain any syntactic correctness for either language. We maintain the practice of specifying the data type of
inputs and outputs as is the practice in C++, because we feel this is more clearly in line with the mathematical practice of specifying the domain and co-domain of a function. The code written for this dissertation used a combination of C++, CUDA and Matlab.

The basic concepts of OOP that we feel are important to emphasize are the following.

1. An object – called a class in OOP – separates data from methods.

2. Objects are composable and have inheritable properties.

To illustrate the first concept, consider the representation of a Fourier mode \( f(r) = \exp(2\pi i k \cdot r) \), where \( r \in \mathbb{R}^3 \). The data required to represent this object is the wavevector \( k \in \mathbb{R}^3 \); the method required to evaluate such a function is the numerical algorithm which computes the output \( f(r) \) for an arbitrary input \( r \). Thus, we can imagine a class named FourierMode, such as in Listing 2.1.

```cpp
class FourierMode {
    data:
        Vector3D k;
    methods:
        ComplexNum Eval(Vector3D r) {
            return \exp(2\pi i \cdot \text{dot}(k, r));
        }
};
```

Listing 2.1: Example of a simple Fourier mode object.

We are making several assumptions in the definition of FourierMode in Listing 2.1. First, we are already making use of one of the properties discussed: compositability. We are assuming that elsewhere, we already have defined classes for two different objects, a Vector3D object and a ComplexNum object, and we are making use of these objects here. Second, we are assuming the existence of a function \( \exp \) which accepts an object of type ComplexNum and returns another, and the existence of a function \( \text{dot} \) which accepts two Vector3Ds and returns a ComplexNum. We are also assuming that methods exist to create, modify and destroy an object of FourierMode type; for instance, to create an object we might simply call \( f = \text{FourierMode}(k) \) where \( k \) is some previously defined Vector3D. The object has now been ‘instantiated’, and we can make a call such as \( f.\text{Eval}([1,1,1]) \) to evaluate the function for the input vector \( r = [1,1,1] \) (the period notation...
is the standard way to call an object’s internal method).

We will use constructs similar to Listing 2.1 throughout the remainder of this work to demonstrate how complex mathematical objects are represented in the computer.

Because parallel computing also plays a significant role in the ability to simulate imaging systems and perform other computational tasks quickly, we also briefly discuss some abstract concepts from parallel computing. Again, we will be reasonably agnostic about the particular language or hardware, instead focusing on the following two components of any parallel program:

1. A kernel, which is a method that can be applied to some data, usually concurrently with other instances of the same kernel (but acting on different data), or concurrently with different kernels, each running within a thread.

2. A launch which is a ‘master’ program that coordinates data transfer, organizes and instantiates kernel threads, and communicates with the user when the method is complete.

We again use the example of the function \( f(r) = \exp(2\pi i k \cdot r) \) to illustrate the notion of launching a group of threads to perform a data-parallel task. Suppose instead of evaluating \( f(r) \) for a single \( r \), we wish to compute \( f(r) \) for a vector of \( r \)s, say \( n \) evaluation points \( r_1, \ldots, r_n \). Clearly, this operation can be performed ‘trivially’ in parallel, since if you had \( n \) friends with calculators, you could tell each one to compute some \( f(r_j) \) and obtain the result \( n \) times faster than if you only had one friend computing the \( n \) values, one after the other (neglecting communication time). To illustrate this concept, we introduce a pseudocode launch syntax which mimics the CUDA language as follows:

\[
\text{GridEval}<<<n>>>(\text{ComplexNum* Output, Vector3D* Input, Vector3D k, int n});
\]

In the above call, we are asking for \( n \) threads (the number inside the ‘triple chevrons’) to compute the result on each of the \( n \) input variables, \( r_1, \ldots, r_n \), which are stored in the array of Vector3Ds called Input, then store the result in the array Output. Note that we are using the standard C pointer notation, where the * indicates that Output is an array of (i.e. pointer-to) ComplexNums. Abstractly, we are assuming that this operation occurs in parallel, as illustrated in
Figure 2.4: Parallel flowchart for the GridEval function, which is an example of a trivially parallelizable algorithm. The user supplies a collection of evaluation points, $r_1, \ldots, r_n$; the GridEval function launches a grid of threads, one for each evaluation point; these threads perform independently to produce the output vector of function evaluations.

Obviously not all algorithms are as trivially parallelizable as GridEval; in many cases, an algorithm will require accessing and contributing to many components of an array, and thus multiple threads may need to read or write the same regions of memory; these race conditions can make the analysis of parallel algorithms difficult. Many of the parallel designs encountered in this work conveniently happen to be of the form displayed in Figure 2.4.

### 2.4 Simulating Random Variables and Vectors

Recall the main goal of $\mathcal{M}$-PMED is to be able to compute model outputs of the form

$$\tilde{q} = \mathcal{M}(\tilde{f})$$
where $\tilde{f}$ is a virtual patient ensemble, $\tilde{q}$ is a quantity of interest, and $\mathcal{M}$ is a model that estimates $\tilde{q}$. We wish to compute probabilities or other statistical properties of $\tilde{q}$, for example

$$
P_{\tilde{q}}(E) = P(\tilde{q} \in E)
$$

where $E$ is some desired set of outcomes. In order to compute such probabilities, we must first be able to simulate random $\tilde{f}$.

We begin by discussing some general strategies for probabilistic computation, including methods for generating random variables and random vectors. Then, we show how these methods can be used to generate a wide variety of random process models for use in the simulation and evaluation of imaging systems.

Probabilistic computation consists of taking a computational object $f$ (for instance, one intended to simulate the process $f$) and making some of its data elements randomizable. In other words, to make $f$ random, we will simply introduce a new new method called $f$.Randomize which will draw new data elements (i.e. $f$.x,f.y,...) from a specified distribution. This new method $f$.randomize may require new data elements that specify the probability distribution from which new data elements will be drawn (for instance, the mean and covariance matrix or an entire object that otherwise specifies the distribution). Then, the overall goal of probabilistic computation is usually to repeatedly call $f$.Randomize, followed by some other method (for instance, one which computes a function value), then compute statistics (i.e. approximate expected values) with the resulting sampled values by using the standard Monte Carlo approximation.

This is precisely how treatment efficacy figures of merit will be computed in Chapter 4. Suppose, for example, an object $M = TreatmentSimulator$ exists which is capable of simulating the effect of a treatment on a patient; say $q = M$.ComputeEffect is a method that produces a real number $q$ summarizing the effect of the treatment. The simulator $M$ depends on some data elements $M$.x, $M$.y, etc., representing for instance the patient’s anatomical, physiological and genetic makeup. Because these elements may be partially or wholly unknown, we may choose to randomize them by selecting (reasonable) probability distributions for $M$.x, $M$.y etc. Then, we can generate for instance an empirical PDF for $q = M$.ComputeEffect by repeatedly calling $M$.Randomize followed by
M.ComputeEffect to generate new samples of $q$, after which we may estimate statistical quantities of interest such as $\langle q \rangle$ or $P(q \in E)$. An example script for such a computation is shown in Listing 2.2.

```matlab
f = PatientModel (); % An object modeling the patient
M = TreatmentSimulator (); % An object modeling the treatment
Q = zeros (nsim,1);
for i = 1:nsim
    f.Randomize (); % Randomize the patient
    Q(i) = M.Evaluate (f); % Simulate treatment on new patient
end
qbar = mean (Q); % For instance
sigq = std (Q);
```

Listing 2.2: An example probabilistic computation script performing a virtual clinical trial. Here, the patient is modeled by the computational object $f$, while the treatment is modeled by the computational object $M$. A new virtual patient is generated by randomizing $f$, then the treatment is tested by applying the model $M$ to the new $f$.

### 2.4.1 Random Variable Generation

Essentially all modern probabilistic simulations are constructed from an elementary object: the uniform pseudorandom number generator. An algorithm which produces pseudorandom numbers is one which strives to produce a stream of numbers $\xi_1, \xi_2, \ldots$ such that, by as many metrics as possible, the distribution of the sequence approximates the uniform distribution $U[0,1]$ as closely as possible. We will denote a generic pseudorandom number generator by the function name `rand()` (which is the standard function name in ISO C, Matlab and many other programming languages). It is understood that a call to `rand()` produces a single double precision number $\xi \in [0,1]$. To produce an $n_1 \times n_2 \times \cdots n_m$ array of random numbers using the pseudorandom stream, we call `rand(n_1,n_2,\ldots,n_m)`; for example `rand(4,1)` produces a $4 \times 1$ vector of pseudorandom numbers. For a discussion of different implementations of `rand()` see Knuth [161] or Robert and Casella [236].

Because all standard (i.e. non-quantum) modern computers are built on deterministic principles (effectively, a standard computer is isomorphic to several billion light switches), in reality a pseudorandom number generator is not random at all: if provided the same seed number ($\xi_0$) every
time, the same sequence will be generated, since a pseudorandom generator will simply apply a deterministic function $\xi_j = R(\xi_{j-1})$ to generate the sequence. For example, every time Matlab is opened on my computer, unless the random number generator is re-seeded, the following sequence of double precision numbers is always produced upon repeated calls to `rand()`:

$$0.814723686393179, 0.905791937075619, 0.126986816293506, 0.913375856139019, \ldots$$

If Matlab is closed and re-opened, the same sequence appears, because the generator is using the same initial value $\xi_0$ to generate the sequence. To avoid this, one first calls the Matlab function `rng('shuffle')`, which re-seeds the sequence using the current computer clock time; other languages employ a similar technique to ‘shuffle’ the sequence. Thus calling `rng('shuffle')` followed by `rand(4,1)` results in the following sequence:

$$0.536080681493685, 0.199879725442388, 0.954699923235847, 0.399923693633554.$$  

If we call `rng('shuffle')` again, a different sequence will be deployed, and the chance of a repetition is extremely small.

Given a pseudorandom number generator `rand()`, we now wish to generate random numbers with distributions on $\mathbb{R}$ different from $U[0,1]$. For instance, suppose $X$ has PDF $p_X(x)$ and CDF $F_X(x)$; we want to use `rand()` to simulate samples from $X$. The best-case scenario is that some simple function $T : [0,1] \to \mathbb{R}$ exists so that $X = T(\xi)$ (in distribution). Then, we can simulate a sample from $X$ by first letting $x_i = \text{rand}()$ and then letting $X = T(x_i)$. Such a map is called a change of variables, or deterministic coupling, and the following theorem provides exactly such a method:

**Theorem 1** ([236]). Let $X$ be a real random variable with distribution function $F_X(x)$. Define the generalized inverse of $F_X$ via

$$F_X^{-1}(t) = \min_x \{ x : F_X(x) \geq t \}, \quad t \in (0,1], \quad F_X^{-1}(0) = -\infty$$

85
Note that because $F_X$ is right-continuous, the minimum exists and $F_X(F_X^{-1}(t)) = t$. Then, if $\xi \sim U_{[0,1]}$ is uniformly distributed on $[0,1]$, the random variable

$$Y = F_X^{-1}(\xi)$$

has distribution $F_X(x)$.

Proof. Recall that $F_X(x) = \mathbb{P}((-\infty, x]) = \mathbb{P}(X \leq x)$; thus

$$F_Y(x) = \mathbb{P}(Y \leq x) = \mathbb{P}(F_X^{-1}(\xi) \leq x) = \mathbb{P}(\xi \leq F_X(x)) = F_\xi(F_X(x)) = F_X(x)$$

the last equality follows since $F_\xi(t) = t$ for all $0 \leq t \leq 1$.

The only issue with implementing (2.4.1) is that an explicit formula for $F_X^{-1}$ might not be readily available. We will now introduce several general methods for simulating random vectors with arbitrary PDF $p_u(x)$. As a special case, these methods also provide alternatives to Theorem 1 for simulating random variables.

### 2.4.2 Simulating Random Vectors: General Issues

While we have just seen that the simulation of scalar random variables $X$ with prescribed distribution is very straightforward if a formula for $F_X^{-1}(t)$ and a standard uniform pseudorandom number generator `rand()` are available, the generation of an $n$-dimensional random vector $u \in \mathbb{R}^n$ with prescribed distribution $p_u(x)$ is less trivial. This is because the components of $u$ can demonstrate reasonably complex correlation structures: one cannot simply draw an $n$-dimensional uniform pseudorandom vector $\xi$ and expect to apply a simple transformation rule $u = F(\xi)$. In principle, such a function $F : \mathbb{R}^n \to \mathbb{R}^n$ exists, but it almost certainly does not have a simple form like (2.4.1). Without going into the details, one method for constructing a map $y = F(\xi)$ such that $p_y(x) = p_u(x)$ for some specified density $p_u(x)$ is via optimal transport \[280\]. However, optimal transport remains computationally very challenging in high dimensions, and since we will be concerned with sampling random vectors of dimension ranging from the hundreds to the millions, this method, while interesting, is not yet practical for our purposes.

86
There are some random vectors that are relatively straightforward to sample, requiring for instance only repeated evaluation of the PDF or a matrix factorization and a matrix-vector multiplication in addition to the drawing of $n$ independent pseudorandom numbers, and we will discuss these.

The simplest random vectors to simulate are finite-dimensional, independent and identically distributed vectors. If $u \in \mathbb{R}^n$ is a random vector with independent components, then, because there are no statistical relationships between different components of $u$, to generate samples from $u$ it suffices to generate random scalars $U_1, \ldots, U_n$, independently, using any standard method. In particular, this can be done trivially in parallel, with each thread computing one sample from $\mathbb{P}_U$. For example, the CUDA language implements a set of random number generators called the cuRAND library [215] [43].

2.4.3 Simulating Gaussian Random Vectors

One of the workhorses of stochastic simulation is the multivariate normal random vector $u \sim N(\mu, C)$, where $\mu \in \mathbb{R}^n$ is an arbitrary vector and $C$ is a positive-semi-definite symmetric real matrix. There are several standard methods that make use of the standard $\text{rand()}$ to draw samples from $N(\mu, C)$; we discuss mainly factorization methods based on the Cholesky and spectral decompositions.

Factorization methods rely on the fact that a normal random vector can be transformed to an independent random vector $w$ by means of a linear transformation. In other words, there exists an invertible matrix $A$ such that $w = Au$ has distribution $N(0, I_n)$; thus $u = A^{-1}w$, and so samples from $N(0, I_n)$ can be transformed into samples of $w$ by application of a matrix multiplication (assuming $A^{-1}$ is known) or by solving the system $Au = w$ (if $A^{-1}$ is not known explicitly). Because it transforms $u$ to a white noise, the matrix $A$ is usually called a whitening matrix in the signal processing and detection literature [26] [279] [155].

There are many equivalent linear transformations that are useful for constructing $N(\mu, C)$; we briefly discuss the Cholesky and spectral decomposition methods. Assume that $C \in \mathbb{R}^{n \times n}$ is a real, symmetric, positive definite matrix. Recall that positive definite means that $x^\top C x > 0$ for
all \( x \in \mathbb{R}^n \). Then, recall [273] that the Cholesky factorization of the symmetric positive definite matrix \( C \) is the unique factorization of the form

\[
C = R^\top R
\]

where \( R \) is upper-triangular and, as usual, \( \top \) denotes the transpose. The spectral factorization of \( C \) consists of an orthogonal matrix \( U \) and a diagonal matrix \( \Lambda \) such that

\[
C = U\Lambda U^\top
\]

Now, let \( w \sim N(0, I_n) \) be an \( n \)-dimensional Gaussian white noise vector i.e. each component is I.I.D. with distribution \( N(0,1) \), and let \( \mu \in \mathbb{R}^n \) be an arbitrary mean vector. Then, either of the following transformations results in \( u \sim N(\mu, C) \):

\[
u = \mu + R^\top w
\]
\[
u = \mu + \Lambda^{1/2}Uw
\]

To prove this, recall that if \( y = Ax + b \) is an affine transformation of a normal random vector \( x \sim N(\bar{x}, C_x) \), then \( \tilde{y} = A\bar{x} + b \), and \( C_y = AC_xA^\top \); thus with \( A = R^\top \), we have \( C_x = R^\top I_n R = R^\top R = C \). With \( A = \Lambda^{1/2}U \) we have \( C_x = \Lambda^{1/2}U(\Lambda^{1/2}U)^\top = \Lambda UU^\top = U\Lambda U^\top = C \) where we have used the fact that diagonal matrices commute with any matrix.

Computing either the Cholesky factorization or the spectral decomposition is a nontrivial computational task: both require effectively \( O(n^3) \) operations for a matrix of size \( n \times n \) – see e.g. [273] and [108]. If the vector \( u \) arose from the discretization of a three-dimensional random process on a grid of size \( m^3 \), say, then the factorization would require \( O(m^9) \) operations. The advantage of the matrix factorization method is that the factorization only needs to be computed once and stored; then, because generating I.I.D. vectors is trivially parallelizable, simulating many I.I.D. realizations of \( u \sim N(\mu, C) \) is trivially parallelizable: each sample requires \( n \) I.I.D. samples from \( N(0,1) \) and a matrix multiplication, but these operations can be performed in parallel (one thread per sample).
Accelerations are possible if $C$ has special structure, for instance if $C$ is circulant, the spectral decomposition of $C$ can be computed using the Fast Fourier Transform (FFT); if $C$ is not circulant but is Toeplitz, it can be embedded into a larger circulant matrix, after which the FFT can be applied. Such methods are discussed at length in [181]. Chow and Saad [56] also discuss a general preconditioning technique that reduces the cost of sampling from $N(\mu, C)$.

### 2.4.4 Accept-Reject Methods

Suppose that $u \sim p_u(x)$. Is there a general method to draw samples of $u$ that does not require knowing a transformation $u = F(w)$? The answer is effectively, yes, but efficiency starts to become an issue as the dimension and complexity of $u$ grow. The simplest general method is the accept-reject method, which is based on what Robert and Casella [236] call the fundamental theorem of simulation:

**Theorem 2** (Fundamental Theorem of Simulation, [236]). *Sampling the $n$-dimensional random vector $u \sim p_u(x)$ is equivalent to sampling $(v, W) \sim U\left(\{(x, w) \in \mathbb{R}^{n+1} : 0 < w < p(x)\}\right)$, where $U(\cdot)$ is the uniform measure.*

In other words, we can draw uniform samples of pairs $(x, w)$ and simply reject those $x$ that do not satisfy $0 < w < p(x)$; the resulting $x$s are then samples from $p_u(x)$. The naïve way to do this is to sample $(x, w)$ uniformly from an $n + 1$-dimensional rectangle $R \times [0, d]$ where $R = \prod_{i=1}^{n}[a_i, b_i]$ and $d \geq \max_{x \in R} p(x)$, then remove all samples that have $w \geq p(x)$. This produces a picture like the one in Figure 2.5a.

A better method is to construct an auxiliary density $q(x)$ and a constant $M$ such that $p(x) \leq Mq(x)$ for all $x$. Then, assuming we can sample easily from $q(x)$ (for instance, if $q(x)$ is Gaussian), we can first draw $x \sim q$, then draw $u$ uniformly from $[0, Mq(x)]$. This results in a picture like in Figure 2.5b. The main issue with accept-reject methods is that the so-called acceptance rate $\alpha$, defined as the percentage of samples $(x, w)$ that end up being used in each iteration, scales very poorly with the dimension $n$; see [236] and Figure 2.6.
(a) In the naïve accept-reject scheme, original pairs \((x, u)\) are drawn from \([-6, 10] \times [0, 0.3]\) (blue dots), and only those satisfying \(0 < u < p(x)\) were accepted (black dots). The final sample values are the \(x\) values (black x’s on x-axis). The original density (red line) and a histogram estimate are shown.

(b) In the improved accept-reject scheme, original pairs \((x, u)\) are drawn uniformly from the region \(\{(x, u) : 0 < u < q(x)\}\) (blue dots), i.e. we only consider points under a scaled ‘envelope’ density \(q(x)\). Then, samples satisfying \(0 < u < p(x)\) are accepted (black dots). The final sample values are the \(x\) values (black x’s on x-axis). The original density (red line) and a histogram estimate are shown.

Figure 2.5: Comparison of naïve and improved accept-reject schemes to sample from 
\(p(x) \propto |\sin(x)| \exp(-x^2/2) + \exp(-(x-3)^2/2)\) \[236\].

2.4.5 Maximum Entropy, Gibbs Distributions and Markov Chain Sampling

Let \(u \in \mathbb{R}^n\) be a random vector. We say that \(u\) has a Gibbs distribution with energy \(\mathcal{E}(x)\) if the PDF of \(u\) has the form

\[
p_u(x) = \frac{1}{Z(\beta)} \exp(-\beta \mathcal{E}(x)) \tag{2.4.2}
\]

Probability distributions of the form (2.4.2) arise in a wide array of contexts, ranging from statistical mechanics \[145\] to image processing \[298\]. The number \(Z(\beta)\) is usually called the partition function and is simply the integral of \(\exp(-\beta \mathcal{E}(x))\) over the domain of \(x\), while \(\beta\) is the (inverse) temperature. A distribution of the type (2.4.2) is perhaps not as specialized as it looks: any PDF \(p_u(x)\) which is everywhere positive can be written in the form (2.4.2) by simply setting \(\mathcal{E}(x) = -\ln p_u(x)\) and \(\beta = Z(\beta) = 1\).
Figure 2.6: A demonstration of the poor scaling of the naïve accept-reject sampling scheme as the dimension $n$ increases. The accept-reject method is used to draw samples from a Gaussian $p(x) \propto \exp\left(-\frac{1}{2}x^T x\right)$ for $x \in \mathbb{R}^n$ with $n$ ranging from 1 to 100 and $q(x) \propto 1$. The acceptance ratio $\alpha$ is displayed on a log scale; for $n = 100$, only 0.25% of samples are accepted. Parallel computing can alleviate this problem somewhat, so long as individual samples are fast to compute; we simply employ more threads.

Of interest to us is the case where (2.4.2) is constructed via the principle of maximum entropy; see e.g. [26] [65] [146] [145] [306] [307]. Let $u$ be a random vector in $\mathbb{R}^n$ with absolutely continuous probability density $p(x) \in \mathcal{P}_{ac}(\mathbb{R}^n)$, where $\mathcal{P}_{ac}$ is the set of absolutely continuous probability measures. We define the entropy of $u$ by the functional

$$H(u) \equiv H(p_u) = -\int_{\mathbb{R}^n} p_u(x) \ln p_u(x) \, dx$$

(2.4.3)

If $\mathbb{P}_u$ is not absolutely continuous with respect to Lebesgue measure, convention dictates that the entropy of $u$ should be negative infinity, though in the case where $u$ is purely discrete, the integral in (2.4.3) would typically be changed to a sum (taking the choice of the counting measure as the reference measure to define the integral) [65].

Now consider the possibility of maximizing (2.4.3) over some class of probability densities, denoted by say $\mathcal{P}_0 \subset \mathcal{P}_{ac}$. We say that $p^*$ is a maximum entropy distribution for the class $\mathcal{P}_0$ if

$$H(p^*) \geq H(p) \quad \forall p \in \mathcal{P}_0$$

91
In other words, to find $p^*$, we must solve the following variational problem:

$$p^* = \arg \max_{p \in \mathcal{P}_0} H(p) \quad (2.4.4)$$

Note that even though we are considering finite-dimensional probability distributions, we must solve an infinite-dimensional optimization problem in order to find $p^*$. Thankfully, (2.4.4) has a particularly elegant solution in the following case. Suppose that the class $\mathcal{P}_0$ is formed by the following set of constraints:

$$\mathcal{P}_0 = \{ p \in \mathcal{P}_{ac} : \langle \psi_j(x) \rangle = \mu_j, 1 \leq j \leq n \} \quad (2.4.5)$$

where

$$\langle \psi_j(x) \rangle = \int_{\mathbb{R}^n} \psi_j(x)p(x) \, dx$$

In other words, the situation we are in is that we don’t know the full density $p_u(x)$ of some random variable $u$, but, we assume that we do ‘know’ (by some means) some expected values of some functionals of $u$; these expected values take the form $\langle \psi_j(x) \rangle = \mu_j$. As a concrete example, imagine that $n = 1$, $\psi_1(x) = x$ and $\psi_2(x) = x^2$; then, $\langle \psi_1(x) \rangle = \langle x \rangle = \mu_1$ is the mean of the random variable, while $\langle \psi_2(x) \rangle = \langle x^2 \rangle = \mu_2$ is the correlation of $x$. Assuming that $\mathcal{P}_0$ is given by (2.4.5), an application of the (infinite-dimensional) method of Lagrange multipliers [26, 304] shows that, if the solution to (2.4.4) exists, then there exist numbers $\lambda_0, \ldots, \lambda_n$ such that

$$p^*(x) = \exp \left( -\sum_{j=0}^{n} \lambda_j \psi_j(x) \right)$$

where $\psi_0 \equiv 1$ represents the implicit normalization condition that $\int_{\mathbb{R}^n} p^*(x) \, dx = 1$. In other words, the solution to the variational problem (2.4.4) is exactly a Gibbs distribution of the form
with energy given by
\[ E(x) = \sum_{j=1}^{n} \lambda_j \psi_j(x) \]
and \( Z^{-1} = \exp(-\lambda_0) \). The Lagrange multipliers can be computed explicitly, if the partition function is available; in practice this may be challenging. More frequently, the Lagrange multipliers are found by maximum likelihood estimation.

To draw samples from a density of the form \((2.4.2)\), we can employ for instance the *Gibbs sampler* or *Hamiltonian Monte Carlo*. Both are examples of Markov Chain (MC) sampling methods, which are a general class of samplers that construct a sequence \( u_1, u_2, \ldots \) which are not independent, but rather dependent in a very particular way: the sequence of samples forms a Markov chain. Space unfortunately does not permit a discussion of MC samplers; see e.g. the books \([236] \ [178] \ [298] \ [150]\).

### 2.5 Simulation of Random Processes

We'll briefly discuss some general strategies for simulating random processes \( f \). There are two possibilities that we consider in this work:

1. The process \( f \in \mathcal{H} \) so that each realization is an ordinary \( L^2(I) \) function.

2. The process \( f \in \mathcal{G} \) is a point process with \( \bar{N} < \infty \). In this case, realizations of the process are fully described by the number \( n \) and the sample points \( a_1, \ldots, a_n \), where each \( a_j \in \mathbb{R}^k \).

We can only possibly represent the realizations of a random process \( f \) in a computer using finitely many numbers. The second case, despite having realizations which are generalized functions, is actually easier to deal with than the first case: the realizations themselves are described by finitely many numbers, so a complete description of \( f(\cdot, \omega) \) is possible in the computer without any approximation (up to finite precision arithmetic). We thus imagine a class \texttt{PointProcess} with data members \( n \) and \( A \), where \( n \) is an integer and \( A \) is a vector or matrix data structure containing \( a_1, \ldots, a_n \). We can then imagine an evaluation routine \texttt{PointProcess.Eval(phi)} whereby a
function $\phi \in \mathcal{T}$, represented by the object $\text{phi}$, returns the number

$$f(\phi) = (f, \phi)_{\mathcal{T}', \mathcal{T}} = \sum_{j=1}^{n} (\delta(a - a_j), \phi)_{\mathcal{T}', \mathcal{T}} = \sum_{j=1}^{n} \phi(a_j)$$

Thus we require only that the object $\text{phi}$ possess an evaluation function, for instance a C++ parenthesis operator with the signature `double operator()(Vector a);`

For processes $f \in \mathcal{H}$, the picture is slightly more complicated: realizations may in general require infinitely many parameters to fully describe (e.g. an infinite sequence of Fourier or KLE expansion coefficients). Thus we must approximate $f$ by choosing an appropriate finite-dimensional random vector $z$, whose realizations can be ‘synthesized’ to give an approximation of $f$. By synthesize, we mean that a map

$$\tilde{\Phi}^\dagger_m : \mathbb{R}^m \to L^2(\mathcal{I})$$

exists which can be used to convert the finite-dimensional representation into a function description.

For example, as we discussed in Section 2.2.5, we can let $\{\phi_j\}_{j=1}^{\infty}$ be any Riesz basis or frame for $L^2(\mathcal{I})$ (for example, an orthonormal basis) [58]. Then, if $f \in \mathcal{H}$, we have that each realization can be written as

$$f(\tau, \omega) = \tilde{\Phi}^\dagger \Phi f = \sum_{j=1}^{\infty} (f(\omega), \phi_j) \tilde{\phi}_j = \sum_{j=1}^{\infty} Z_j(\omega) \tilde{\phi}_j(\tau)$$

We can then choose to approximate $f(\tau, \omega)$ by its projection onto a finite-dimensional subspace $V_m \subset L^2(\mathcal{I})$, spanned by the first $m$ synthesis functions i.e. $\{\tilde{\phi}_j\}_{j=1}^{m}$. Given a realization of the random vector $z$ (say $z = z(\omega) = [Z_1(\omega), \ldots, Z_m(\omega)]^T$), we can construct an approximation to $f(\cdot, \omega)$ by setting

$$f(\tau, \omega) \approx \sum_{j=1}^{m} Z_j(\omega) \tilde{\phi}_j(\tau) = \tilde{\Phi}^\dagger_m z(\omega), \quad (2.5.1)$$

where $z(\omega)$ is the (random) vector of expansion coefficients and $\tilde{\Phi}^\dagger_m$ is the synthesis operator for
the finite basis $\{\tilde{\phi}_j\}_{j=1}^m$. Note that the synthesis functions $\tilde{\phi}_j$ need not be equal to the analysis functions $\phi_j$. The expansion (3.7.9) is very similar to the approximate Karhunen-Loève expansion already discussed (see equation (2.2.19)), though it is not necessarily the case that $\{\phi_j\}$ are the eigenfunctions of $C_f$ (in fact, we may not know $C_f$, or may not be able to or want to solve the eigenvalue problems). In general, unless the $\phi_j$ are KL basis functions, the components of the random vector $z$ will not be independent or even uncorrelated.

An approximation of the type (3.7.9) will be called a Galerkin-type approximation, because it approximates a function by an element of a finite-dimensional subspace of $L^2(\mathcal{I})$ spanned by a known set of basis functions. The name Galerkin comes from the finite-element literature [181]. The approximation (3.7.9) is also sometimes only weak in the sense that we may only have $\tilde{\Phi}_m^\dagger z \to f$ as $m \to \infty$ in the $\mathcal{H}$ topology, i.e. not pointwise in either $\omega$ or $\tau$. Stronger approximation properties can be shown if the process is known to have smooth realizations [181].

One of the most common examples of (3.7.9) in imaging is the classical voxel representation [26]. In a voxel representation, the functions $\tilde{\phi}_j$ are characteristic functions of a collection of cubes $Q_j \subset \mathbb{R}^n$ that form a regular grid. It is more typical then to use a multiindex, say $\tilde{\phi}_{ij}$ or $\tilde{\phi}_{ijk}$, where each index represents an offset in some direction. Voxel representations, while standard, are somewhat pathological for several reasons, as discussed extensively in [26]. Another common set of basis functions for (3.7.9) consists of the singular functions of some compact linear operator $\mathcal{A} : L^2(\mathcal{I}) \to L^2(\mathcal{I}')$; recall briefly that the SVD provides two orthonormal bases $\{v_j\}_{j=1}^\infty$ and $\{u_j\}_{j=1}^\infty$, for respectively $L^2(\mathcal{I})$ and $L^2(\mathcal{I}')$, such that $\mathcal{A}v_j = \sigma_j u_j$, and hence

$$\mathcal{A}f = \sum_{j=1}^\infty \sigma_j (f, v_j)_{L^2(\mathcal{I})} u_j$$

Taking the singular functions $v_j$ as the basis for an expansion of the type (3.7.9) will then be convenient for applying the operator $\mathcal{A}$, since it will act diagonally:

$$\mathcal{A} \sum_{j=1}^m Z_j(\omega) v_j = \sum_{j=1}^m \sigma_j Z_j(\omega) u_j$$

Of course, for a given random process $f$ and operator $\mathcal{A}$, it is unlikely that the singular functions of
\[ \mathcal{A} \] are also the KLE functions for \( f \), so we can either have the coefficients in the expansion (3.7.9) be uncorrelated, or we can have the operator act diagonally [20].

In (3.7.9), we approximated a process by projecting it onto finite-dimensional span of some known basis functions. Another way to represent a process \( f \in \mathcal{H} \) is through a finite-dimensional vector of point samples, that is,

\[
\mathbf{u}_{\tau_1: \tau_m}(\omega) = [f(\tau_1, \omega), \ldots, f(\tau_m, \omega)]^T
\]  

(2.5.2)

The points \( \{\tau_1, \ldots, \tau_m\} \) can either lie on a regular grid or not (in fact, they can be randomly located). Then, given a sample (2.5.2), we can reconstruct an approximation to \( f \) by interpolation. For instance, given an interpolatory kernel function \( \eta(\tau) \), we can set

\[
f(\tau, \omega) \approx \sum_{j=1}^{m} w_j \eta(\tau - \tau_j)
\]  

(2.5.3)

where \( w_1, \ldots, w_m \) are some appropriately chosen weights. An approximation method consisting of (2.5.2) and (2.5.3) is called an interpolatory approximation. Random functions of the type (2.5.3) will reemerge in Section 2.6 under the name lumpy-type processes.

Note that in either case - the Galerkin-type or interpolatory-type - the representation of the process always consists of finitely many numbers. Because we do not necessarily require the points in (2.5.2) to lie on a fixed grid, a representation of the type (2.5.3) may actually be infinite-dimensional in the sense that functions of the type (2.5.3), where \( \{\tau_j\} \) are allowed to be arbitrary, do not lie in the span of some finite set of functions. Regardless, either (3.7.9) or (2.5.3) can be evaluated on a grid as fine as we choose; if a grid is used for the representation, this same grid is not required for evaluation.

2.6 A Survey of Random Process Models & their Simulation

In this section, we offer a small collection of spatial and spatiotemporal random processes that are useful in the modeling of physiological and imaging systems. We provide explicit mathematical
formulas for the characteristic functional, moment functions and finite-dimensional distributions whenever possible. We also discuss some practical simulation techniques and display some numerical realizations.

2.6.1 Gaussian and Lognormal Processes

The class of Gaussian processes is one of the most commonly known, studied and applied due to their relative tractability and broad applicability; see e.g. [230] [136] [261] [113] [181]. We will first discuss classical Gaussian processes, then discuss generalized Gaussian process and lognormal processes.

The complete statistical characterization of a classical Gaussian process relies on only the specification of a mean function \( \mu_f = \mu_f(\tau) \) and a covariance function \( k_f = k_f(\tau, \tau') \), where \( \tau, \tau' \in \mathcal{I} \). The covariance function is required to be Hermitian symmetric (i.e. \( k_f(\tau, \tau') = k_f^*(\tau', \tau) \)) and positive-definite; we will further assume that \( k_f \) is integrable on the diagonal (for instance, if \( \mathcal{I} \) is compact and \( k_f \) is bounded). Recall that with these conditions, \( k_f \) defines a trace class covariance operator \( C_f : L^2(\mathcal{I}) \to L^2(\mathcal{I}) \) via

\[
(C_f \phi)(\tau) = \int_{\mathcal{I}} k_f(\tau, \tau') \phi(\tau') \, d\tau'
\]

Given a mean function \( \mu \) and a valid covariance function \( k \), we say that \( f \sim \mathcal{N}(\mu, k) \) (or \( f \sim \mathcal{N}(\mu, C) \)) if \( f \) has characteristic functional [60], [26]

\[
\Psi_f[\phi] = \exp \left[ -2\pi i (\mu, \phi) \right] \exp \left[ -2\pi^2 (C \phi, \phi) \right] \quad (2.6.1)
\]

where

\[
(\mu, \phi) = \int_{\mathcal{I}} \mu(\tau) \phi^*(\tau) \, d\tau, \quad (C \phi, \phi) = \int_{\mathcal{I}} \int_{\mathcal{I}} k(\tau, \tau') \phi(\tau) \phi^*(\tau') \, d\tau d\tau'
\]

Note the similarity of (2.6.1) to the characteristic function of a Gaussian random vector, e.g. (2.6.2): the scalar product \( \xi^i \bar{u} \) and quadratic form \( \xi^i C \xi \) are generalized to their infinite-dimensional
counterparts \((\mu, \phi)\) and \((C\phi, \phi)\).

One of the key features of a Gaussian process is that any finite-dimensional sample is a multivariate Gaussian random vector. Indeed, let \(f \sim \mathcal{N}(\mu, k)\) where \(\mu\) and \(k\) are real-valued and continuous, and let \(n \geq 1, \tau_1, \ldots, \tau_n \in \mathcal{I}\). Then, note that \(\Psi f\) is well-defined for delta function inputs, since

\[
(C \delta(\cdot - \tau_j))(\tau) = k(\tau, \tau_j)
\]

and hence \((C \delta(\cdot - \tau_j), \delta(\cdot - \tau_k)) = k(\tau_j, \tau_k)\) (recall that \(k\) is symmetric). Thus, \(u_{\tau_1: \tau_n}\) has characteristic function

\[
\psi_{\tau_1: \tau_n}(\xi) = \Psi f \left[ \sum_{j=1}^{n} \xi_j \delta(\cdot - \tau_j) \right]
\]

\[
= \exp \left[ -2\pi i \sum_{j=1}^{n} \xi_j (\mu, \delta(\cdot - \tau_j)) \right] \exp \left[ -2\pi^2 \sum_{j,k=1}^{n} \xi_j \xi_k (C \delta(\cdot - \tau_j), \delta(\cdot - \tau_k)) \right]
\]

\[
= \exp \left[ -2\pi i \xi^t u_{\tau_1: \tau_n} \right] \exp \left[ -2\pi^2 \xi^t C_{\tau_1: \tau_n} \xi \right]
\]  (2.6.2)

where the mean vector and covariance matrices are given by

\[
\bar{u}_{\tau_1: \tau_n}(j) = \mu(\tau_j), \quad C_{\tau_1: \tau_n}(i, j) = k(\tau_i, \tau_j).
\]  (2.6.3)

Assuming \(C_{\tau_1: \tau_n}\) is invertible, we can apply the inverse Fourier transform to (2.6.2), we obtain the PDF of \(u_{\tau_1: \tau_n}\):

\[
p_{\tau_1: \tau_n}(x) = \frac{(2\pi)^{-n/2}}{\sqrt{\det(C_{\tau_1: \tau_n})}} \exp \left( -\frac{1}{2} (x - \bar{u}_{\tau_1: \tau_n})^t C_{\tau_1: \tau_n}^{-1} (x - \bar{u}_{\tau_1: \tau_n}) \right)
\]  (2.6.4)

If \(C_{\tau_1: \tau_n}\) is not invertible, we can still define the PDF of \(u_{\tau_1: \tau_n}\) using the pseudoinverse \(C_{\tau_1: \tau_n}^+\) in place of the inverse; the resulting distribution will be concentrated on a hyperplane spanned by the principal components (singular vectors) corresponding to nonzero eigenvalues of \(C_{\tau_1: \tau_n}\).

The finite-dimensional distributions (2.6.4) provide a reasonably straightforward method to
draw samples from $\mathcal{N}(\mu, k)$: simply choose sample locations $\tau_1, \ldots, \tau_n \in \mathcal{I}$, form $\bar{u}_{\tau_1:\tau_n}$ and $C_{\tau_1:\tau_n}$ using (2.6.3), then use an existing multivariate Gaussian sampler to draw from $\mathcal{N}(\bar{u}_{\tau_1:\tau_n}, C_{\tau_1:\tau_n})$.

However, this can be very inefficient if $n$ is large, for instance sampling a fine grid of points in 3D: Cholesky factorization is $O(n^3)$ and matrix multiplication is $O(n^2)$, so to draw $K$ samples on a grid with $m$ points per dimension, this method would be $O(m^9)$ for the first sample, then $O(Km^6)$ thereafter. Faster specialized methods are available if for instance $k$ is shift-invariant, including the circulant embedding method; see e.g. [181].

In addition to the sample-based finite-dimensional distributions (2.6.4), it is also useful to have the density of scalar random variables of the sort $X_\phi = (f, \phi) = \int_{\mathcal{I}} f(\tau)\phi(\tau)\ d\tau$, i.e. a scalar product with a fixed test function. Such a random variable has (univariate) PDF $p_\phi(x)$ given by

$$p_\phi(x) = \frac{1}{\sqrt{2\pi\sigma^2_\phi}} \exp\left(-\frac{(x - m_\phi)^2}{2\sigma^2_\phi}\right)$$

(2.6.5)

where $\sigma_\phi = (C\phi, \phi)$ and $m_\phi = (\mu_f, \phi)$, with $C$ and $\mu_f$ respectively the covariance operator and mean function of the process $f$.

We can also define a generalized Gaussian random process by letting $\mu \in \mathcal{T}'$ be an arbitrary generalized function and letting $C : \mathcal{T} \rightarrow \mathcal{T}'$ be an arbitrary continuous, positive definite operator (see [100]); the definition (2.6.1) still defines $f \sim \mathcal{N}(\mu, C)$, but the scalar product must now be interpreted as a generalized function pairing, i.e. $(\cdot, \cdot) = (\cdot, \cdot)_{\mathcal{T}', \mathcal{T}}$.

One application of (2.6.1) is to provide a rigorous description of the Gaussian white noise process $w$. While somewhat challenging to describe using standard probabilistic definitions, the characteristic functional provides a direct method to define a process which has mean zero and ‘independent standard normal values at every point’. Intuitively, one would like the covariance function to be zero for $\tau \neq \tau'$, but nonzero for $\tau = \tau'$; the logical choice is a Dirac delta covariance, that is, $k(\tau, \tau') = \delta(\tau - \tau')$. This leads to a covariance operator $C$ which is the identity operator, and hence the characteristic functional [100]

$$\Psi_w[\Phi] = \exp(-2\pi^2(\phi, \phi)) = \exp\left(-2\pi^2\int_{\mathcal{I}} |\phi(\tau)|^2\ d\tau\right)$$

(2.6.6)
A fairly wide variety of random processes can be obtained under the Gaussian model by choosing different mean and covariance functions. The only requirement on $\mu$ and $k$ is that $k$ be non-negative-type, which means that any matrix of the form $C_{\tau_1,\tau_n}(i,j) = k(\tau_i, \tau_j)$ must be symmetric and non-negative definite i.e. a valid covariance matrix. Several common examples of covariance functions are provided in Table 2.1. In Figure 2.7 we display realizations for several choices of $k$.

<table>
<thead>
<tr>
<th>#</th>
<th>Name</th>
<th>Covariance function $k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1)</td>
<td>Gaussian</td>
<td>$k(\tau, \tau'; \sigma, A) = \sigma^2 \exp\left(-\frac{1}{2} (\tau - \tau') A (\tau - \tau')\right)$</td>
</tr>
<tr>
<td>(2)</td>
<td>Exponential</td>
<td>$k(\tau, \tau'; \sigma, \ell) = \sigma^2 \exp(-|\tau - \tau'|/\ell)$</td>
</tr>
<tr>
<td>(3)</td>
<td>Matérn-Whittle</td>
<td>$k(\tau, \tau'; \sigma, \nu, \ell) = \sigma^2 \frac{\Gamma(\nu)}{2^{\nu-1}\Gamma(\nu)} \left(\frac{|\tau - \tau'|}{\ell}\right)^{\nu} K_\nu\left(\frac{|\tau - \tau'|}{\ell}\right)$</td>
</tr>
<tr>
<td>(4)</td>
<td>Bessel</td>
<td>$k(\tau, \tau'; \sigma, \nu) = \sigma^2 \Gamma(d/2) J_\nu\left(\frac{|\tau - \tau'|}{|\tau - \tau'|/2}\right)$</td>
</tr>
<tr>
<td>(5)</td>
<td>White noise</td>
<td>$k(\tau, \tau') = \delta(\tau - \tau')$</td>
</tr>
</tbody>
</table>

Table 2.1: A short list of common covariance functions used in spatial random process modeling, found in refs. [181, 57, 100, 261, 189]. The special functions $\Gamma(\cdot), K_\nu, J_\nu$ are respectively the Gamma function, modified Bessel function of the second kind, and Bessel function of the first kind. The matrix $A$ in (1) must be symmetric nonnegative definite. Covariances (2)-(4) are their isotropic versions; anisotropic variants can also be defined. The covariance (5) must be interpreted as the generalized kernel of an integral operator.

In Appendix 6.2 we derive the Matérn-Whittle class of isotropic covariance functions as arising from solving a pseudodifferential equation of the form

$$(I - \nabla^2)^{s/2} u_\nu = w$$

where $w$ is a Gaussian white noise, and $(I - \nabla^2)^{s/2}$ is defined via its Fourier multiplier:

$$\mathcal{F}[(I - \nabla^2)^{s/2} f](\xi) = \left(1 + 4\pi^2 \|\xi\|^2\right)^{s/2} \hat{f}(\xi)$$

The resulting class of covariances contains both the exponential and the Gaussian as special cases; the exponential represents the case of $\nu = n/2$, while $\nu \to \infty$ corresponds to the Gaussian case. The smoothness of the realizations of a Matérn random field increases with $\nu$, with $\nu = n/2$ representing...
\( \nu = 2, \ell = 1/40 \quad \nu = 2, \ell = 1/20 \quad \nu = 1, \ell = 1/20 \quad \nu = 1, \ell = 1/10 \)

\[ a_{11} = 50 \quad a_{11} = 500 \quad a_{11} = 100 \quad a_{11} = 1000 \]
\[ a_{22} = 50 \quad a_{22} = 500 \quad a_{22} = 2000 \quad a_{22} = 100 \]
\[ a_{12} = a_{21} = 0 \quad a_{12} = a_{21} = 0 \quad a_{12} = a_{21} = 0 \quad a_{12} = a_{21} = 300 \]

Figure 2.7: Simulated Gaussian random processes \( f(r) \) for \( r \in V = [0, 1]^2 \). The top row uses the isotropic Matérn-Whittle covariance function (function (3) in Table 2.1) with parameters indicated above, while the bottom row uses the Gaussian covariance (function (1) in Table 2.1) with \( A = (a_{ij}) \) indicated above. All samples use variance \( \sigma^2 = 1 \) and were computed using Algorithm 7.6 in [181].

Fairly rough realizations while \( \nu \to \infty \) corresponds to very smooth realizations (see also Figure 2.7).

Given a Gaussian random process \( f \sim \mathcal{N}(\mu, \mathcal{C}) \), we can form a lognormal random process by exponentiating the realizations of \( f \) pointwise as follows:

\[
  u(\tau) = \exp(f(\tau))
\]

We would then say that \( u \) is a lognormal process, written \( u \sim \mathcal{LN}(\mu, \mathcal{C}) \) since \( \ln(u) \sim \mathcal{N}(\mu, \mathcal{C}) \). A lognormal random process is an example of a non-Gaussian process formed by applying a pointwise (sometimes called memoryless [114]) transformation to a Gaussian process. Note that we are specifying the mean and covariance operator of the log-transformed process, i.e. \( \mu \) is the mean of \( \ln(u) \) and \( \mathcal{C} \) is the covariance operator of \( \ln(u) \).
The lognormal random process is an example of a process for which, despite the characteristic functional being well-defined for every test function $\phi$, an explicit functional form for $\Psi_u[\phi]$ (such as (2.6.1)) is unfortunately not available [183].

Simulating a lognormal random process (indeed any pointwise transformed Gaussian) is trivially equivalent to simulating a Gaussian random process – one first generates a sample $f_j \sim \mathcal{N}(\mu, \sigma^2)$, then produces $u_j = \exp(f_j)$.

### 2.6.2 Poisson Point Processes and Cox Processes

In this section, we introduce a particular class of generalized stochastic process that is quite useful in both physiology and image science: the class of point processes. General references for point processes include [233] [259] [67] [71] [72] [169]. Barrett and Myers [26] include a fairly extensive discussion of point processes in Chapter 11. In Chapter 3 of this work, we will require point processes on fairly general domains, so we give a correspondingly general treatment here. In this section, we discuss only theoretical aspects of point processes; in Section 2.6 we will discuss the simulation of point processes.

As usual, let $(\Omega, \mathcal{F}, \mathbb{P})$ be a background probability space. We consider a set $\mathcal{A} \subset \mathbb{R}^k$, which we call the attribute set for reasons that will become clear in Chapter 3. In that Chapter, elements of $\mathcal{A}$ will parameterize physical attributes of a photointeraction event occurring within a high-energy radiation detector. More generally, point processes can describe physical processes which occur with discrete, localized physical attributes such as position and momentum. A realization of a point process on $\mathcal{A}$ is a (generalized) function of the type

$$f(a) = \sum_{j=1}^{n} \delta(a - a_j)$$

(2.6.7)

where $A = \{a_j\}_{j=1}^{n} \subset \mathcal{A}$ and $n \in \mathbb{N}$. As a generalized function, we have that for a test function $\phi \in \mathcal{T}$ (where $\mathcal{T}$ is any reasonable test function space such as the Schwartz class $\mathcal{S}$),

$$(f, \phi) = \sum_{j=1}^{n} \phi(a_j)$$
One can also think of a point process as a discrete measure, that is, a measure of the form

\[ \mu_A = \sum_{j=1}^{n} \delta_{a_j} \]

Such a measure would act on an event set \( E \subset A \) via \( \mu(E) = \#(A \cap E) \) where \( \#(\cdot) \) is the number of elements of the set. We will maintain the generalized function viewpoint, in order to make use of the theory developed earlier.

To make (2.6.7) into a random process, we must make \( f \) a function of \( \omega \in \Omega \) as well. The parameters that describe the realizations are the number of points \( n \) and the point locations \( A = \{a_j\} \); thus, a random point process will simply make each of these entities random.

The way in which (2.6.7) is defined can be fairly general: the points need not be selected independently from each other (for instance, they could be generated by a Markov chain), and the entire method by which the points are generated can even be randomized. The most general description of a point process is that is a random process taking values in the subspace of \( T' \) consisting of at-most-countable sums of delta functions (i.e. (2.6.7) with \( n \leq \infty \)), such that the points \( a_j \) do not ‘accumulate’ \( ^{12} \). We discuss two specific point process models: the classical Poisson point process and the Cox process.

In a standard Poisson point process, we define a function \( \lambda(a) \) for \( a \in A \) called the intensity function; we require only that \( \lambda(a) \geq 0 \) for all \( a \in A \) and that \( \lambda \in L^1(A) \). We then define the number

\[ \bar{N} = \int_A \lambda(a) \, da, \]

which is the mean number of points (\( \bar{N} \) is not required to be an integer). Lastly, we define the probability density function for \( x \in A \) via

\[ p_a(x) = \frac{\lambda(x)}{\bar{N}} \quad \text{(2.6.8)} \]

---

\(^9\) As usual, generalizations are possible: we do not actually need an intensity function, but really only an intensity measure; we will assume that all PPPs have \( L^1(A) \) intensity functions.
To sample from the Poisson Point Process (PPP) with intensity \( \lambda \), we first draw \( N \sim \text{Poi}(\bar{N}) \), resulting in realization \( N = n \). Then, we consider \( n \) I.I.D. samples from the random vector \( \mathbf{a} \) with PDF (2.6.8), which results in \( \mathbf{a} = \mathbf{a}_1, \ldots, \mathbf{a} = \mathbf{a}_n \). Then, we form the process as in (2.6.7):

\[
f(\mathbf{a}) = \sum_{j=1}^{n} \delta(\mathbf{a} - \mathbf{a}_j).
\]

Realizations of three PPPs with a common \( p_{\mathbf{a}}(\mathbf{r}) \) but different intensities \( \lambda(\mathbf{r}) \) are shown in Figure 2.8.

\[\begin{align*}
\bar{N} &= 1000, \ n = 1013 \\
\bar{N} &= 5000, \ n = 5086 \\
\bar{N} &= 10000, \ n = 9920
\end{align*}\]

Figure 2.8: Simulation of three Poisson point processes with fixed PDF \( p(\mathbf{r}) = \lambda(\mathbf{r})/\bar{N} \) and varying \( \bar{N} \). In each panel, a single realization of \( f(\mathbf{r}) \) is displayed (white dots) over the color intensity plot of \( p(\mathbf{r}) \). The mean number of points \( \bar{N} \) and the realized number of points \( n \) are displayed above each panel. The intensity function \( \lambda(\mathbf{r}) \) is a realization of a lumpy background process, discussed in Section 2.6.3.

The process described above generates a probability measure on \( \mathcal{F}' \), which we denote \( \text{Poi}(\lambda) \) (there should be no confusion between the PPP with intensity function \( \lambda \) and a Poisson random number with mean \( \bar{N} \)). Note that in the abstract point process literature (e.g. [71] [233] [169]), \( \text{Poi}(\lambda) \) would be described as giving rise to random elements of a space of measures, and hence \( \text{Poi}(\lambda) \) would technically be a probability measure on a space of measures, instead of on \( \mathcal{F}' \).

As discussed in Section 2.2.6, it is only possible to define a PDF for probability measures on infinite-dimensional spaces such as \( \mathcal{F}' \) when an appropriate reference measure is defined, and as we saw, in many cases the choice of reference measure is not at all obvious. This is not the case for Poisson point processes: in fact, because we are assuming that every PPP has an intensity function
\(\lambda\), we can prove the following:

**Theorem 3** (Adapted from [169, Theorem 1.3], attributed to [37]). Let \(\text{Poi}(\lambda)\) be a Poisson point process with intensity function \(\lambda \in L^1(A)\). Then, the measure \(\mathbb{P}_\lambda \equiv \text{Poi}(\lambda)\) is absolutely continuous with respect to \(\mathbb{P}_0 = \text{Poi}(\lambda_0)\), where \(\lambda_0\) is the uniform intensity function, taking the value 1 uniformly on \(A\), and the PDF (Radon-Nikodym derivative) of \(\mathbb{P}_\lambda\) with respect to \(\mathbb{P}_0\) is given by

\[
\frac{d\mathbb{P}_\lambda}{d\mathbb{P}_0}(f) = C \exp(-\bar{N}) \prod_{j=1}^n \lambda(a_j) = C \exp(-\bar{N}) \bar{N} \prod_{j=1}^n p_a(a_j) \tag{2.6.9}
\]

where \(f \equiv f(a) = \sum_{j=1}^n \delta(a - a_j)\), \(C\) is a constant independent of \(f\) and \(\lambda\), \(\bar{N} = \int_A \lambda(a)\,da\), and \(p_a(a) = \lambda(a)/\bar{N}\).

To paraphrase, one can indeed define a probability density function for the generalized stochastic process \(\text{Poi}(\lambda)\), if one takes as the ‘reference’ measure another Poisson point process with uniform intensity measure. The astute reader will notice the similarity between (2.6.9) and the so-called list-mode likelihood formula [23] [175] [26]; we will indeed see (2.6.9) return as a likelihood for data that takes the form of a Poisson point process.

In the above, we considered a fixed function \(\lambda\) such that \(\lambda(a) \geq 0\) for all \(a \in A\) and \(\lambda \in L^1(A)\). Supposing instead that \(\lambda\) were a realization of a stochastic process \(f\), we obtain a doubly stochastic or Cox Poisson point process [66] [67] [259] [26]. We must assume that the realizations of \(f\) are almost surely non-negative, and almost surely \(L^1(A)\); for example, a lognormal process \(f \sim \mathcal{L}\mathcal{N}(\mu, k_f)\) (discussed in Section [2.6] on a bounded set \(A\) with continuous covariance function would have nonnegative realizations which were almost surely integrable; the various classes of generalized lumpy backgrounds, also discussed in Section [2.6] are also possible candidates.

To derive the characteristic functional \(\Psi_f[\phi]\) for \(f \sim \text{Poi}(\lambda)\), we let \(\phi \equiv \phi(a) \in \mathcal{T}\) be a test function. Then,

\[
(f, \phi) = \sum_{j=1}^N \phi(a_j)
\]
where $N \sim \text{Poi}(\bar{N})$ and $a_j \sim p_{\alpha}(x)$ are the random quantities discussed above. Exponentiation gives a random product:

$$\exp \left[ -2\pi i (f, \phi) \right] = \prod_{j=1}^{N} \exp \left[ -2\pi i \phi(a_j) \right]$$

We then compute the expected value using the law of total expectation and the definition of the random vector $a_j$:

$$\Psi_f[\phi] = \left\langle \exp \left[ -2\pi i (f, \phi) \right] \right\rangle_{\{a_j\}|N}$$

$$= \left\langle \left( \int_{A} \exp(-2\pi i \phi(a)) \frac{\lambda(a)}{N} \, da \right)^{N} \right\rangle_{N}$$

$$= \sum_{n=0}^{\infty} \frac{q_{\phi}^n \exp(-N)N^n}{N^n n!}$$

$$= \exp(-N) \exp(q_{\phi})$$

where we have defined $q_{\phi} \equiv \int_{A} \exp(-2\pi i \phi(a)) \lambda(a) \, da$. Recalling the definition of $N$, we can simplify once more to obtain

$$\Psi_f[\phi] = \exp \left[ \int_{A} (\exp(-2\pi i \phi(a)) - 1) \lambda(a) \, da \right]$$

This functional is of the special form

$$\Psi_f[\phi] = \exp \left[ \int_{A} F(\phi(a)) \, da \right], \quad (2.6.10)$$

which Unser et al. call an innovation process or Lévy white noise [274]; in a sense, the Poisson point process is a non-Gaussian analogue of the Gaussian white noise process, which has $F(\phi) = \frac{1}{2}|\phi|^2$.

### 2.6.3 Generalized Lumpy Backgrounds

We now consider filtered versions of point processes of the form $(2.6.7)$, whereby we compute a random process $g(a) = (Bf)(a)$ where $B$ is a linear operator (appropriately defined for generalized
function inputs of the form (2.6.7). For instance, suppose $\mathcal{A} = \mathbb{R}^k$ and $h(a)$ is a real, continuous, compactly supported function. Then, define $g = \mathcal{B}f = h * f$ to be the convolution of $f$ with $h(a)$:

$$g(a) = (h * f)(a) = \int_{\mathbb{R}^k} h(a - a') \sum_{j=1}^{N} \delta(a' - a_j) \, da' = \sum_{j=1}^{N} h(a - a_j)$$  \hspace{1cm} (2.6.11)$$

The characteristic functional of (2.6.11) then follows by the general relation for linearly transformed random processes, namely that $\Psi_g[\phi] = \Psi_f[\mathcal{B}^\dagger \phi]$. With $\mathcal{B} = h*$, we have $\mathcal{B}^\dagger = \tilde{h}$ where $\tilde{h}(a) = h(-a)$, and so

$$\Psi_g[\phi] = \exp \left[ \int_{\mathbb{R}^k} \left( \exp(-2\pi i (\tilde{h} * \phi)(a)) - 1 \right) \lambda(a) \right]$$

More generally, suppose that $h(a, a')$ is any continuous function on $\mathcal{A} \times \mathcal{A}$, then define

$$(\mathcal{B}f)(a) = \int_{\mathcal{A}} h(a, a') f(a') \, da'$$

Then, if $f$ is a point process of the form (2.6.7), we have

$$g(a) = (\mathcal{B}f)(a) = \sum_{j=1}^{N} h(a, a_j)$$  \hspace{1cm} (2.6.12)$$

and

$$\Psi_g[\phi] = \exp \left[ \int_{\mathbb{R}^k} \left( \exp(-2\pi i (\mathcal{B}^\dagger \phi)(a)) - 1 \right) \lambda(a) \right]$$

where $\mathcal{B}^\dagger$ is the integral operator with kernel $\tilde{h}(a, a') = h(a', a)$.

Taking (2.6.11) and (2.6.12) as a starting point, one can construct a wide array of possible random processes by choosing a different kernel functions $h$, or even allowing multiple kernel
functions $h_j(a)$ in the sum (2.6.11). In general, we call a random process of the type

$$f(a) = \sum_{j=1}^{N} h_j(a; \theta_j)$$  \hspace{1cm} (2.6.13)

a \textit{generalized lumpy background process}. The random variables that generate (2.6.13) are $N$ and $(\theta_1, \ldots, \theta_N)$. The functions $h_j(a; \theta)$ need not be the same type of function nor do they need to form a basis for any function space. Functions similar to (2.6.13) have been called \textit{kernel density estimators} and \textit{mixture models} [255], \textit{shot noise} [259, 74], \textit{lumpy backgrounds} [238, 32] and \textit{texton noise} [94]. The model (2.6.13) can be evaluated in parallel at very high speed, and $h_j$ can be chosen to match observed texture statistics [94, 168] or display desired regularity, nonnegativeness, boundedness, or any other desired sample function property. In particular, (2.6.13) can be made highly non-stationary and non-Gaussian, making it a particularly appealing class of models for physiological processes. We display several examples of generalized lumpy background processes in Figure 2.9.

Figure 2.9: Three realizations of generalized lumpy background processes of the form (2.6.13) on the unit square $V = [0,1]^2$. On the left, each lump takes the form $h_j(r) = A_j \exp(-\|r\|^2/2\sigma_j^2)$ with $\sigma_j^2$ drawn randomly from a lognormal distribution. In the middle, each lump takes the form $h_j(r) = A \exp(-r^t B_j r/2)$ where $B_j$ is a $2 \times 2$ positive definite matrix formed by rotating a matrix $B_0$ randomly. On the right is a \textit{clustered} lumpy background [32] where each lump is of the form $h_j(r) = \sum_{k=1}^{n_j} h(r - r_{jk})$. 

108
2.7 Review of Statistical Inference

One of the main topics we discuss in this dissertation is the usage of imaging data to make inferences and decisions about a patient. We have discussed ways in which both patient anatomy and image data can be described as samples from random processes. Now, imagine that we have access only to some random imaging data \( g \in \mathcal{Y} \), and we wish to make inferences and decisions concerning the patient (whom we assume to be parameterized by \( f \in \mathcal{X} \)). We have now entered the realm of statistics: one of the key viewpoints of this work is that all imaging problems can be thought of as statistical inference problems. The probability and random process theory discussed previously gives us a way to compute image statistics given knowledge of the object (patient), but we will usually be faced with the reverse problem - determining properties of the object given the image - this is an inverse problem. We will discuss inverse problems more in Chapter 3 but for now we simply say that an inverse problem consists of using measured data to make inferences about some object which was not directly measured.

One of the main tools in statistical inference is a function called the likelihood. While the classical likelihood is a familiar device to all advanced statistics students, we must employ a more sophisticated version: both the objects we wish to make inferences about and our sampled data can in principle be infinite-dimensional in nature, so we must define appropriate likelihoods in this setting. In the statistics literature, we would say that our data are usually functional and our inference problem might be nonparametric or semiparametric\[284][229][88][277]. We describe both of these terms below.

2.7.1 Statistical Experiments and Models

A statistical experiment consists of measuring a random quantity \( g \in \mathcal{Y} \), which for now we assume only to be a generalized random vector, as discussed in Section 2.2.6. It is assumed that the distribution of \( g \), written in the abstract measure picture as \( P_g \), is an element of a certain class of measures, written \( \mathcal{M} \) (for ‘model’). In other words, it is assumed that there is a distribution \( P \in \mathcal{M} \) such that \( P = P_g \). Otherwise, the model \( \mathcal{M} \) is said to be misspecified\[135][277][159]. The central problem of statistical inference is to the estimate properties of \( P_g \) from a sample of \( g \).
We made the following distinction between so-called ‘parametric’ statistical models and ‘non-parametric’ models: a parametric model is one in which there exists a set $\Theta$ which is a subset of a finite-dimensional vector space, such that every element of $\mathcal{M}$ is specified by some $\theta = [\theta_1, \ldots, \theta_p]^T \in \Theta$. A nonparametric model is one in which infinitely many parameters are required to fully describe the distributions in $\mathcal{M}$. In the nonparametric case, we will still use the symbol $\theta \in \Theta$ to indicate a parameter used to describe $P \in \mathcal{M}$, but in the nonparametric case, $\Theta$ is a subset of an infinite-dimensional space, such as a space of functions or infinite sequences. A third class of models is sometimes called semi-parametric if the parameter can be written as

$$\theta = (\theta_0, \theta_1) \in \Theta_1 \times \Theta_2$$

where $\theta_0$ is a finite-dimensional ‘parameter of interest’ and $\theta_1$ is a (possibly infinite-dimensional) nuisance parameter. Semiparametric models are of particular interest in medical imaging because the finite-dimensional parameter of interest represents some critical patient-specific parameters while the infinite-dimensional nuisance parameter represents the rest of the information contained in the object that either does not contribute to these key parameters or is not of direct interest.

If the data space $\mathcal{Y}$ is a (subset of a) finite-dimensional vector space, we say our data are classical, whereas if $\mathcal{Y}$ is infinite-dimensional, we say that our data are generalized or functional. This distinction is not exactly the same as that made in the statistics community: there, ‘functional data’ is a term typically used to describe something that could, in principle, be sampled infinitely finely; we will actually see in Chapter 3 that it is possible to observe ‘truly’ infinite-dimensional data with a real imaging system.

To illustrate the difference between parametric and nonparametric models, consider an elementary example: suppose that $g = [G_1, \ldots, G_m]$ is a standard I.I.D. random sample from a univariate normal distribution, i.e. $G_i \sim N(\mu, \sigma^2)$. We know that the $G_i$ are identically normally distributed (somehow), we just don’t know $\mu$ and $\sigma^2$. Thus the probability density for $g$ is given by a product
of \(N(\mu, \sigma^2)\)s:

\[
p_g(x) = \prod_{j=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(x_i - \mu)^2\right) \tag{2.7.1}
\]

Thus a correctly specified model is given by \(\mathcal{M} = \{\otimes_{j=1}^{m} N(\mu, \sigma^2) : \mu \in \mathbb{R}, \sigma^2 > 0\}\), where the \(\otimes\) notation indicates that the PDF takes the product form \(2.7.1\). Since only two parameters \((\mu\) and \(\sigma^2\)) are required to specify a distribution from this model, it is parametric. Now, suppose that we still know that \(G_i\) are I.I.D. but didn’t know that the \(G_i\) were normally distributed: we only know that \(P_{G_i}\) is absolutely continuous with some PDF \(p_{G_i}(x)\). Then, we must take \(\mathcal{M}\) to be set of all product densities:

\[
p_p(x) = \prod_{j=1}^{m} p(x_j) \tag{2.7.2}
\]

where \(p(\cdot)\) is an arbitrary PDF. Thus in order to be well-specified, we must take the model \(\mathcal{M} = \otimes_{j=1}^{m} \mathcal{P}_{ac}(\mathbb{R})\), and this is an infinite-dimensional set. To see why, suppose \(m = 1\) so that \(\mathcal{M} = \mathcal{P}_{ac}(\mathbb{R})\), then note that the Gaussian mixture distributions \(p(x) = \sum_{k=1}^{K} \lambda_k N(\mu_k, \sigma_k^2)\), where \(\lambda_k > 0\) and \(\sum_k \lambda_k = 1\) are contained in \(\mathcal{M}\); taking the infinite family \(N(k,1)\) where \(k \in \mathbb{Z}\) and forming arbitrary mixtures shows that \(\mathcal{M}\) cannot be described by any finite-dimensional parameter.

To illustrate the difference between classical and functional data, we suppose that instead of measuring a random variable \(X\) repeatedly, we are sampling a continuous function of time \(f(t)\), at some locations \(t_1, \ldots, t_n\). Treating \(f(t) \equiv f(t, \omega)\) as a stochastic process, we see that a single sample corresponds to \(\omega = \omega_1\), and thus consists of the \(n\)-dimensional vector \(g = [f(t_1, \omega_1), \ldots, f(t_n, \omega_1)]^\top\); however we can also sample multiple (say \(m\)) realizations, resulting in a dataset \([g_1; \ldots; g_m] \in \mathbb{R}^{n \times m}\). On the face of it, this is still classical data: we have sampled a finite-dimensional random quantity \(g \in \mathbb{R}^n\), \(m\) times, resulting in a dataset \(G \in \mathcal{X}\) where \(\mathcal{X}\) is finite-dimensional. However, the choice of the sample times \(t_1, \ldots, t_n\) for each sample is arbitrary: \(n\) could be made arbitrarily large and the times could be different for each sample. Thus in reality we should model the data as \(\{g_1, \ldots, g_m\}\) where each \(g_k \in \mathbb{R}^{nk}\); because the next sample may have more time points, it is not possible to model this dataset as a finite-dimensional object. In Section 2.6.2 we discussed Poisson
Point Processes (PPPs). Each sample of a PPP takes the form

\[ u(a) = \sum_{j=1}^{n} \delta(a - a_j) \]

Because the \( a_j \) are random, such a (generalized) function does not lie in any finite-dimensional space, hence if our data are a sample from a PPP, it is functional by our definition. In Chapter [3], we discuss a particular imaging system which results in PPP datasets.

The elements of a model \( \mathcal{M} \) are probability distributions (i.e. measures or densities) on the space \( \mathcal{Y} \), which is the space where the dataset \( g \) takes its values. For instance, if \( g \) is an \( m \)-dimensional random vector, elements of \( \mathcal{M} \) are probability densities on \( \mathbb{R}^m \); if \( g \) is a Poisson point process, elements of \( \mathcal{M} \) are technically probability measures on \( \mathcal{D}' \), but since a PPP is parameterized by its intensity function \( \lambda \in L^1(\mathcal{A}) \), this is not as pathological as it sounds. As we have seen, probability measures can either be described by finitely many parameters, or may require an infinite-dimensional parameter to describe. This is true both when the data space \( \mathcal{Y} \) is finite-dimensional and infinite-dimensional: for instance it is possible to describe a stochastic process \( f \) using only finitely many parameters. For example, assume that \( f \sim \mathcal{N}(0, k_f(\theta)) \), where \( k_f(\theta) \) is a covariance function parameterized by a single number \( \theta \); this is a parametric description of an infinite-dimensional object (\( \mathcal{N}(0, k_f) \) is a measure on \( L^2(T) \)). If we did not want to model \( k_f \) in such a way, instead treating \( k_f \) as an arbitrary positive definite covariance function, we would instead have a nonparametric description of an infinite-dimensional object.

### 2.7.2 The Likelihood

We now introduce the concept of the likelihood in both the classical and functional settings. Consider an experiment \( g \) with realizations in the space \( \mathcal{Y} \) and a model \( \mathcal{M} \), where elements of \( \mathcal{M} \) are specified by the (possibly infinite-dimensional) parameter \( \theta \in \Theta \). For convenience, we consider the classical case by supposing for that elements of \( \mathcal{M} \) are absolutely continuous probability densities.
on $\mathbb{R}^m$, i.e. we can write an element of $\mathcal{M}$ as

$$p_\theta(g), \quad \theta \in \Theta, \quad g \in \mathbb{R}^m$$

Note again $p_\theta(g)$ is a finite-dimensional PDF with possibly infinite-dimensional parameter. For now, we are simply treating $\theta$ as a parameter that specifies the density. It is also common to use the conditional probability notation

$$p_\theta(g) = p(g|\theta), \quad (2.7.3)$$

This notation is technically only appropriate when we can reasonably treat $(\theta, g)$ as a jointly varying random quantity, with some joint probability distribution. This perspective is the starting point of the Bayesian approach to statistics, which we discuss briefly later, but it is very common to use (2.7.3) even in the case when $\theta$ is non-random, and we will adhere to this convention.

When the experiment is performed, the random vector $g$ is observed to be some value $g$, i.e. $g = g \in \mathbb{R}^m$ is the outcome of the experiment. Then, for any given $\theta \in \Theta$, we can evaluate the resulting probability density for the (now fixed) value $g$. The result is a function $\ell : \Theta \to [0, \infty)$ called the classical likelihood, defined as

$$\mathcal{L}(\theta) = \mathcal{L}(\theta|g) = p(g|\theta) \quad (2.7.4)$$

Note that because we allow $\theta$ to be infinite-dimensional, (2.7.4) can be a functional (that is, a function of an infinite-dimensional input). However, for each $\theta \in \Theta$, $p(g|\theta)$ is a probability density function on $\mathbb{R}^m$. We also define the log-likelihood as the natural logarithm of (2.7.4) (taking the value $-\infty$ if $\mathcal{L}(\theta) = 0$):

$$\ell(\theta) = \ell(\theta|g) = \ln(p(g|\theta)) \quad (2.7.5)$$

Now suppose that the data $g \in \mathcal{Y}$ is not a finite-dimensional random vector, but rather a generalized random vector, for instance, a Poisson point process or some other stochastic process.
The definition of likelihood in this case is not as obvious: we would like to use (2.7.4), but as we discussed in Section 2.2.6 defining a probability density on an infinite-dimensional set \( \mathcal{Y} \) requires the selection of a reference measure in order to invoke the Radon-Nikodym theorem, and the choice of reference measure is not always obvious. For our purposes, the only functional data we work with takes the form of a Poisson Point Process (PPP), and in that case a good reference measure on \( \mathcal{Y} \) is readily available in the form of a PPP with uniform intensity (as discussed in Theorem 3). Likelihoods in the case where \( g \) is a Gaussian random process are also tractable; see e.g. [266], [105].

The likelihood for a PPP is surprisingly simple, despite the infinite-dimensional nature of the data space \( \mathcal{Y} \). In fact, we have already provided the formula for the PPP likelihood in Theorem 3, but we can describe it in a more elementary way as follows; refer also to [23], [43] and [26]. Recall that a realization of a Poisson point process \( u \sim \text{Poi}(\lambda) \) consists of a random number of points \( n \) and a collection of points \( A = \{a_j\}_{j=1}^n \subset \mathcal{A} \), where each \( a_j \) is sampled I.I.D. from

\[
p_a(x) = \frac{\lambda(x)}{\bar{N}(\lambda)}, \quad \text{where} \quad \bar{N}(\lambda) = \int_{\mathcal{A}} \lambda(a) \, da.
\]

We thus associate the process \( u(\cdot, \omega) \) with the pair \((n, A)\), and we can think of a probability ‘density’ \( p(n, A) \), which factorizes as \( p(n, A) = p_N(n)p(A) \) because the points \( a_j \) are selected independently from \( n \). In other words, the probability of observing the point process \((n, A)\) can be written as the product of observing \( N = n \) and observing the collection of points \( A = \{a_j\}_{j=1}^n \). Since the \( a_j \) are I.I.D., we can also write this as \( p(n, A) = p_N(n)p_a(a_1) \cdots p_a(a_n) \). Since \( n \) is drawn from \( \text{Poi}(\bar{N}) \) and each \( a_j \) is drawn I.I.D. from the probability density \( p_a(x) = \lambda(x)/\bar{N} \), we can write

\[
p(n, A|\lambda) = \exp(-\bar{N})\frac{\bar{N}^n}{n!} \prod_{j=1}^n \frac{\lambda(a_j)}{\bar{N}} = \exp(-\bar{N})\frac{\bar{N}^n}{n!} \prod_{j=1}^n \lambda(a_j).
\]

Thus the likelihood function for a Poisson point process with intensity function \( \lambda \in L^1(\mathcal{A}) \) can be
The likelihood (2.7.6) is an example of a likelihood function whose input is infinite-dimensional (since $\lambda \in L^1(A)$). Taking the log and ignoring an additive constant, we have the log-likelihood function for a Poisson point process

$$\ell(\lambda) = -\bar{N}(\lambda) + \sum_{j=1}^n \ln(\lambda(a_j)) = -\int_A \lambda(a) \, da + \sum_{j=1}^n \ln(\lambda(a_j))$$  \hspace{1cm} (2.7.7)

Note that (2.7.6) is identical to the formula derived rigorously in Theorem 3. We will encounter (2.7.6) and (2.7.7) later, in a slightly different form, when we discuss the photon processing emission imaging system model. There, the intensity function $\lambda$ will be related to an underlying object $f$ through the application of a linear operator, i.e. we will have $\lambda \propto Hf$, and then the formula (2.7.6) is called the list-mode likelihood [23].

In Section 2.7.3 we will discuss methods of estimating a parameter such as $\lambda$ using the likelihood function; one such method is the so-called maximum likelihood estimator, which seeks to maximize the function $L(\theta)$ (or equivalently $\ell(\theta)$) over the set of admissible parameters $\Theta$. Immediately it is apparent that (2.7.7) is unbounded above: one can simply concentrate $\lambda(\cdot)$ on any of the sample points $a_j$ to make $\ell(\lambda)$ as large as desired. Whether or not this leads to a desired estimator is a matter of the task at hand: this may be a perfectly acceptable choice in some cases, while in other cases it may be desired to have a smoother $\lambda$.

### 2.7.3 Estimation Theory and Asymptotic Properties

Suppose we have a statistical dataset $g \in \mathcal{Y}$, which we assume to be sampled from a probability measure $\mathbb{P}_g = \mathbb{P}_g(\theta) \in \mathcal{M}$ for some $\theta \in \Theta$. The parameter set $\Theta$ is assumed to be a subset of some vector space $\mathcal{X}$, for instance a finite dimensional Euclidean space or an infinite dimensional space such as $L^1(V)$ or $\mathcal{S}'$. Given a reference measure $\mathbb{P}_1$ on $\mathcal{Y}$ (typically the Lebesgue measure or counting measure if $\mathcal{Y} \subset \mathbb{R}^n$, or an appropriate measure for functional data as discussed above),
we can (if $P_g(\theta)$ is absolutely continuous with respect to $P_1$) define a likelihood function

$$L(\theta) = p(g|\theta) = \frac{dP_g(\theta)}{dP_1}.$$ 

For each $\theta \in \Theta$, $p(g|\theta)$ is a probability density, even if $\mathcal{Y}$ is infinite dimensional (assuming again absolute continuity, which can be difficult in infinite dimensions). The goal of point estimation is to compute a particular $\hat{\theta} \in \Theta$ that might, in some sense, be a ‘likely’ value that could have given rise to the sampled data $g$. Any method which maps $g \in \mathcal{Y}$ to $\theta \in \Theta$ is called an estimation procedure, and the selected $\hat{\theta} = \mathcal{E}(g)$ is called an estimator. Because the data $g$ is random, and $\hat{\theta} = \mathcal{E}(g)$, an estimator is a generalized random vector; the distribution of $\hat{\theta}$ is a probability distribution $P_{\hat{\theta}}$ on $\Theta$ and is called the sampling distribution of the estimator.

We say that a particular $\theta \in \Theta$ is estimable with respect to the model $\mathcal{M}$ if $P(\theta') = P(\theta)$ implies that $\theta = \theta'$. In other words, if at least two distinct $\theta$ give rise to the same data distribution, those parameters are not estimable. An example that we discuss in Section 3.6 that is relevant to image science and $\mathcal{M}$-PMED is when an imaging system is described by a linear operator $\mathcal{H}$, in the sense that the distribution of imaging data $g$ is parameterized by $\mathcal{H}f$; if the operator $\mathcal{H}$ has a nontrivial null space, the objects $f$ are not estimable since $P_g(\mathcal{H}f) = P_g(\mathcal{H}(f + f_n))$ where $f_n$ is any object such that $\mathcal{H}f_n = 0$.

A very obvious (perhaps the most obvious) idea to compute a point estimator $\hat{\theta} \in \Theta$ given a realization of $g$ is to simply maximize the likelihood function over $\Theta$:

$$\hat{\theta}_{ML} = \arg \max_{\theta \in \Theta} L(\theta) \quad (2.7.8)$$

If the solution to (2.7.8) exists, we say that $\theta^*_{ML}$ is the Maximum Likelihood Estimator or MLE of $\theta$. When we are in the classical parametric estimation, finite-dimensional data setting, the MLE has many desirable properties that make it very popular. For the following statements, we assume that $\Theta \subset \mathbb{R}^p$ is a compact set and that $L(\theta) = p(g|\theta)$ is a PDF on $\mathbb{R}^n$ which is continuous with respect to $\theta$ for each fixed $g \in \mathbb{R}^n$. Then, in this case,

1. The MLE (2.7.8) always exists, by the extreme value theorem (recall we assumed that $\Theta$ is
compact and $\mathcal{L}(\theta)$ is continuous);

2. If an unbiased estimator of $\theta$ exists, the MLE is unbiased, meaning that its expected value over all realizations of the data gives the true parameter, \\
\[
\langle \hat{\theta}_{ML} \rangle_{g|\theta} = \theta
\]
In general, the MLE may only be asymptotically unbiased \[174]\;

3. The sampling distribution of the MLE is asymptotically normal, meaning that if $\hat{\theta}_n$ is the MLE of $\theta$ for $g \in \mathbb{R}^n$, then \\
\[
\sqrt{n}(\hat{\theta}_n - \theta) \xrightarrow{n \to \infty} N(0, F(\theta)^{-1})
\]
where $F(\theta)$ is the (asymptotic) Fisher information matrix, defined as \\
\[
F_{ij}(\theta) = -\langle \frac{\partial^2 \ell(\theta)}{\partial \theta_i \partial \theta_j} \rangle_{g|\theta},
\]
and the convergence is in distribution. This theorem, which requires some additional regularity assumptions on the likelihood, is discussed in e.g. \[174] \[277].

4. The MLE is asymptotically efficient, meaning (in the standard case) that as the number of samples $n \to \infty$, the MLE attains the Cramer-Rao lower bound:
\[
\text{Var}(\hat{\theta}_{ML}(j)) \to (F(\theta)^{-1})_{jj}
\]
Similar results are possible for other asymptotic scenarios besides $n \to \infty$.

5. The MLE is equivariant, meaning that if $q = M(\theta)$, then the MLE of $q$ is given by $M(\theta_{ML}^*)$. Note that this is true even if the mapping from $\theta$ to $q$ is not invertible; we define MLE in

117
this case via the profile likelihood \[ \mathcal{L}_M(q) = \sup_{\{\theta: M(\theta) = q\}} \mathcal{L}(\theta) \] i.e.

see also \[209\].

In general, if we remove the assumptions given above, the existence of a solution to the optimization problem \( (2.7.8) \) is by no means guaranteed: if \( \Theta \) is infinite-dimensional, as it can be in imaging, \( (2.7.8) \) is a problem in the calculus of variations, that is, the maximization of a functional over a subset of an infinite-dimensional space. If \( \Theta \) is a compact set and \( \mathcal{L} \) is lower semicontinuous on \( \theta \), then a solution does exist (this is an infinite-dimensional generalization of the extreme value theorem, see \[36\]), but otherwise it may be the case that no solution to \( (2.7.8) \) exists.

To illustrate the problems that arise when \( \Theta \) is a non-compact subset of an infinite-dimensional space, recall the definition of the (log) likelihood for a Poisson point process, \( (2.7.7) \). Suppose that 
\( \mathcal{A} \subset \mathbb{R}^k \) is a \( k \)-dimensional attribute set. As we mentioned, if one considers \( \Theta \) to be the convex cone of all \( \lambda \in L^1(A) \) such that \( \lambda(a) \geq 0 \) for all \( a \in A \), then no maximizer of \( \ell(\lambda) \) exists over \( \Theta \): we can take \( \lambda(a) \) to be a sum of approximate delta functions:

\[ \lambda_\epsilon(a) = \sum_{j=1}^{n} \phi_\epsilon(a - a_j) \] (2.7.9)

where \( \phi_\epsilon \) is a (positive) approximate delta function i.e. \( \lim_{\epsilon \to 0} \phi_\epsilon = \delta(\cdot) \). For example,

\[ \phi_\epsilon(a) = \frac{1}{(2\pi\epsilon)^{k/2}} \exp\left( -\frac{1}{2\epsilon} a^T a \right) \]

Then, for all \( \epsilon > 0 \) we have \( \int_{\mathcal{A}} \lambda_\epsilon(a) \, da = n \), but as \( \epsilon \to 0, \phi_\epsilon(0) \to \infty \) and hence \( \ell(\lambda) \to \infty \). Hence the ‘maximum likelihood estimator’ \(10\) for a PPP is simply the realization itself:

\[ \lambda_{ML}^*(a) = \sum_{j=1}^{n} \delta(a - a_j) \] (2.7.10)

This solution is actually to be expected: \( (2.7.10) \) is effectively on the ‘boundary’ of \( \Theta \). This

\footnote{technically it isn’t, because the resulting \( \lambda \) is not an \( L^1 \) function, and hence not an element of \( \Theta \).}
outcome can also be described as a kind of ‘over-fitting’, because we have effectively provided too many parameters to fit to our data. If the solution (2.7.10) is a problem (it may not be, depending on your purposes), there are three possible solutions:

1. One can expand or restrict the set $\Theta$ so that a more desirable solution to (2.7.8) is guaranteed to exist;

2. One can work with a Bayesian solution to the problem, which keeps the original $\Theta$ but defines a prior measure $\mathbb{P}_0$ over it, which emphasizes that certain solutions are ‘more desirable’ than others. In the Bayesian setting, we think of the posterior generalized random vector $\theta|g$ as the full solution to the inference problem. If desired, a point estimator can be selected; common choices are the posterior mean $\langle \theta|g \rangle$ and the maximizer of the posterior PDF (assuming it to be appropriately defined [266]):

   $$\theta_{MAP}^* = \arg \max_{\theta \in \Theta} p(\theta|g).$$

3. One can work only with approximate MLEs, that is, for a given $\epsilon > 0$, we can search for some $\theta_\epsilon \in \Theta$ such that

   $$\mathcal{L}(\theta_\epsilon) \geq \mathcal{L}(\theta) - \epsilon \quad \forall \theta \in \Theta$$

   These type of solutions are studied in e.g. [300].

All of the above suggested modifications can be called regularizations of the maximum likelihood estimator (2.7.8), because they produce solutions which may posses desired degree of smoothness (regularity). It can be shown that both the restricted domain and Bayesian MAP approaches lead to well-known deterministic regularization procedures such as Tikhonov regularization, depending on the choice of $\Theta$ or prior $\mathbb{Q}_0$ - see e.g. [266] or [150]. Due to their complexity, we will only briefly discuss Bayesian methods in this dissertation; there exist several very extensive surveys [266] [73] [150], and the investigation of such methods is a subject of future work.
2.7.4 The EM Algorithm

The EM algorithm, first presented in 1977 by Dempster, Laird and Rubin [78] is a general optimization method that is applicable for statistical estimation problems where there may be missing data. To discuss the algorithm, we assume first that we are considering a parametric estimation problem with classical data (i.e. both $\mathcal{Y}$ and $\Theta$ are finite-dimensional). The data $g$ is drawn from $p_g(g|\theta)$; the log-likelihood is defined, as usual, as $\ell(\theta) = \log(p_g(g|\theta))$ where $g$ is the observed data. Assume furthermore that $g$ is indirect or incomplete in the following sense: there exists a different random vector $g' \in \mathcal{Y}'$ called the complete data, and a mapping $g = \phi(g')$ that expressed the measured data in terms of the complete data. A simple example is when we observe binned counts instead of the raw data: the raw data is the ‘complete’ data, while the binned counts are ‘incomplete’, and the mapping $\phi$ represents the binning. The complete data are sampled from a PDF $p_c(g'|\theta)$ (we assume the same parameter vector specifies $p_c$ as specifies $p_g$). The complete-data likelihood is the function

$$\ell_c(\theta) = \log(p_c(g'|\theta))$$

Because we have assumed that $g = \phi(g')$, we have the following relationship between $p_g(g)$ and $p_g'(g')$:

$$p_g(g) = \int_{\{g':\phi(g')=g\}} p_g'(g') \, dg'$$

This is a consequence of the general change-of-variables theorem.

The EM algorithm is an iterative algorithm for computing the maximum likelihood estimate of $\theta$, using only the incomplete data. The algorithm alternates between computing the conditional expected value of the complete likelihood over the unobserved data, given the current parameter estimate, then subsequently maximizing the resulting function over the parameter set $\Theta$. Mathematically, we have the following iterative algorithm [78][192]:

1. Set an initial estimate $\theta = \theta_0$
2. Compute the conditional expected value of $\ell_c(\theta)$ over the unobserved data, using the current value of $\theta$ as the parameter (and keeping the observed data $g$ fixed):

$$Q(\theta, \theta_k) = \langle \ell_c(\theta) \rangle_{g'|g, \theta_k}$$

(2.7.11)

3. Maximize the resulting function of $\theta$ over $\Theta$:

$$\theta_{k+1} = \arg\max_{\theta \in \Theta} Q(\theta, \theta_k)$$

(2.7.12)

4. Iterate steps 2. and 3. until a convergence criteria on $\theta_k$ is satisfied.

In many cases the function $Q(\theta, \theta_k)$ can be computed explicitly, so that the maximization problem (2.7.12) is easier to compute. Generalizations of the EM algorithm for the case of infinite dimensional parameter set $\Theta$ are considered in e.g. [234].
Chapter 3

Theory and Simulation of Emission Imaging Systems

In order to make a data-informed decision about a patient, we must first collect data about the patient. Preferably, this data should contain as much information as possible pertaining to the task at hand: for our purposes, we desire to gain information about spatiotemporally inhomogeneous physiological processes that are occurring within a patient such as cell proliferation and drug susceptibility. As discussed in Chapter 4, knowledge of such processes is crucial for the prediction of treatment outcomes in cancer therapy. Thus, our data must be spatiotemporal in nature. The only practical method to gain true spatiotemporal information about internal anatomical and physiological features in vivo is via imaging techniques.

As we discussed in Section 1.3, the objective of imaging is to manipulate some form of energy to probe the physical structure of an object. While visible light imaging is the most widely known to the general public, imaging as we mean it is much more general: spatiotemporal information can be extracted using nearly the entire electromagnetic spectrum, from radio waves to gamma rays, but also using heat, sound and nuclear decay products such as alpha and beta particles. With in vivo clinical and preclinical imaging, different types of energy can probe anatomical structures such as bone, water and fat. By introducing pharmacological substances into the subject, it is also possible to image a broad range of physiological processes, hence providing functional information.
in addition to anatomical information. A basic example of the type of spatiotemporal functional information which is available via imaging is glucose metabolism: a glucose surrogate called fluorodeoxyglucose (FDG) can be ‘tagged’ with a radioactive $^{18}$F fluorine atom, producing a chemical $^{18}$F-FDG which emits radiation. By administering this drug to a patient or animal subject, then detecting the emitted radiation and performing a reconstruction step, we can produce an estimate of spatiotemporal glucose uptake in the subject. The broad class of functional imaging techniques which seek to image biomolecular processes occurring within a subject has been coined molecular imaging; $^{18}$F-FDG is perhaps the most widely used in vivo molecular imaging technique, but many other similar techniques are now available in both the lab and the clinic.

In this chapter, we discuss the mathematical modeling of a particular class of molecular imaging procedure, based on the emission of high energy radiation by a tagged (or ‘labeled’) drug. This class of imaging methods includes planar nuclear imaging, PET and SPECT techniques, as well as a class of more recent methods based on the emission of light in the visible band due to stimulated fluorescence or bioluminescence. With some relatively minor modifications, the models presented here can also be applied to the imaging of nuclear decay products (alpha and beta particles). While we do not discuss optical tomography or X-Ray tomography, the mathematical modeling for these techniques is related.

We have already mentioned that imaging techniques can either be anatomical or physiological, but techniques can also be broadly classified as planar (if only two-dimensional information is available) or tomographic, which refers to any technique which seeks to gain true three-dimensional spatial information. Imaging can also be dynamic (if time information is available) or static (the image produced is a time snapshot or average). Imaging techniques can also be quantitative or non-quantitative, depending whether the method seeks to accurately quantify some patient parameter as opposed to some qualitative feature of the patient. This last distinction is somewhat vague, because regardless of whether a particular technique is capable of producing quantitative information, whether or not an image dataset is used quantitatively depends on the task at hand.

\[1\text{ tomos and graphic are derived from the ancient Greek for ‘slice’ and ‘image’ \cite{26}\]
3.1 Overview of Emission Imaging

Emission imaging concerns making images of the behavior of drugs in a body, and the biochemical fact that enables emission imaging is the tracer principle, which in its classical form is stated as:

*If any atom in any chemical’s molecular structure is changed to a radioactive isotope of the same atom, the biological function of the chemical is unchanged.*

More generally, it may not be possible to swap an atom directly for a radioactive isotope – it may be necessary to wholly modify the structure of the molecule in order to tag it. A more general tracer principle could be stated as:

*Many molecules can be modified to become radioactive or fluorescent, in such a way that the biological function of the molecule is largely unchanged.*

We will treat the modeling of emission imaging systems in two stages. First, we model the emission and propagation of radiation in the body: this is an exercise in the physics of energy transport, as modeled by the Radiative Transport Equation (RTE). We then discuss modern high-energy photodetectors and the resulting stochastic models of photodetection. Combining the propagation and detection models, we obtain a complete deterministic and statistical model of ECT systems. A more complete discussion of emission imaging might also include mathematical models of the behavior of drugs in a body (so-called pharmacokinetic and pharmacodynamic modeling), but we will simply assume the presence of a drug concentration \( c \equiv c(r,t) \) which has been tagged to produce some form of radiation; we discuss pharmacokinetics briefly in Chapter 4. For now, we assume only that \( c \sim P_c \) is a generalized random vector, where a patient \( j \) gives rise to a realization \( c_j = c(r,t,\omega_j) \in X \). Note that the true distribution \( P_c \) may be unknown; one can select a prior model for it, for instance by assuming that \( P_c = \mathcal{LN}(\mu,C) \) or one of the other random processes presented in Section 2.6, but it is not always necessary to do so.

In terms of the complete parameterization \( f \) of our patient, we assume the drug concentration \( c \) corresponds directly to one of the components of \( f \), i.e. \( c = f^{(k)} \) for some \( k \). In this sense, we
are treating the drug concentration itself as a physiological random process, though in many cases the drug is intended to target and hence image a true physiological process: for instance, with $^{18}$F-FDG, the physiological process that we are interested in is glucose metabolism, but we can only image the distribution of a drug $c(r,t)$ which targets glucose metabolism, so in a sense, we are not imaging the process itself: some portion of $c(r,t)$ may be in the vascular or interstitial space, thus not corresponding directly to cellular metabolism, but the reconstructed image will usually not make this distinction. This is related to the so-called concentration decomposition problem that we discuss briefly in Chapter 4 and Section 5.2.

For now, we focus on a single tagged drug concentration $c(r,t)$ and a single imaging system designed to measure the radiation emitted by $c(r,t)$. One of the goals of this chapter is to discuss how the imaging system gives rise to a precise decomposition of $c$ into measurement and null components. For the full object $f$, we may consider additional imaging systems designed to image other drug concentrations, which are in turn related to other components of $f$, and each imaging system will give rise to a measurement component $f_m^{(k)}$ and null component $f_n^{(k)}$. The full null component of the object, $f_n$, will contain, in addition to the null components of the various imaging systems, any other anatomical and physiological information that does not contribute to the measured image data for any available imaging system. For example, if no X-Ray CT scan is performed, the patient’s X-Ray attenuation map must be treated as a component of $f_n$, hence unknown and random.

The objective of molecular imaging is to gain information about the drug distribution $c_j \in \mathcal{X}$, but because the drug is inside a body, we do not have direct access to it: however, because we have assumed that $c_j$ has been tagged with an emissive label, we can detect the radiation that it emits. We will show that this implies that we have access to a sample from a generalized random vector $g|c_j \in \mathcal{Y}$, which is related to $c_j$ via a conditional distribution, i.e. $g|c_j \sim \mathbb{P}_g|c_j$. In other words, we collect indirect, randomly sampled data $g$ which will allow us to estimate properties of $c_j$. We will show that the mapping from $c_j$ to the average data $\langle g|c_j \rangle$ will give rise to a linear operator $\mathcal{H}$.
between spaces \( \mathcal{X} \) and \( \mathcal{Y} \) via the relation

\[
\mathcal{H}c_j = \langle g | c_j \rangle.
\]  

(3.1.1)

This operator is called the *system model*, and the mean \( \langle g | c_j \rangle \) will serve as a parameter in the distribution of \( g | c_j \); in fact, under the assumptions we make later, we will show that \( g | c_j \) will either be a Poisson random vector in \( \mathbb{R}^M \) or Poisson point process on a certain attribute set \( \mathcal{A} \), either of which is fully described by its mean, which is given by (3.1.1). Thus, we will show that

\[
g | c_j \sim \text{Poi}(\mathcal{H}c_j),
\]

where \( \text{Poi}(\cdot) \) stands for either a Poisson random vector or Poisson point process. Overall, because \( c_j \) is a realization of a random process, the (unconditioned) random data \( g \) is a doubly stochastic Poisson random vector or a doubly stochastic Poisson point process. The full statistical model for raw ECT imaging data can be written abstractly using a general form of the law of total probability [52]:

\[
P_g = \int_{\mathcal{X}} \text{Poi}(\mathcal{H}c)d\mathbb{P}_c
\]  

(3.1.2)

where again \( \text{Poi}(\cdot) \) indicates either a Poisson random vector in \( \mathbb{R}^M \) or a Poisson point process, and \( c \sim \mathbb{P}_c \) is a generalized random vector. Given data \( g | c_j \), to produce an estimate of \( c_j \), or more generally a patient-specific virtual ensemble \( \tilde{c}_j \), we must then employ one of the statistical procedures outlined in Section 2.7 – we treat \( c_j \) as an infinite dimensional parameter in a statistical model \( \mathcal{M} \), where the data is given by \( g | c_j \) and the model \( \mathcal{M} \) is given by \( \text{Poi}(\mathcal{H}c_j) \), with \( c_j \in \Theta = \mathcal{X} \) (or some appropriate subset of \( \mathcal{X} \)). Again, both maximum likelihood and Bayesian approaches to constructing \( \tilde{c}_j \) are appropriate. Evaluating the merits and trade-offs of different methods of constructing \( \tilde{c}_j \) will depend strongly on the task, which in our case is prediction of treatment outcomes; see Chapter 4.

The system operator \( \mathcal{H} \) will be further decomposed into an *energy propagation* component
denoted $\mathcal{P}$ and a detection component denoted $\mathcal{D}$, so that $\mathcal{H} = \mathcal{D}\mathcal{P}$. The propagation operator will arise as the solution of an RTE, while the detection operator will relate to the geometry of the system and the detector type; we consider two detector types, namely *photon-counting* detectors which count photointeractions which take place within discrete pixels, and *photon-processing* detectors which seek to estimate attributes of the individual photointeraction events.

The goal of this chapter is to discuss both the deterministic properties of ECT imaging systems (as modeled by the linear operator $\mathcal{H}$) and the statistical properties of ECT imaging systems (as modeled by Poisson statistics); understanding both is essential to quantifying uncertainties in $\mathcal{M}$-PMED – the deterministic properties of $\mathcal{H}$ will dictate which properties of the tagged drug concentration $c_j$ are estimable from $g|c_j$, explicitly giving the decomposition $f^{(k)} = [f^{(k)}_m, f^{(k)}_n]$ discussed above, while the stochastic properties will inspire methods to construct the component of the virtual patient ensemble $\tilde{f}^{(k)}_m$ and allow us to understand its statistical properties, and hence ultimately the statistical properties of $\tilde{q} = \mathcal{M}(\tilde{f})$, which will allow us to compute patient-specific probabilities. Again, only the measurement component of the measured drug concentration will be estimable from $g|c_j$; any other physiological processes must be treated as unknown and hence random, until patient-specific data relating to them is available.

### 3.2 Physics of Radiation and the Radiative Transport Equation

As discussed above, we assume that a radiolabeled drug concentration, denoted by $c = c(r, t)$, is a source of detectable energy, which we assume to be in the form of high-energy electromagnetic radiation; other forms of emissive energy, such as alpha and beta particles, can also be considered, and the modeling is similar [80]. For instance, SPECT tracers which have been tagged with $^{99m}$Tc emit gamma rays with an energy of 140 keV [293]; these rays will subsequently propagate in the body, then travel towards an imaging detector. So, in order to understand the relationship between detected energy and the tagged drug concentration, we must understand how high energy light propagates in a medium.

The behavior of all electromagnetic fields is fully described by Maxwell’s equations [26], which provide a complete description of the wave properties of light in the classical vector field picture.
One can also consider light to be a quantum mechanical state, which leads to quantum electrodynamics \[26\] \[186\]. However, for our purposes, we are considering light that has a very short wavelength in relation to the other relevant dimensions of the system, so instead of either the classical wave field or quantum picture, we think instead in terms of a *ray* picture of light: light propagates as a discrete quantity along rays, and we can treat the propagation of energy directly using the phenomenological laws of geometrical optics and radiometry. We will describe radiation as follows: for each position \( r \in \mathbb{R}^3 \) and *direction* \( \hat{s} \in S^2 \) (where \( S^2 \) is the unit sphere), we can describe an amount of light traveling in the direction \( \hat{s} \) from \( r \) with a given photon energy \( E \). This can be described entirely by a function \( w = w(r, \hat{s}, E, t) \), which is a function of six variables and time: three position variables, three momentum variables and time, where the momentum variable is decomposed into a direction \( \hat{s} \) and an energy \( E \). The function \( w \) is completely analogous to a classical phase-space density from the kinetic theory of gases and particle transport - in a certain sense, we are working within a kinetic regime for light \[83\] §1.4 \[14\]. The function \( w \) can be used to compute a flux of electromagnetic particles – which we loosely call photons – in a region of space. When we discuss photodetection, \( w \) will be directly proportional to an intensity function \( \lambda(a) \) for a certain Poisson point process describing the interaction of photons with a detector. When dimensions of the physical apparatus are much larger than the wavelength of the light – as is the case in emission imaging – this description of the radiation is sufficiently accurate.

While the detection of electromagnetic radiation is in earnest a quantum process, we will make some basic assumptions about photodetectors that lead directly to a Poisson statistical model without any discussion of quantum mechanics. We refer to e.g. \[26\] \[246\] \[186\] \[96\] for a more complete discussion of the quantum nature of light detection in both the semi-classical and fully quantum descriptions.

### 3.2.1 Radiometry

Starting in section 3.2.2 we will derive an equation for the propagation of the phase-space density \( w \) introduced above. With some assumptions, is possible to derive this model rigorously from first principles (i.e. from Maxwell’s equations), in certain asymptotic regimes. Such derivations are
discussed for example in Section 5.7 in Mandel and Wolf [186], the papers by Ryzhik et al. [244], Bal [14], Caze and Schotland [47], and Wolf [299].

To briefly describe the connection between the classical wave picture and the radiometric picture, consider a scalar wavefield \( u(r,t) \) (for instance, a component of the vector electric or magnetic field). Assuming that \( u(r,t) \) is a temporally stationary and ergodic random field, one can consider its \textit{mutual coherence function} \( \Gamma \) and its temporal Fourier transform, called the \textit{cross spectral density} function \( W \):

\[
\Gamma(r_1, r_2, \tau) = \langle u(r_1, t)u^*(r_2, t + \tau) \rangle \\
W(r_1, r_2, \nu) = \int_{-\infty}^{\infty} \Gamma(r_1, r_2, \tau) \exp(-2\pi i \nu \tau) d\tau
\]

Then, define the \textit{generalized radiance} for \( u \) as the spatial Wigner transform of \( W \):

\[
L^P_\nu(r, \hat{s}, \nu) = \frac{\cos \theta}{\lambda^2} \int_P W \left( r + \frac{1}{2} r', r - \frac{1}{2} r', \nu \right) \exp(-i\hat{s} \cdot r') d^2 r'
\]  

(3.2.1)

where \( P \) is a 2D plane and \( \cos \theta = \hat{s} \cdot \hat{n} \) is the (cosine of the) angle between \( \hat{s} \) and the unit normal to \( P \). Then, under certain physical circumstances (for instance, when the source is ‘quasi-homogeneous’ [186]), the function (3.2.1) satisfies the same properties as the classical radiance function. It can then be shown that as the wavenumber \( k \to \infty \) (equivalently \( \lambda \to 0 \)), the function \( L^P_\nu(r, \hat{s}, \nu) \) satisfies a radiative transfer equation. In particular, if one assumes further that \( L^P_\nu \) is propagating in a random medium, the scattering operator is related to the statistical properties of the random media [244] [139].

An alternative to the rigorous derivation of radiative transfer from electromagnetic wave theory is to proceed via classical radiometry, which is a phenomenological theory of light detection; see [26] [186] and references for a complete discussion. For our purposes, we only require the definition of the \textit{spectral photon radiance} \( L_{p,E}(r, \hat{s}, E, t) \) and the phase-space distribution function \( w(r, \hat{s}, E, t) \) which we have already introduced. These quantities are connected via the speed of light in the medium, which we assume is constant (a very accurate assumption for gamma radiation), as follows
\[ L_{p,E}(r, \hat{s}, E, t) = c_m w(r, \hat{s}, E, t) \]

The units of \( L_{p,E} \) are photons (a unitless number) per steradian per unit (projected) area per unit energy per second i.e. \([L_{p,E}] = m^{-2} \text{ster}^{-1} \text{eV}^{-1} \text{sec}^{-1}\); the units of \( w \) are photons per unit volume per steradian per unit energy, i.e. \([w] = m^{-3} \text{ster}^{-1} \text{eV}^{-1}\). If an oriented surface \( D \subset \mathbb{R}^3 \) with normal \( \hat{n}(r) \) is present and we wish to compute the flux of photons with energy in \( \Delta E \) through this surface during a time \( \Delta t \), we can compute the surface integral:

\[
\Phi_p = \Delta t \Delta E \int_D \int_{S^2_+ (r)} L_{p,E}(r, \hat{s}, E, t) \hat{s} \cdot \hat{n}(r) \, d\hat{s} \, dr = c_m \Delta t \Delta E \int_D \int_{S^2_+ (r)} w(r, \hat{s}, E, t) \hat{s} \cdot \hat{n}(r) \, d\hat{s} \, dr
\]

where \( S^2_+ \) is the hemisphere of directions on the same side of the surface as \( \hat{n}(r) \).

The phase space distribution function \( w \) satisfies an energy transport equation, which we will now derive; the solution of this equation will lead to an explicit description of the propagation operator \( \mathcal{P} \) for an ECT system.

### 3.2.2 The Phase Space Distribution and Radiative Transport Equation

The Radiative Transport Equation (RTE) is an integrodifferential equation that models the propagation of energy, as described by a phase-space density function \( w \), in high energy photon-based imaging systems. Its validity as a mathematical model is well accepted in this context \([26, 285, 8]\). As discussed above, this modeling regime can be derived rigorously as an asymptotic limit of wave optics in phase space, or can be considered as a phenomenological law. All of the emission imaging modalities discussed in Section 3.1 concern the emission and propagation of high-energy light in tissue, and some form of radiative energy transfer model is applicable. Transport equations similar to the RTE discussed here are also applicable to optical tomography \([8]\) and other particle-based emission imaging techniques such as alpha and beta particle imaging \([80]\), but we will not discuss these techniques here.

We begin by defining \textit{optical phase space}. Let \( V \subset \mathbb{R}^3 \) be a compact, convex subset of \( \mathbb{R}^3 \) with
piecewise smooth boundary (for instance, a box, sphere, or cylinder), and let $S^2$ denote the sphere of unit length directions. Then, optical phase space is defined as the Cartesian product

$$\Gamma := V \times S^2.$$ 

An element of $\Gamma$ will be denoted by $\gamma = (r, \hat{s})$, where $r \in V$ and $\hat{s} \in S^2$. These represent respectively a position and a direction; if photon energy is required, we will denote it with $E$, so that $(r, \hat{s}, E)$ defines an optical ‘ray’ with origin $r \in V$, direction $\hat{s} \in S^2$, and photon energy $E$.

We will think of the spatial domain $V$ as the support set of an imaging system, like that shown in Figure 3.6. We also define two special subsets of $\Gamma$, called the inflow and outflow boundaries:

$$\partial_{\pm} \Gamma = \{(r, \hat{s}) \in \Gamma | r \in \partial V, \pm \hat{s} \cdot \hat{n}(r) > 0\}$$

The vector $\hat{n}(r)$ is the outward pointing unit normal vector on the boundary of the field of view at position $r$. The sets $\Gamma$ and $\partial_{\pm} \Gamma$ are illustrated in Figure 3.1. It should be clear from that figure that $\partial_+ \Gamma$ is exactly the subset of positions and directions that are available for detection; positions inside the domain are assumed to not be directly accessible, and directions traveling ‘in’ are also not accessible. We will see later that a particular arrangement of detectors and collimators defines a subset of the outflow boundary, $A \subset \partial_+ \Gamma$, called the visible boundary (terminology due to [131]).

The function spaces $L^2(V)$, $L^2(\Gamma)$ and $L^2(\partial_{\pm} \Gamma)$ are defined as usual, with the inner products being defined as:

$$(u, v)_{L^2(V)} = \int_V u(r)v(r) \, dr \quad (3.2.2)$$

$$(w, v)_{L^2(\Gamma)} = \int_{\Gamma} w(\gamma)v(\gamma) \, d\gamma = \int_{S^2} \int_V w(r, \hat{s})v(r, \hat{s}) \, drd\hat{s} \quad (3.2.3)$$

$$(w, v)_{L^2(\partial_{\pm} \Gamma)} = \int_{\partial_{\pm}\Gamma} w(\gamma)v(\gamma) \, d\xi(\gamma) = \int_{S^2} \int_{\partial V} w(r, \hat{s})v(r, \hat{s}) \, |\hat{s} \cdot \hat{n}|drd\hat{s} \quad (3.2.4)$$

where we have defined the surface integral measure $d\xi(\gamma) = |\hat{s} \cdot \hat{n}|drd\hat{s}$. We will use (3.2.4) to compute an adjoint operator later on. Note that when we write $d\hat{s}$, this is shorthand for the
surface measure on the sphere; in spherical coordinates, for instance, we would have \( \hat{s} = \hat{s}(\theta, \varphi) \) and \( d\hat{s} = \sin(\theta) d\theta d\varphi \).

As discussed above, the function describing the distribution of photons is called the phase space distribution and is written \( w(r, \hat{s}, \mathcal{E}, t) \) or \( w(r, \hat{s}, \mathcal{E}) \) (in the steady state case). In other words, \( w \) is a function of a phase space parameter \( \gamma \), a photon energy \( \mathcal{E} \) and a time \( t \), if necessary. The physical interpretation of \( w \) is that it is a photon number density. Thinking of photons as classical particles (that is, objects with well-defined position and momentum), \( w(r, \hat{s}, \mathcal{E}, t) \) is a density such that

\[
\int_{\mathcal{E}}^{\mathcal{E} + \Delta \mathcal{E}} \int_{S} \int_{V} w(r, \hat{s}, \mathcal{E}, t) \, d^3r d\hat{s} d\mathcal{E}
\]

gives the expected number of photons with position \( r \in V \), traveling in directions \( \hat{s} \in S \), with energies \( \mathcal{E} \in [\mathcal{E}, \mathcal{E} + \Delta \mathcal{E}] \) at time \( t \). Appropriately normalized, \( w \) is a probability density function: a collection of \( N \) 'photons' can be defined by sampling \( (r_1(t), \hat{s}_1(t), \mathcal{E}_1(t)), \ldots, (r_N(t), \hat{s}_N(t), \mathcal{E}_N(t)) \) I.I.D. from \( \tilde{w} = \frac{w}{\bar{w}} \). As discussed in the previous section, \( w \) is also related to the spectral photon radiance \( L_{p,\mathcal{E}} \) via \( w = c_m L_{p,\mathcal{E}} \).

Because electromagnetic energy propagates, we expect the function \( w(r, \hat{s}, \mathcal{E}, t) \) to satisfy a
propagation equation, and indeed it does: the equation that \( w \) satisfies is the RTE. To derive the RTE, we follow a rigorous phenomenological derivation similar to that outlined in [83]. This derivation differs slightly from that in [26] and [49], only in that we use exact integral quantities instead of ‘delta’ approximations. We will assume throughout that the speed of light in the medium is a constant \( c_m > 0 \), which is a realistic assumption for X-Ray and gamma-ray radiation; assuming a non-constant light speed leads to transport equations in non-euclidean geometries where rays are curved instead of straight - an interesting digression that is discussed extensively elsewhere [15] [171] [202] [92].

Consider a fixed subset of optical phase space \( \Sigma \times E \subset \Gamma \times [0, \infty) \) and define

\[
W(t) = \int_E \int_{\Sigma} w(\gamma, \mathcal{E}, t) \, d\gamma d\mathcal{E}
\]  

The basic ‘energy balance’ that we will follow is

\[
\frac{dW}{dt} = \left( \frac{dW}{dt} \right)_{\text{prop}} + \left( \frac{dW}{dt} \right)_{\text{coll}} + \left( \frac{dW}{dt} \right)_{\text{source}}
\]

where the three terms are, respectively, due to propagation, photon-material collisions (i.e. photons scattering off and absorbing into the medium), and sources. With \( W(t) \) defined as in (3.2.5), we have

\[
\frac{dW}{dt} = \int_E \int_{\Sigma} \frac{\partial w}{\partial t}(\gamma, \mathcal{E}, t) \, d\gamma d\mathcal{E}
\]

\[
= -c_m \int_E \int_{\partial_{+}\Sigma} w(\gamma, \mathcal{E}, t) \hat{s} \cdot d\hat{n} d\mathcal{E} + \int_E \int_{\Sigma} \left( \frac{\partial w}{\partial t} \right)_{\text{coll}} d\gamma d\mathcal{E} + \int_E \int_{\Sigma} \Xi(\gamma, \mathcal{E}, t) \, d\gamma d\mathcal{E}
\]

where \( c_m w(\gamma, \mathcal{E}, t) \hat{s} \cdot \hat{n} \) is the particle flux through the boundary of the spatial domain, \( (\cdot)_{\text{coll}} \) is a material collision term to be determined, and \( \Xi(\gamma, \mathcal{E}, t) \) is an arbitrary photon source. Using the divergence theorem, we can write

\[
- \int_E \int_{\partial_{+}\Sigma} w(\gamma, \mathcal{E}, t) \hat{s} \cdot d\hat{n} d\mathcal{E} = \int_E \int_{\Sigma} \nabla_{\mathcal{E}} \cdot (\hat{s} w(\gamma, \mathcal{E}, t)) \, d\gamma d\mathcal{E} = \int_E \int_{\Sigma} \hat{s} \cdot \nabla_{\mathcal{E}} w(\gamma, \mathcal{E}, t) \, d\gamma d\mathcal{E}
\]
where we used \( \nabla r \cdot (\hat{s} w) = w \nabla r \cdot \hat{s} + \hat{s} \cdot \nabla r w = \hat{s} \cdot \nabla r w \). We now have all the integrals in (3.2.7) over \( \Sigma \times E \), and so (3.2.7) becomes

\[
\int_E \int_\Sigma \left( \frac{\partial w}{\partial t}(\gamma, \mathcal{E}, t) + c_m \hat{s} \cdot \nabla_r w(\gamma, \mathcal{E}, t) - \left( \frac{\partial w}{\partial t} \right)_{\text{coll}} - \Xi(\gamma, \mathcal{E}, t) \right) d\gamma d\mathcal{E} = 0
\]

Since \( \Sigma \) was an arbitrary phase volume, the integrand must be identically zero. Hence we have

\[
\frac{\partial w}{\partial t}(\gamma, \mathcal{E}, t) + c_m \hat{s} \cdot \nabla_r w(\gamma, \mathcal{E}, t) = \left( \frac{\partial w}{\partial t} \right)_{\text{coll}} + \Xi(\gamma, \mathcal{E}, t)
\]

The derivation of the collision term is discussed in [26] and [285]. The result is that interactions between the radiance field and the propagation medium result in two types of behavior: energy can be absorbed by the atoms that make up the material, and rays can be elastically scattered. These processes can be expressed as [26]

\[
\left( \frac{\partial w}{\partial t} \right)_{\text{coll}} = -c_m \mu_{\text{tot}} w + \mathcal{K} w,
\]

where \( \mu_{\text{tot}} \) is the total attenuation coefficient, defined as the sum of the attenuation due to scatter and absorption:

\[
\mu_{\text{tot}} = \mu_{\text{sc}}(r, \hat{s}, \mathcal{E}, t) + \mu_{\text{abs}}(r, \hat{s}, \mathcal{E}, t)
\]

We assume temporarily that the attenuation can depend on position, direction, energy and time, but later (for technical reasons) we will assume that it only depends on position and energy. The operator \( \mathcal{K} \) is the scattering operator, which has the general form

\[
(\mathcal{K} w)(r, \hat{s}, \mathcal{E}, t) = \int_0^\infty \int_{S^2} k(r, \hat{s}, \mathcal{E}, \hat{s}', \mathcal{E}', t) w(r, \hat{s}', \mathcal{E}', t) d\hat{s}' d\mathcal{E}'
\]

The differential equation that \( w \) must satisfy thus finally takes the form

\[
\frac{\partial w}{\partial t} + c_m \hat{s} \cdot \nabla_r w = -c_m \mu_{\text{tot}} w + \mathcal{K} w + \Xi
\quad (3.2.8)
\]
The equation (3.2.8) is called the Radiative Transport Equation or RTE. It is also frequently called the (linear) Boltzmann transport equation after Ludwig Boltzmann [26]. The general Boltzmann equation models kinetics of particle transport when particles can collide with each other, which leads to a nonlinear equation in general.

In the steady-state regime, \( \frac{\partial w}{\partial t} \equiv 0 \) and so \( w \equiv w(r, \hat{s}, \mathcal{E}) \) must satisfy the stationary RTE:

\[
c_m \hat{s} \cdot \nabla_r w + c_m \mu_{\text{tot}} w - \mathcal{K} w = \Xi
\]  

(3.2.9)

We will assume that the attenuation is isotropic (i.e. \( \mu = \mu(r, \mathcal{E}) \)) to prevent technical issues regarding the null space of the solution operator to (3.2.9) (we will discuss null space more later).

The appropriate boundary condition for (3.2.9) in emission imaging is that \( w|_{\partial_{-\Gamma}} = 0 \), i.e. there is no incoming radiance field incident on the domain. In transmission imaging (such as X-Ray CT) or fluorescence imaging, this incoming term will be nonzero [8].

Writing the ‘transport’ operator as \( \mathcal{T}_\mu = c_m \hat{s} \cdot \nabla_r + c_m \mu_{\text{tot}} \), we can express the equation (3.2.9) in operator form as

\[
\begin{cases}
(T_\mu - \mathcal{K}) w = \Xi \\
 w|_{\partial_{-\Gamma}} = 0
\end{cases}
\]  

(3.2.10)

In emission imaging, the source \( \Xi \) is directly related to the drug concentration of interest \( c(r, t) \), because we assume \( c(r, t) \) to be tagged with a photon emitter. The goal of emission imaging is thus to recover \( \Xi \) from measurements. As we will see later, the measurements available will be related to the solution to (3.2.10), evaluated on the (outgoing) boundary \( \partial_+ \Gamma \). It stands to reason that if (3.2.10) has a solution \( w(r, \hat{s}, \mathcal{E}) \) that can be evaluated on the outgoing boundary (that is, for \( (r, \hat{s}) \in \partial_+ \Gamma \)), then the following operator will be of interest for emission imaging:

\[
\mathcal{P} \Xi = w|_{\partial_+ \Gamma}
\]  

(3.2.11)

where \( w \) is the solution to (3.2.10). The operator (3.2.11) will be called the propagation operator for emission imaging. One of the key features of any real emission imaging system, and where our
work departs from the classical studies of inverse transport problems (e.g. [26] [131] [17]), is that we will never have direct measurements of (3.2.11) (i.e. we cannot evaluate \( w|_{\partial_+ \Gamma} \) or even any restricted version of it) but rather a compact integral operator \( D : L^2(\partial_+ \Gamma) \to \mathcal{Y} \) will describe the (mean) measurement process, and Poisson statistics will describe the stochastic properties of the measurement process. We will discuss the measurement process and resulting operator models in Section 3.4, but first we discuss the solution to the equation (3.2.10), which provides an explicit form for the operator (3.2.11).

### 3.3 Solutions of the Radiative Transport Equation

To solve (3.2.10), we follow [26] and first consider the ‘ballistic’ case where the scatter \( \mathcal{K} \equiv 0 \), so that we are solving the equation \( T_\mu w = \Xi \) with the no-incident-flux boundary condition \( w|_{\partial_- \Gamma} = 0 \). To compute \( w(r, \hat{s}, \mathcal{E}) \), note that \( \hat{s} \cdot \nabla_r w \) is simply the directional derivative of \( w \) in the direction \( \hat{s} \). Then, define the single-variable function

\[
\tilde{w}(z) = w(r + (z - \tau_-)\hat{s}, \hat{s}, \mathcal{E}),
\]

where \( \tau_- \) is the reverse domain exit time, defined in Figure 3.2.

![Figure 3.2](image_url)

Figure 3.2: Definition of the forward and backward exit times \( \tau_\pm = \tau_\pm(r, \hat{s}) \). The forward exit time \( \tau_+ \) is the unique \( \tau \) such that \((r + \tau \hat{s}, \hat{s}) \in \partial_+ \Gamma \); \( \tau_- \) is the unique \( \tau \) such that \((r - \tau \hat{s}, -\hat{s}) \in \partial_+ \Gamma \). Uniqueness of \( \tau_\pm \) is guaranteed by the convexity of \( V \). Note that if \((r, \hat{s}) \in \partial_- \Gamma \), \( \tau_- = 0 \); if \((r, \hat{s}) \in \partial_+ \Gamma \), \( \tau_+ = 0 \).
Then, note that
\[ \frac{d\tilde{w}}{dz} = \hat{s} \cdot (\nabla_r w)(r + (z - \tau_-)\hat{s}, \hat{s}, \mathcal{E}), \]
and furthermore \( \tilde{w}(0) = w(r - \tau_- \hat{s}, \hat{s}, \mathcal{E}) = 0 \) and \( \tilde{w}(\tau_-) = w(r, \hat{s}, \mathcal{E}). \) Thus, defining similarly \( \tilde{\mu} \)
and \( \tilde{\Xi} \) via \( r \rightarrow r + (z - \tau_-)\hat{s}, \)
we consider the (single-variable) initial-value problem
\[
\begin{cases}
\frac{d\tilde{w}}{dz}(z) + \tilde{\mu}(z)\tilde{w}(z) = \frac{1}{c_m}\tilde{\Xi}(z) \\
\tilde{w}(0) = 0 
\end{cases}
\tag{3.3.1}
\]
By solving \((3.3.1)\) until \( z = \tau_- \), we will obtain \( w(r, \hat{s}, \mathcal{E}). \) Furthermore, the ODE \((3.3.1)\) will play a key role in a parallel numerical scheme for computing \( \mathcal{P} \) in the ballistic case, which we discuss in Section 3.7. Solving the ODE \((3.3.1)\) using an integrating factor and returning to the original coordinates shows that the full solution to \((3.2.10)\) for \( K = 0 \) is given by the attenuated X-Ray transform of the source density:
\[ w(r, \hat{s}, \mathcal{E}) = (\mathcal{X}_\mu \Xi)(r, \hat{s}, \mathcal{E}) = \frac{1}{c_m} \int_{0}^{\tau_-} \Xi(r - \ell \hat{s}, \hat{s}, \mathcal{E}) \exp \left( -\int_{0}^{\ell} \mu(r - \ell' \hat{s}, \mathcal{E}) \, d\ell' \right) \, d\ell \]  
\tag{3.3.2}
In operator notation, we have formally that \( \mathcal{X}_\mu = \mathcal{T}_\mu^{-1} \). We will frequently assume that the source function \( \Xi \) is isotropic and monoenergetic, so that it can be written as
\[ \Xi(r, \hat{s}, \mathcal{E}) = \frac{1}{4\pi} f(r) \delta(\mathcal{E} - \mathcal{E}_0) \]
Abusing notation and writing \( \mu(r) \equiv \mu(r, \mathcal{E}_0) \), we can in this case simplify \((3.3.2)\) to
\[ w(r, \hat{s}) = (\mathcal{X}_\mu f)(r, \hat{s}) = \frac{1}{4\pi c_m} \int_{0}^{\tau_-} f(r - \ell \hat{s}) \exp \left( -\int_{0}^{\ell} \mu(r - \ell' \hat{s}) \, d\ell' \right) \, d\ell \]  
\tag{3.3.3}
If we evaluate \((3.3.3)\) for only those \( (r, \hat{s}) \) in the outgoing boundary \( \partial_+ \Gamma \), we will arrive at the propagation operator \((3.2.11)\). More rigorously, in \([260]\), it is shown that for a generic set\(^2\) of

\(^{2}\)Generic means an ‘open and dense’ set, which means that every pair \((\mathbf{k}, \mathcal{E})\) has a neighborhood for which the same is true, and for every pair there is another pair arbitrarily close for which it is true.
smooth attenuation and scattering functions \((\mu, k)\), including a neighborhood of \((\mu, k) \equiv (0, 0)\), the solution operator to the full equation \(3.2.10\) extends as a map from \(L^2(V)\) to \(L^2(\partial_+ \Gamma)\). Thus we can, in the \(K \equiv 0\) case, rigorously define

\[
\mathcal{P} = \mathcal{X}_\mu|_{\partial_+ \Gamma} : L^2(V) \to L^2(\partial_+ \Gamma)
\]  

(3.3.4)

We will consider only the \(K \equiv 0\) case here, because this is a reasonable model for high-energy gamma-ray imaging; in biological tissues, gamma rays only undergo inelastic Compton scatter, which lowers their energy, and these lower-energy photons can subsequently be rejected by the detection hardware. For completeness we briefly discuss the case of nonzero scattering, because it is useful for lower-energy radiation.

If \(K \neq 0\), as is frequently the case in optical emission and charged particle modalities, we must return to the operator equation \(3.2.10\). Applying \(\mathcal{X}_\mu\) to both sides of \((\mathcal{T}_\mu - K)w = \Xi\) and using the fact that \(\mathcal{X}_\mu \mathcal{T}_\mu = \mathcal{I}\) (the identity operator) results in a Fredholm equation[26][232]:

\[
(\mathcal{I} - \mathcal{X}_\mu K)w = \mathcal{X}_\mu \Xi
\]

(3.3.5)

It is proved rigorously in [260], using Fredholm theory, that the equation \(3.3.5\) is solvable for an open dense set of \(\mu\) and \(K\). In imaging, one typically makes one of two assumptions: either \(\|\mathcal{X}_\mu K\| < 1\) so that \(3.3.5\) can be inverted using Neumann series [26][149], or that \(K\) is ‘large enough’ so that a diffusion equation approximation for \(3.3.5\) is applicable [26][285]. In the case that \(\|\mathcal{X}_\mu K\| < 1\), the Neumann series solution to \(3.3.5\) reads as

\[
w = (\mathcal{I} - \mathcal{X}_\mu K)^{-1} \mathcal{X}_\mu \Xi = \sum_{n=0}^{\infty} (\mathcal{X}_\mu K)^n \mathcal{X}_\mu \Xi.
\]

(3.3.6)

In any case, when \((\mu, k)\) is such that a solution to \(3.3.5\) exists, the solution restricted to the outflow boundary again defines the propagation operator \(\mathcal{P}\):

\[
\mathcal{P} = (\mathcal{I} - \mathcal{X}_\mu K)^{-1} \mathcal{X}_\mu|_{\partial_+ \Gamma}
\]

(3.3.7)
In [260], it is shown that this (generically) gives rise to a bounded operator $\mathcal{P} : L^2(V) \to L^2(\partial_+ \Gamma)$.

### 3.3.1 The Adjoint Propagation Operator and the Normal Operator

In Section 3.3, we derived the propagation operator $\mathcal{P}$ for emission imaging systems that are described by an RTE. In the scatter-free case, we saw that this operator is the attenuated X-Ray transform, evaluated on the outflow boundary (equation (3.3.4)). An important exercise with any linear operator is to derive a formula for its $L^2$ adjoint, $\mathcal{P}^\dagger$. Since $\mathcal{P} : L^2(V) \to L^2(\partial_+ \Gamma)$, the adjoint will be a mapping $\mathcal{P}^\dagger : L^2(\partial_+ \Gamma) \to L^2(V)$. To derive the adjoint, we will use the definition:

$$\langle \mathcal{P} f, v \rangle_{L^2(\partial_+ \Gamma)} = \langle f, \mathcal{P}^\dagger v \rangle_{L^2(V)} \quad (3.3.8)$$

The formula for $\mathcal{P}^\dagger$ is found in [131] and [260], but we provide a full derivation here. The derivation relies on a ‘polar coordinates’ type lemma for computing integrals of functions $u \in L^2(\Gamma)$, given as Lemma 2.1 in [55]. This lemma states the obvious (once you draw a few pictures - see Figure 3.3) fact that any phase point $(r, \hat{s}) \in \Gamma$ can be parameterized uniquely by some $(r', \hat{s}') \in \partial_+ \Gamma$: in other words, we can change variables from $(r, \hat{s})$ to $(r', \hat{s}')$ by using the relation $r' = r + \tau_+ \hat{s}$ (recall the definition of $\tau_+$ in Figure 3.2) and $\hat{s}' = \hat{s}$ to obtain

$$\int_{V \times S^2} u(r, \hat{s}) \, dr d\hat{s} = \int_{\partial_+ \Gamma} \int_0^{\tau_-(r', \hat{s})} u(r' - \ell \hat{s}, \hat{s}) \, d\ell \, |\hat{s} \cdot \hat{\nu}| \, dr d\hat{s} \quad (3.3.9)$$

Note that the inner integral is from zero to the reverse exit time, traveling backwards from the boundary. The proof of equation (3.3.9) is found in Appendix 6.4 and is a nice exercise in the multivariate change-of-variables formula; the crux of the proof is showing that $dr d\hat{s} = |\hat{s} \cdot \hat{\nu}| dr' d\hat{s}' d\ell$.

To proceed deriving $\mathcal{P}^\dagger$, we recall the formula for $\mathcal{P}$ from (3.3.3) and (3.3.4), simplifying the
notation by defining a function \( E(r, \hat{s}, \ell) \) as follows:

\[
(Pf)(r, \hat{s}) = \frac{1}{4\pi c m} \int_{\tau - (r, \hat{s})}^{\tau + (r, \hat{s})} f(r - \ell \hat{s}) \exp \left( - \int_{0}^{\ell} \mu(r - \ell' \hat{s}) \, d\ell' \right) \, d\ell
\]

\[
= \frac{1}{4\pi c m} \int_{\tau - (r, \hat{s})}^{\tau + (r, \hat{s})} f(r - \ell \hat{s}) E(r, \hat{s}, \ell) \, d\ell
\]

\[
\Rightarrow E(r, \hat{s}, \ell) = \exp \left( - \int_{0}^{\ell} \mu(r - \ell' \hat{s}) \, d\ell' \right)
\]

We now start from the left side of (3.3.8), using the inner product (3.2.4), and re-arrange until we can apply (3.3.9) and obtain something that looks like the right side of (3.3.8):

\[
(P f, v)_{L^2(\partial_+ \Gamma)} = \int_{\partial_+ \Gamma} (Pf)(\gamma) v(\gamma) \, |\hat{s} \cdot \hat{\nu}| \, d\gamma
\]

\[
= \frac{1}{4\pi c m} \int_{\partial_+ \Gamma} \int_{\tau - (\gamma)}^{\tau + (\gamma)} f(r' - \ell \hat{s}) E(r, \hat{s}, \ell) \, d\ell |\hat{s} \cdot \hat{\nu}| \, d\gamma
\]

\[
= \frac{1}{4\pi c m} \int_{V} \int_{S^2} f(r, \hat{s}) E(r + \tau_+ \hat{s}, \hat{s}, \tau_+) v(r + \tau_+ \hat{s}, \hat{s}) \, d\hat{s} \, dr \quad \text{(by (3.3.9))}
\]

\[
= \frac{1}{4\pi c m} \int_{V} f(r) \left( \int_{S^2} E(r + \tau_+ \hat{s}, \hat{s}, \tau_+) v(r + \tau_+ \hat{s}, \hat{s}) \, d\hat{s} \right) \, dr
\]

\[
= (f, P^\dagger v)_{L^2(V)}
\]

Note that we used the function \( \tilde{f}(r, \hat{s}) \equiv f(r) \) only in order to apply equation (3.3.9); the \( \hat{s} \) variable is not necessary. Thus we have for any \( v \in L^2(\partial_+ \Gamma) \) that

\[
(P^\dagger v)(r) = \frac{1}{4\pi c m} \int_{S^2} E(r + \tau_+ \hat{s}, \hat{s}, \tau_+) v(r + \tau_+ \hat{s}, \hat{s}) \, d\hat{s}
\]

(3.3.11)

While it looks obtuse, the formula (3.3.11) actually has a simple interpretation: like the usual backprojection formula for the Radon or X-Ray transform [26], it considers a point \( r \in V \); it then goes outward in every direction, picking up the signal defined on the boundary (this is \( v(r + \tau_+ \hat{s}, \hat{s}) \)), then sums the result, weighted by the appropriate attenuation factor. The geometry implied by (3.3.11) is illustrated in Figure 3.3.
Figure 3.3: Geometry for the adjoint propagation operator $P^\dagger$ (in 2D view). Given the point $r \in V$, we trace every possible line out to the boundary $\partial V$, compute $v(r + \tau_+ \hat{s}, \hat{s})$ there, then integrate with the weight $E(r + \tau_+ \hat{s}, \hat{s}, \tau_+)$ where $E(\cdot)$ is defined in (3.3.10).

By composing $P^\dagger$ and $P$, one can obtain the so-called normal operator for propagation,

$$N_p = P^\dagger P,$$

(3.3.12)

where the subscript $p$ indicates that this is the normal operator for the propagation component alone; later, we will consider the normal operator $N = H^\dagger H$ for a complete system model, and study its properties numerically using the Fourier Crosstalk Matrix (FCM). The normal operator is frequently employed to study linear inverse problems, because its spectral properties (eigenvalues and eigenfunctions) lead directly to practically relevant information about solving the deterministic problem $Pf = \bar{g}$; this analysis leads to the Singular Value Decomposition or SVD, and we discuss it further in Section 3.5.1. If $K \neq 0$, the authors in [260] show that $N_p = P^\dagger P$ is a relatively compact perturbation of the $K = 0$ normal operator, and use this fact to prove that $P$ is injective (has no null functions) for an open dense set of $(\mu, k)$. This result implies that it is unlikely\(^3\) that a null component will arise due to the propagation operator: any system null functions will

---

\(^3\)we say ‘unlikely’ because there are several physical assumptions involved here, such as isotropy of $\mu$; the fact that the result in [260] is a ‘generic’ result makes it difficult to make a concrete statement about a particular object
thus arise as a result of the detection operator $\mathcal{D}$, which is more or less under the control of the system designer. This fact is extended in [131] to the case where only partial data is available, where the author defines partial data as measuring $w$ for a restricted set $\mathcal{A} \subset \partial_+ \Gamma$, for instance as defined by an arrangement of detectors. They then prove that only functions supported in the so-called visible set, which is a subset of the field of view $V$, will have no null component; any object which has support outside the visible set may have nonzero null component. This result also assumes that non-noisy, perfectly spatially and angularly resolved data is available, which is unrealistic, but it still indicates that partial data may lead to null functions. Again, this is a result of a detection process (which restricts the available data), not as a result of the propagation process; since the detection process is at least partially under the system designer’s control, the possibility of choosing a design which minimizes the impact of null functions on task performance is certainly plausible; see also [20]. In our case, we view the results of [260] and [131] as giving an absolute upper bound on system performance. Since any real detector will not measure $w(r, \hat{s}, \mathcal{E})$ with infinite accuracy for any subset $\mathcal{A} \subset \partial_+ \Gamma$, but will rather measure some averaged, restricted, and noisy version of it, we expect that actual systems will demonstrate worse performance than predicted by [260] and [131], and we propose numerical strategies to study the exact performance of a particular system design, taking into account all realistic effects. We also note that [260] and [131] provide only theoretical guarantees which relate to visible and measurement sets, and offer no specific computational approaches to computing null functions: the exact form of the null component $f_n$ of an object $f$ is crucial in the context of $\mathcal{M}$-PMED.

### 3.4 Radiation Detection and ECT System Modeling

Recall that the objective of these sections is to describe the relationship between the tagged drug concentration $c$ and the recorded photon data $g$. In Section 3.3, we discussed how $c \in L^2(V)$ is mapped to a phase space distribution $w \in L^2(\partial_+ \Gamma)$ on the boundary of the domain via the propagation operator $\mathcal{P} : L^2(V) \rightarrow L^2(\partial_+ \Gamma)$. We will now describe how real radiation detectors are modeled using an operator $\mathcal{D} : L^2(\partial_+ \Gamma) \rightarrow \mathcal{Y}$, where $\mathcal{Y}$ is a vector space of ‘average data’. The composition of $\mathcal{D}$ and $\mathcal{P}$ will give rise to the full system model $\mathcal{H} = \mathcal{D}\mathcal{P}$, which describes the...
mapping from \( c \) to the mean detector data \( \langle g | c \rangle \); Poisson statistics will then describe the statistical fluctuations of \( g | c \).

The detector model consists of two parts: an aperture (or collimator) which organizes rays into a usable form, and a detector that records either individual photon counts or estimates photon interaction attributes.

### 3.4.1 Collimators and Pinhole Apertures

The radiation flux available for detection on the boundary, i.e. \( w(r, \hat{s}, \mathcal{E}) \) with \( (r, \hat{s}) \in \partial_+ \Gamma \), is highly unstructured. In theory, one could simply place a position-sensitive detection surface outside the field of view \( V \) and detect incoming rays, but this would lead to an essentially useless dataset because we would have no knowledge of the direction from which the energy came. This would be like exposing traditional camera film to light without a lens - no information is conveyed on the resulting blank photo! In order to collect useful image data, we need to know something about both the position of a ray and the direction it came from. The way to gain directional information is through the use of apertures. An aperture (or collimator), in the context of high-energy radiation such as gamma rays, is simply a physical obstacle that blocks all rays outside a certain set; see Figure 3.4b. We assume the dimensions of any aperture are sufficiently larger than the wavelength of the radiation so diffraction effects can be ignored.

The effect of an aperture is to restrict the set of outgoing directions \( \gamma \in \partial_+ \Gamma \); in other words, an aperture (or collection of apertures) defines a set \( \mathcal{A}_{\text{det}} \subset \partial_+ \Gamma \) of detectable attributes. We will assume that an aperture is perfectly attenuating away from the opening, so that its action on \( w \) is as a multiplication operator with a sharp cutoff function: for a single aperture, we can define \( \mathcal{A} : L^2(\partial_+ \Gamma) \to L^2(\partial_+ \Gamma) \) via

\[
(\mathcal{A} w)(r, \hat{s}) = T_{ap}(r, \hat{s}) w(r, \hat{s}), \quad T_{ap}(r, \hat{s}) = \begin{cases} 1 & \hat{s} \in C(r) \subset S^2 \\ 0 & \text{else} \end{cases}
\]  

(3.4.1)

where \( C(r) \) is the cone with vertex \( r \in \partial V \) defined by the aperture (see Figure 3.5). Note that \( T_{ap}(r, \hat{s}) \) is simply the indicator function of the (complicated) set \( \mathcal{A}_{\text{det}} \subset \partial_+ \Gamma \). We can also consider
(a) Without an imaging aperture, a ray detected at $r \in P$ cannot be localized.

(b) The pinhole aperture reduces the uncertainty in the direction of the ray.

Figure 3.4: Illustration of the effect of an aperture on angular uncertainty. With no aperture, we cannot localize the ray any better than the cone formed by the field of view; with an aperture, this cone is narrowed.

A system with multiple apertures, for example if each detector has multiple pinhole apertures, or a parallel bore collimator [26], or the system has multiple detectors each with one or more apertures. With such a collection, we would modify (3.4.1) to

$$ (A_w)(r, \hat{s}) = T_{ap}(r, \hat{s})w(r, \hat{s}), \quad T_{ap}(r, \hat{s}) = \begin{cases} 1 & \hat{s} \in C_j(r) \subset S^2 \text{ for some } j \\ 0 & \text{else} \end{cases} $$

(3.4.2)

where $C_j(r)$ is the cone with vertex $r \in \partial V$ defined by aperture $j$.

At this point, we are almost able to apply a very powerful result proved by Hubenthal in [132]. There, the author (who also considers a nonzero scattering operator $K$), considers the ‘detection’ of $w$ for a subset of outgoing directions, $A_{det} \subset \partial_+ \Gamma$, as defined by a detector arrangement (in our case, the system of apertures). However, in [132], they assume that it is possible to evaluate $w(r, \hat{s})$ for all $(r, \hat{s}) \in A_{det} \subset \partial_+ \Gamma$. While the aperture operator $A$ does define such a subset of ‘detectable’ points, it is actually impossible to evaluate $w(r, \hat{s})$ exactly in practice; it is always necessary to average $w$ over subsets of $\partial_+ \Gamma$, or as we will see, draw samples $\hat{a}_1, \ldots, \hat{a}_n$ from a PDF.
which is proportional to a blurred version of \( w \). While it is possible to process these samples to produce an estimate of \( w \), it is unnecessary to do so: we can work directly with the samples \( \hat{a}_j \) themselves. In any case, we must consider an additional operator \( D : L^2(\partial_+ \Gamma) \to \mathcal{Y} \), where \( \mathcal{Y} \) is an appropriate vector space of mean detector data. Before discussing the two forms of \( D \) typically encountered in ECT systems, we spend a brief moment discussing some common system designs for preclinical small-animal ECT systems and the statistical principles of photodetection which lead to the Poisson noise model. We focus on preclinical small-animal systems primarily because they are simpler to describe: clinical ECT systems will usually have a rotating gantry, which introduces some minor mathematical modeling complications.
3.4.2 ECT System Design

While it is not the main goal of this work to discuss the specific engineering design aspects of ECT systems (see for example the recent dissertations from the Center for Gamma-Ray Imaging (CGRI), [198] [48]), we would like to briefly discuss how the geometry of an ECT systems influences the system operators that describe the mapping from the object $f$ to the imaging data $g$.

We will focus on a class of ECT system designs based on the FastSPECT series of systems developed at CGRI [160] [93] [196]. In such systems, an example of which is illustrated in Figure 3.6, a series of $N_d$ fixed detectors is arranged around a field of view $V \subset \mathbb{R}^3$, which we take to a ball of radius $r_V$. We assume that each detector is a distance $\ell_d$ from the origin, with the normal of the detector face given by the direction $\hat{n}_j$, $1 \leq j \leq N_d$. Each detector consists of an aperture, and the collection of all $N_d$ apertures gives rise to an aperture function $T_{ap}(r, \hat{s})$ as in (3.4.2) and Figure 3.5.

![Diagram of a fixed-head SPECT imaging system](image)

**Figure 3.6**: Two-dimensional rendering of a fixed-head SPECT imaging system, similar to the FastSPECT series of small-animal imaging system designs [160] [93] [196]. The system consists of $N_d$ detectors, each with a single pinhole aperture and detector module.

In Matlab, we have implemented a basic ECTSystem object which defines the system geometry, provides methods to display the system geometry, and compute the resulting system operator $\mathcal{H}$. 

146
The object consists of an array of \texttt{Detector} objects, where \texttt{Detector} is either a \texttt{PPDetector} or a \texttt{IntDetector}, corresponding to the case of a photon-processing detector and an integrating detector (described below). We abstractly associate an instance \texttt{S} of \texttt{ECTSystem} with the corresponding system geometry; namely, \texttt{S.nd} gives the number of detectors, \texttt{S.lvd} gives the detector distance to the origin, etc.

### 3.4.3 Poisson Photointeraction Statistics

As discussed above, an ECT system consists of an arrangement of detectors and apertures around a field of view \(V\). The phase-space distribution \(w = \mathcal{P}f\), restricted to the detectable attribute set \(\mathcal{A}_{\text{det}} \subset \partial_+ \Gamma\), is now ‘available’ for detection. As we mentioned briefly above, it is unfortunately not possible for any real detector to evaluate \(w(r, \hat{s}, \varepsilon)\) for \((r, \hat{s}) \in \mathcal{A}_{\text{det}}\): photodetection is a stochastic process, and the statistics of this process dictate both the design of systems and the mathematical modeling of the data produced. See [134] for an extensive discussion of stochastic processes in high-energy photodetection.

While space prohibits us to work through the entire physical derivation of photodetection statistics (which would require a discussion of quantum electrodynamics – see e.g. [26] and [246]), the physical postulates which lead to the Poisson detection model are relatively simple to state and are as follows. We suppose that a detector is described by a 2D planar set \(D \subset \mathbb{R}^2\) with outward pointing (i.e. away from the field of view) unit normal \(\hat{n}\). We then define the attribute set for this detector as

\[
\mathcal{A} = D \times S_+^2 \times [0, \infty) \times [0, \infty),
\]

where \(S_+^2 = \{ \hat{s} \in S^2 : \hat{s} \cdot \hat{n} > 0 \}\) is the hemisphere pointing away from the field of view. An element of \(\mathcal{A}\) is denoted \(\mathbf{a} = (r, \hat{s}, \varepsilon, t)\), and is called an event \textit{attribute}. Note that the attribute set \([3.4.3]\) corresponds to a unique subset \(\mathcal{A}_{\text{det}}\) of \(\partial_+ \Gamma \times [0, \infty) \times [0, \infty)\), where \((r, \hat{s}) \in D \times S_+^2\) corresponds to a point \((r - \tau \hat{s}, \hat{s}) \in \partial_+ \Gamma\). Thus we could have defined the attribute set in terms of \(\partial_+ \Gamma\), but the definition \([3.4.3]\) seems more natural in this setting. We assume that any function

---

\(^4\)real detectors have depth as well, but we do not consider this case here.
$w \in L^2(\partial \Gamma \times [0, \infty)^2)$ has been extended to be a function $w \in L^2(\mathcal{A})$ by this correspondence.

With this notation, the Poisson photodetection postulates are as follows: 

1. Photointeractions occur discretely and randomly; each interaction corresponds to an attribute $a \in \mathcal{A}$, i.e. a position, direction, energy and time. In other words, for any given subset $A \subset \mathcal{A}$, we have a finite random number $N(A) \in \{0, 1, 2, \ldots\}$ of photointeraction events with attributes $a \in A$.

2. For any other subset $A' \subset \mathcal{A}$, the number of photointeractions $N(A')$ is statistically independent from $N(A)$ if $A' \cap A = \emptyset$, i.e. the number of events that occur in non-overlapping subsets of attribute space are statistically independent.

3. As the size of the set $A$ shrinks to a point i.e. $A \rightarrow a$, the probability of observing a single count in $A$, i.e. $\mathbb{P}(N(A) = 1)$ approaches a constant (that depends on $a = (r, \hat{s}, \mathcal{E}, t)$ times $\Delta A = \Delta r \Delta \hat{s} \Delta \mathcal{E} \Delta t$, i.e.

$$\mathbb{P}(N(A) = 1) \rightarrow \lambda(a) \Delta r \Delta \hat{s} \Delta \mathcal{E} \Delta t,$$

and this constant is equal to the differential light intensity (i.e. photon flux) incident on $A$, multiplied by a quantum efficiency:

$$\lambda(a) = \lambda(r, \hat{s}, \mathcal{E}, t) = c_m \eta(\mathcal{E}) w(r, \hat{s}, \mathcal{E}, t) \hat{s} \cdot \hat{\nu} \quad (3.4.4)$$

4. As the set $A$ shrinks to a point $a$, the probability of observing two or more counts in $A$ approaches zero, i.e.

$$\lim_{\Delta A \rightarrow 0} \mathbb{P}(N(A) > 1) = 0$$

These postulates imply that the random number $N(A)$ must be Poisson distributed, i.e. we have

$$\mathbb{P}(N(A) = n) = \frac{\exp(-\lambda(A))(\lambda(A))^n}{n!}$$
Furthermore, the constant $\lambda(A)$ is given by

$$
\lambda(A) = \int_A \lambda(a) da = c_m \int_A \eta(E) w(r, \hat{s}, E, t) \hat{s} \cdot \hat{\nu} \, dr d\hat{s} dE dt
$$

In the particular case that $A = P \times S \times E \times T \subset A$, we have

$$
\lambda(A) = c_m \int_T \int_E \int_S \int_P w(r, \hat{s}, E, t) \hat{s} \cdot \hat{\nu} \, dr d\hat{s} dE dt
$$

If furthermore $w$ is constant in time, then we have

$$
\lambda(A) = c_m \Delta t \int_E \int_S \int_P w(r, \hat{s}, E) \hat{s} \cdot \hat{\nu} \, dr d\hat{s} dE
$$

Such a process – i.e. one which assigns a Poisson random number to each subset $A \subset A$, is a Poisson Point Process (PPP) in attribute space (see Section 2.6), with intensity function $\lambda(a) = c_m \eta(E) w(r, \hat{s}, E, t) \hat{s} \cdot \hat{\nu}$. We can thus think of photointeractions as occurring with well-defined attributes $a \in A$. A realization of a PPP in attribute space thus takes the form

$$
u(a) = \sum_{j=1}^{n} \delta(a - a_j), \quad a = (r, \hat{s}, E, t), \quad (3.4.5)
$$

where $n$ is a realization of a Poisson random variable $N \sim \text{Poi}(\bar{N})$, where

$$
\bar{N} = c_m \int_A \eta(E) w(r, \hat{s}, E, t) \hat{s} \cdot \hat{\nu} \, da
$$

it is assumed that $\bar{N}$ is finite. Note that the statement (3.4.5) does not contradict any quantum measurement hypotheses about measuring position and momentum simultaneously; we will see that real detectors can only estimate the attribute $a$, hence there is an estimation variance involved that should satisfy the appropriate uncertainty principle.
3.4.4 Photon-Counting Detector Model

In the previous section, we demonstrated that photointeractions occurring within a detector form a Poisson point process. We now consider the case that there is a collection of detectors, \( D_1, \ldots, D_{nd} \), each of which is partitioned into disjoint subsets \( d_m \), with \( 1 \leq m \leq M_j \), giving a total of \( M = M_1 \times \cdots \times M_{nd} \) detector pixels, for instance as shown in Figure 3.7. Each detector pixel is assumed to be able to count photons that interact within its boundaries. We also assume for simplicity that \( w \) is independent of time.

For a fixed amount of time \( \Delta t \), the number of photointeractions \( g_m \) that occurs within pixel \( d_m \) is Poisson distributed with mean

\[
\tilde{g}_m = c_m \Delta t \int_0^\infty \eta(\mathcal{E}) \int_{S_+^2} \int_{d_m} w(r, \hat{s}, \mathcal{E}) \hat{s} \cdot \hat{\nu} \, dr \, d\hat{s} \, d\mathcal{E}
\]

where again \( S_+^2 \) is the hemisphere of directions pointing towards the field of view. Because there are \( M \) total detector pixels, the complete data produced is thus (conditional on \( w \)) a Poisson random
vector \( \mathbf{g} \mid \mathbf{w} \) with mean given by

\[
\bar{\mathbf{g}} = \langle \mathbf{g} \mid \mathbf{w} \rangle = \mathcal{D}_{pc} \mathbf{w},
\]

where \( \mathcal{D}_{pc} : L^2(\mathcal{A}) \to \mathbb{R}^M \) is the continuous-to-discrete photon-counting detector model, with

\[
(\mathcal{D}_{pc} \mathbf{w})_m = c_m \Delta t \int_0^\infty \eta(\mathcal{E}) \int_{S^2_+} \int_{\partial_m} \mathbf{w}(\mathbf{r}, \hat{s}, \mathcal{E}) \hat{s} \cdot \mathbf{n} \, d\mathbf{r} \, d\hat{s} \, d\mathcal{E} \quad (3.4.7)
\]

Hence we have arrived at the complete system model for an ECT imaging system with pixelated (photon-counting) detectors. We propagate the source \( \mathbf{f} \) through the medium to obtain \( \mathbf{w} = \mathcal{P} \mathbf{f} \), where \( \mathbf{w} \equiv \mathbf{w}(\mathbf{r}, \hat{s}, \mathcal{E}) \) with \( (\mathbf{r}, \hat{s}) \in \partial \Gamma \). Then, the detector and aperture arrangement produces \( \mathbf{w} = \mathcal{A} \mathcal{P} \mathbf{f} = T_{ap}(\mathbf{r}, \hat{s}) \mathbf{w}(\mathbf{r}, \hat{s}, \mathcal{E}) \). Lastly, the photon-counting detector model \( (3.4.7) \) acts to produce the vector \( \bar{\mathbf{g}} = \mathcal{D}_{pc} \mathbf{w} = \mathcal{D}_{pc} \mathcal{A} \mathcal{P} \mathbf{f} \).

The complete system model is thus given by an operator

\[
\mathcal{H}_{pc} : L^2(\mathcal{V}) \to \mathbb{R}^M, \quad (3.4.8)
\]

This operator is given explicitly by

\[
(\mathcal{H}_{pc} \mathbf{f})_m = \bar{g}_m = c_m \Delta t \int_0^\infty \int_{S^2_+} \int_{\partial_m} T_{ap}(\mathbf{r}, \hat{s})(\mathcal{P} \mathbf{f})(\mathbf{r}, \hat{s}, \mathcal{E}) \hat{s} \cdot \mathbf{n} \, d\mathbf{r} \, d\hat{s} \, d\mathcal{E} \quad (3.4.9)
\]

A key feature of the photon-counting detector model is that the finite-dimensional range \( \mathbb{R}^M \) immediately implies that the system model \( \mathcal{H}_{pc} \) has an infinite-dimensional null space; this is a result of the singular value decomposition, discussed in Section 3.5.

Because we have assumed Poisson statistics, the random vector \( \mathbf{g} \mid \mathbf{f} \) is independent Poisson
distributed with component means given by \((3.4.9)\), i.e.

\[
p_g f(x) = \prod_{m=1}^{M} \frac{\exp(-\bar{g}_m)(\bar{g}_m)^{x_m}}{x_m!} = \prod_{m=1}^{M} \frac{\exp(-(\mathcal{H}_{pc}f)_m)((\mathcal{H}_{pc}f)_m)^{g_m}}{g_m!} \tag{3.4.10}
\]

where \(x \in \mathbb{Z}_0^M\) is a vector of nonnegative integers. Assuming a vector of counts \(g\) has been observed, we can take \((3.4.10)\) to define a likelihood function and log-likelihood for the object \(f\) (refer to Section \(2.7.2\)):

\[
\mathcal{L}_{pc}(f|g) = \prod_{m=1}^{M} \frac{\exp(-(\mathcal{H}_{pc}f)_m)((\mathcal{H}_{pc}f)_m)^{g_m}}{g_m!} \tag{3.4.11}
\]

\[
\ell_{pc}(f|g) = \sum_{m=1}^{M} [g_m \ln(\mathcal{H}_{pc}f)_m - (\mathcal{H}_{pc}f)_m] \tag{3.4.12}
\]

### 3.4.5 Photon-Processing Detector Model

In the photon-processing model, we consider a scintillation-type detector such as the Anger camera \cite{227} or the iQID detector \cite{197}. In such a camera, a single photointeraction occurs with attribute \(a = (r, \hat{s}, E, t)\) as described by the PPP model of photointeractions. However, the interaction is not immediately binned or recorded as having taken place in some pixel, but rather the event produces a secondary shower of visible light scintillation photons which can subsequently be recorded and used to estimate the original attribute vector \(a\) as accurately as possible. Modern gamma-ray detectors are capable of performing maximum likelihood estimation of \(a\) in real-time \cite{45}; to perform this event estimation, we require a likelihood model.

As stated above, the original interaction produces a secondary visible light transport process in the detector. This secondary light is then imaged or sensed on the backside of the detector using either \(p\) photomultiplier tubes or a semiconductor photon counting device with \(p\) pixels. Thus a single primary event with attribute \(a\) is mapped to a discrete random vector \(u|a \in \mathbb{Z}_0^p\); we typically assume that \(u|a\) is Poisson with mean \(\bar{u}(a)\); the function \(\bar{u}(a)\) is called the Mean Detector Response Function or MDRF \cite{43}. Note that the MDRF is a \(p\)-dimensional vector-valued function, where \(p\) is the number of photomultipliers or pixels; occasionally, these are arranged in a grid, so that \(\bar{u}(a)\) is a matrix-valued function (see e.g. Figure \(3.8\)). Supposing that we observe
counts $u|a = x$, we define the attribute likelihood and attribute log-likelihood via

$$L(a|x) = p_{u|a}(x) = \prod_{j=1}^{p} \frac{\exp(-\bar{u}_j(a))(\bar{u}_j(a))^{x_j}}{x_j!},$$

$$\ell(a|x) = \sum_{j=1}^{p} (x_j \ln \bar{u}_j(a) - \bar{u}_j(a)),$$

where for both we have $x \in \mathbb{Z}_{\geq 0}^p$. Using maximum likelihood estimation, we can estimate the most likely attribute for each event: this attribute estimate is denoted $\hat{a}_{ML}$ (or simply $\hat{a}$ for simplicity) and is defined via

$$\hat{a} = \arg \max_{a \in A} p_{u|a}(x) = \arg \max_{a \in A} \ell(a|x).$$

We note that in practice, an approximate MDRF $\bar{u}(a)$ is measured for each detector via a detector calibration process, whereby a series of point sources with attributes $a_j$ are created, then the corresponding $\bar{u}(a_j)$ are estimated and an interpolation model is chosen so that arbitrary $\bar{u}(a)$ can be evaluated [175] [133]. Performing the MLE attribute estimation (3.4.15) on-the-fly is challenging, because in principle $A$ is a 7-dimensional set, and we must hence solve a 7-dimensional optimization problem for each detected photon; fast parallel optimization algorithms such as the contracting grid search [129] have been developed to tackle this problem, at least when a small number of attributes are estimated. We also note that timing is usually performed by independent circuitry and does not necessarily require MLE, though we assume it is estimated via (3.4.15) anyway for convenience.

It is also not always the case that all the elements of the attribute $a$ are estimable from $u$ – for example, most cameras estimate only position $r \in D$ and do not attempt to estimate the direction $\hat{s}$. Calibration of the MDRF for larger attribute dimensions is also challenging – MDRF calibration for only 2 attributes, i.e. an $x$ and $y$ position on the detector face, already takes many hours, and the number of sampling points is exponential in the dimension, if a uniform grid is employed.

If the dimension of $u$ (i.e. the number of secondary photodetectors) is large, the asymptotic
properties of maximum likelihood estimation imply that to high accuracy we can write

$$\hat{a} = a + \epsilon, \quad \epsilon \sim N(0, F(a)^{-1}) \quad (3.4.16)$$

where $F(a)$ is the *Fisher information matrix* for the attribute estimate, defined via:

$$(F(a))_{ij} = -\left\langle \frac{\partial^2}{\partial a_i \partial a_j} \ln p(u|a) \right\rangle_{u/a}$$

Now, suppose that a Poisson random number $n$ photon events occurs, and for each event we estimate an attribute vector $\hat{a}$. We can thus form a Poisson point process of *estimated* attributes, $g(\hat{a})$, as follows:

$$g(\hat{a}) = \sum_{j=1}^{n} \delta(\hat{a} - \hat{a}_j) \quad (3.4.17)$$

Note the similarity between (3.4.17) and (3.4.5) - the actual interactions occur with attributes $a_j$, while (3.4.17) provides *estimates* $\hat{a}_j$. We illustrate the process of computing $\hat{a}$ using the MLE procedure (3.4.15) in Figure 3.8 using simulated data and an assumed MDRF form.

The mean (i.e. intensity function) of the point process (3.4.17) is derived in [44] and [175]. Assuming maximum likelihood estimation noise as in (3.4.16), and the possibility of a system sensitivity cutoff function $s(a)$, the photon-processing detector model thus takes the form of a convolution of the (cutoff) photon density $s(r, \hat{s}) w(r, \hat{s})$ with an *estimation blur* kernel $k_e(a, \hat{a})$:

$$\bar{g}(\hat{a}) = (D_{pp} w)(\hat{a}) = \int_{A} k_e(a, \hat{a}) \lambda(a) \, da, \quad (3.4.18)$$

where $\lambda(a)$ is given by (3.4.4). Hence the *photon-processing detector* model is an operator

$$D_{pp} : L^2(A) \to L^2(A) \quad (3.4.19)$$
Figure 3.8: Simulated MLE attribute estimation procedure for the estimation of an interaction position \( \mathbf{a} = (x, y) \). The secondary photodetector was assumed to be a \( 512^2 \) grid of pixels, with MDRF taken to be a Gaussian, with \( \tilde{u}_{jk}(x, y) = \exp\left(-\frac{(x - x_{jk})^2 + (y - y_{jk})^2}{2\sigma^2}\right) \), where the \( (x_{jk}, y_{jk}) \)s correspond to the centers of the \( 512^2 \) pixels.
The complete system model for a photon-processing emission imaging system thus takes the form

\[ H_{pp} : L^2(V) \rightarrow L^2(\mathcal{A}), \quad (H_{pp}f)(a) = (D_{pp}Pf)(a) = c_m \Delta t \int_{\mathcal{A}} k_e(a, \hat{a}) \eta(\mathcal{E})(Pf)(a) \hat{s} \cdot \hat{n} \, da \]

Notice that the propagation operator \( P \) is the still given by (3.3.7). The key difference between \( H_{pp} \) and \( H_{pc} \) is that \( D_{pp} \) has infinite-dimensional codomain \( L^2(\mathcal{A}) \) while \( D_{pc} \) has only finite-dimensional codomain. Thus the issue with null functions is not inherent to the photon-processing detector. In fact, since convolution of an \( L^2 \) function with a nondegenerate multivariate Gaussian has no null space, the only possible null functions arise from the aperture function \( T_{ap}(r, \hat{s}) \). Of course, the singular values of \( D_{pp} \) will decay at an exponential rate (see e.g. \[148\]), so we expect that there is still an effective null space, where certain functions will be exponentially attenuated below say, machine precision, and hence produce ‘effectively’ zero data.

We can now define the likelihood for photon-processing data, using Theorem 3 in Section 2.6.2:

\[ L_{pp}(f|g) = \exp(-\bar{N}(f)) \frac{1}{n!} \prod_{j=1}^{n} (H_{pp}f)(\hat{a}_j) \]

\[ \ell_{pp}(f|g) = \sum_{j=1}^{n} \ln((H_{pp}f)(\hat{a}_j)) - \bar{N}(f). \]

### 3.5 Inverse Problems in Emission Imaging

“And this is the true rule by which those who analyse the effects of nature must proceed: and although nature begins with the cause and ends with the experience, we must follow the opposite course, namely, begin with the experience, and by means of it investigate the cause.”

*Leonardo da Vinci [70]*

We have now described how the drug concentration \( c_j(r, t) \) is related to the detected data \( g \), in two cases: one, when the detector bins photointeraction events into pixels so that \( g \) is a finite-dimensional Poisson random vector, and two, when the detector performs event attribute estimation so that \( g \) is a realization of a Poisson point process. In either case, the propagation...
model $P$ takes the form of a solution operator to a certain PDE, while the detection model $D$ takes the form of a compact integral operator. In both cases, the complete system model operator $H : L^2(V) \to Y$ is a compact linear operator describing the mapping from $c_j$ to $\langle g | c_j \rangle$; since we are assuming Poisson statistics, knowledge of $\langle g | c_j \rangle$ suffices to describe the statistics completely. We thus have a complete deterministic and statistical model of ECT data $g$:

$$g|_{c_j} \sim \text{Poi}(Hc_j),$$

where $H$ is either $H_{pp}$ or $H_{pc}$, and $\text{Poi}(\cdot)$ indicates either a Poisson point process or a finite-dimensional Poisson random vector. We are now faced with a classic problem in science: we want to know $c_j$, but we’re stuck with $g$, which is a sample from a random process that is only indirectly related to $c_j$ through the linear operator $H$. Dealing with problems like this is the realm of inverse problems: we have described a certain ‘forward’ problem, that is, describing how to obtain $g$ given knowledge of a drug distribution $c_j$, but we are faced with the reverse problem: given knowledge of $g$, we would like to obtain information about $c_j$.

Inverse problems arise in nearly every branch of science. It would be impossible to summarize the entire field here, but let us only mention their application in image science [26] [29] [249], image processing [12] [305] [185], physical optics [107], quantum physics [176], and earth sciences [270]. The mathematics of inverse problems is interlaced with many subfields of interest, including Riemannian geometry, analysis of partial differential equations, harmonic analysis, convex geometry [97], complex analysis and numerical analysis.

There are several categorical approaches to solving inverse problems; many are summarized in [26]. First, broadly speaking, methods can be categorized as being deterministic or statistical. We briefly discuss some highlights of deterministic theory.

In the deterministic theory of inverse problems, we typically consider an operator of the form $H : X \to Y$, where $X$ and $Y$ are typically Banach or Hilbert spaces. This gives rise to an operator equation of the form $y = Hx$. The application of the operator $H$ is taken as the ‘forward’ operator, while the solution of the equation is what is meant by the ‘inverse’ problem. Neither $x$ nor $y$ nor $H$ are taken as random; even if $y$ is a realization of some random process, we assume
this realization to be fixed. As first elucidated by Jaques Hadamard [118], a solution to such a
problem (indeed, he claimed any physical problem) is only ‘satisfactory’ if it satisfies three rules
(stated assuming that $\mathcal{X}$ and $\mathcal{Y}$ are metric spaces):

1. For every $y \in \mathcal{Y}$, there exists an $x \in \mathcal{X}$ such that $y = \mathcal{H}x$ (a solution exists);

2. For every $y \in \mathcal{Y}$, there is exactly one $x \in \mathcal{X}$ such that $y = \mathcal{H}x$ (the solution is unique);

and

3. If $d_{\mathcal{Y}}(y_1, y_2)$ is small, $d_{\mathcal{X}}(x_1, x_2)$ is comparably small, where $x_1$ and $x_2$ are the respective
solutions to $y_j = \mathcal{H}x$

If a problem satisfies (1)-(3), it is said to be well-posed; otherwise, it is ill-posed [86]. In the case
that $\mathcal{A}$ is a bounded linear operator between Banach spaces, conditions (1)-(3) amount respectively
to the condition that the range of $\mathcal{H}$ is $\mathcal{Y}$, that $\mathcal{H}$ has trivial null space, i.e. $N(\mathcal{H}) = \{0\}$, and that
the inverse operator $\mathcal{H}^{-1}$ is bounded from $\mathcal{Y}$ to $\mathcal{X}$ [86]. In fact, the third condition is redundant
in this case, as a consequence of the bounded inverse theorem [232, Theorem III.11].

The following result crashes the hopes that many ‘typical’ inverse problems will be well-posed
in this deterministic sense: [86, page 38]

**Theorem 4.** Let $\mathcal{H} : \mathcal{X} \to \mathcal{Y}$ be a compact linear operator between Hilbert spaces. Then, the
equation $y = \mathcal{H}x$ is ill-posed.

*Proof.* See [86].

As an example, many imaging systems can be modeled (in the mean) as linear continuous-to-
discrete operators $\mathcal{H} : \mathcal{X} \to \mathbb{R}^M$; such an operator is worse than compact, it is finite rank, and
hence has infinite-dimensional null space [26]. Thus the equation $\mathcal{H}f = \bar{g}$ is always ill-posed.

In light of this fact, several deterministic approaches have arisen which claim to resolve the
issue of ill-posedness. Turning an ill-posed problem into a well-posed problem is categorically
called *regularization* [26] [86] [29]. Many (if not most) deterministic regularization methods define
the regularized solution to $y = \mathcal{H}x$ via an optimization problem:

$$x^* = \arg\min_x F(x, y) + R(x; \theta)$$  (3.5.1)

where $F(\cdot)$ is a data fit functional and $R(\cdot; \theta)$ is a regularization functional, which depends on a vector of regularization parameters $\theta$. In principle, (3.5.1) is a variational problem, because $x$ is typically infinite-dimensional; a finite-dimensional approximation of $x$ converts (3.5.1) into a finite-dimensional optimization problem.

We will discuss statistical methods of regularization which extract estimates of object properties despite ill-posedness; there are deep connections between many seemingly deterministic regularization methods of the sort (3.5.1) (such as the classical Tikhonov method) and the statistical methods we discuss below; see e.g. [267] [150] [266]. It has also been argued extensively that regularization – specifically the selection of a particular regularization method and regularization parameters – must be framed in the setting of task-based image science; see [26] and Section 5.2. To our knowledge, a rigorous well-posedness theory of task-based image science has not been attempted: it is certainly empirically true that systems which correspond to ill-posed operator equations still give rise to data which can be used to perform tasks, but proving rigorously that the selection of a task (which is effectively a form of dimensionality reduction) regularizes an ill-posed problem is (to our knowledge) an untapped area of study; see Section 5.2 (Future Work).

3.5.1 Functional analysis of system operators: Null space, range, SVD

Given a compact linear operator $\mathcal{H} : \mathcal{X} \to \mathcal{Y}$ where $\mathcal{X}$ and $\mathcal{Y}$ are Hilbert spaces, we would like an efficient way to describe its mapping properties and the specific ill-posedness behavior of $\mathcal{H}$. The Singular Value Decomposition (SVD) is one of the most effective methods available – it describes the null space, range, measurement space, consistency space and stability properties in terms of a set of orthonormal eigenfunctions of the normal operator $\mathcal{N} = \mathcal{H}^\dagger \mathcal{H}$.

**Theorem 5** ([26] [232]). Let $\mathcal{H} : \mathcal{X} \to \mathcal{Y}$ be a compact linear operator with adjoint $\mathcal{H}^\dagger$. Then, there exists a sequence $\sigma_1 \geq \sigma_2 \geq \cdots \sigma_r > 0$ and orthonormal sequences $\{v_j\}_{j=1}^r \subset \mathcal{X}$ and
\{u_j\}_{j=1}^r \subset \mathcal{Y} \text{ such that } \mathcal{H}v_j = \sigma_ju_j, \text{ and for every } f \in \mathcal{X},
\begin{align*}
\mathcal{H}f &= \sum_{j=1}^r \sigma_j(f,v_j)_\mathcal{X}u_j \quad \text{and} \quad \mathcal{H}^\dagger g = \sum_{j=1}^r \sigma_j(g,u_j)_\mathcal{Y}v_j \tag{3.5.2}
\end{align*}

Furthermore, \((\sigma_j^2, v_j)\) are the eigenvalues and eigenfunctions of \(\mathcal{N} = \mathcal{H}^\dagger \mathcal{H}\), and \((\sigma_j^2, u_j)\) are the eigenvalues and eigenfunctions of \(\mathcal{N}^\dagger\). If \(r < \infty\), we can complete the sets \(\{v_j\}\) and \(\{u_j\}\) so that they are orthonormal bases for \(\mathcal{X}\) and \(\mathcal{Y}\), respectively; the vectors \(\{v_j\}_{j=r+1}^\infty\) span the null space of \(\mathcal{A}\), and the vectors \(\{u_j\}_{j=r+1}^\infty\) span the null space of \(\mathcal{A}^\dagger\).

The number \(r\) is called the rank of the operator \(\mathcal{H}\), and it can be finite or infinite. A null function is any function \(f_n\) such that
\[ \mathcal{H}f_n = 0 \]

The set of all null functions forms a subspace \(\mathcal{V}_n \subset \mathcal{X}\) called the null space of \(\mathcal{H}\), and the vectors \(v_j\) corresponding to zero singular values form a basis for \(\mathcal{V}_n\). The vectors \(v_j\) which correspond to non-zero singular values span the orthogonal complement of \(\mathcal{V}_n\), which is a subspace \(\mathcal{V}_m\) called the measurement space of \(\mathcal{H}\). Given any \(f \in \mathcal{X}\), the orthogonal projection of \(f\) onto the null and measurement spaces gives rise to a decomposition \[ f = f_m + f_n, \quad f_m \in \mathcal{V}_m, \quad f_n \in \mathcal{V}_n \tag{3.5.3} \]

We thus say that \(\mathcal{X} = \mathcal{V}_m \oplus \mathcal{V}_n\). The vector \(f_m\) is called the measurement component of \(f\), and the vector \(f_n\) is called the null component of \(f\). Applying \(\mathcal{H}\) to \(f\), we see that
\[ \mathcal{H}f = \mathcal{H}(f_m + f_n) = \mathcal{H}f_m + \mathcal{H}f_n = \mathcal{H}f_m \]
so that only the measurement component contributes to the (mean) data \(\bar{g}\). In Section 3.6 we will say that \(f_n\) is nonestimable from \(g \sim \text{Poi}(\mathcal{H}f)\), because we can change \(f_n\) arbitrarily and the distribution of \(g\) does not change.
Returning to our central problem of understanding uncertainties in \(\mathcal{M}\)-PMED, recall that we seek a representation of the object \(f\) as a measurement and null component, i.e. we have written the central \(\mathcal{M}\)-PMED problem as

\[
\tilde{q} = \mathcal{M}(\tilde{f}_m, \tilde{f}_n, \tilde{f}_c)
\]

The estimated measurement component \(\tilde{f}_m\) will arise from an attempt to reconstruct (estimate) \(f_m\) from the data \(g\), while the estimated null component \(\tilde{f}_n\) must be selected as a model of \(f_n\) (the control component \(\tilde{f}_c\) is independent from the data and corresponds to a parameterization of a treatment). The SVD of \(\mathcal{H}\) suggests a concrete way to construct \(\tilde{f}_m\) and \(\tilde{f}_n\): they should be generalized random vectors taking values in \(\mathcal{V}_m\) and \(\mathcal{V}_n\), respectively.

### 3.5.2 Generalized Crosstalk Matrices

While the SVD (3.5.2) provides an appealing, elegant method to decompose an operator \(\mathcal{H}\), and provides an explicit basis for the null space and measurement space, the SVD suffers from a distinct drawback: the decomposition is system-dependent, and it requires solving the (infinitely many) eigenvalue problems

\[
\mathcal{H}^\dagger \mathcal{H} v_j = \sigma_j^2 v_j \tag{3.5.4}
\]

While in some idealized cases the eigenvalue problems (3.5.4) are analytically solvable [26], for many system operators \(\mathcal{H}\) this is not the case, and a numerical method must be employed to compute the singular system for \(\mathcal{H}\). In fact, this is the same situation we faced with the Karhunen-Loéve expansion – there, we needed to solve the eigenvalue problems \(C_J \phi_j = \lambda_j \phi_j\) in order to obtain the KLE basis; we argued that in many cases, it is just as well to expand objects into a fixed basis or frame expansion

\[
f = \Phi^\dagger \tilde{\Phi} f = \sum_{j=1}^{\infty} F_j \phi_j, \quad F_j = (f, \tilde{\phi}_j)_{L^2(I)} \tag{3.5.5}
\]
We can take the same approach to the analysis of a system operator \( \mathcal{H} \); instead of seeking the SVD representation (3.5.2), we simply allow the object to be written as an expansion of the form (3.5.5), then compute the action of \( \mathcal{H} \) on \( f \) as follows:

\[
\mathcal{H} f = \mathcal{H} \Phi^\dagger \tilde{\Phi} f = \sum_{j=1}^{\infty} F_j \mathcal{H} \phi_j
\]

We define the operator \( \mathcal{A} = \mathcal{H} \Phi^\dagger \), which acts on an \( \ell^2 \) sequence as follows:

\[
\mathcal{A} z = \mathcal{H} \Phi^\dagger z = \mathcal{H} \sum_{j=1}^{\infty} z_j \phi_j = \sum_{j=1}^{\infty} z_j \mathcal{H} \phi_j
\]

With this definition, we have \( \mathcal{H} = \mathcal{A} \tilde{\Phi} \), which has the simple interpretation of first ‘analyzing’ \( f \) with the dual basis or frame \( \tilde{\Phi} \), then computing the action of \( \mathcal{H} \) on the resulting sequence of expansion coefficients, and synthesizing the resulting using \( \Phi \). We can now express the normal operator in terms of \( \mathcal{A} \) and the dual synthesis operator as follows:

\[
\mathcal{N} = \mathcal{H}^\dagger \mathcal{H} = (\mathcal{A} \tilde{\Phi})^\dagger (\mathcal{A} \tilde{\Phi}) = \tilde{\Phi}^\dagger \mathcal{A}^\dagger \mathcal{A} \tilde{\Phi}
\] (3.5.6)

Next, we define the operator \( \mathcal{B} = \mathcal{A}^\dagger \mathcal{A} \); this is called the generalized crosstalk matrix, because it is a generalization of the Fourier crosstalk matrix proposed by [25] [103] [19]. The operator \( \mathcal{B} \) is an infinite matrix, in the sense that it can be written as an operator

\[
\mathcal{B} : \ell^2 \to \ell^2
\]

The entries of this matrix are given by

\[
B(k, k') = (\mathcal{H} \phi_k, \mathcal{H} \phi_{k'})_\mathcal{Y}, \quad k, k' \in \mathbb{N}
\] (3.5.7)

The matrix (3.5.7) could also be called a Gram matrix [58]. If the inner products (3.5.7) are readily computable – either analytically or numerically – we can study the normal operator \( \mathcal{N} \) more directly without solving the eigenvalue problems (3.5.4). We also define a truncated version
of $B$,

$$B_n = (B(k, k'))_{k, k' = 1}^n,$$

which is an $n \times n$ matrix. In Section 3.8, we discuss computational methods for computing $B_n$ for ECT systems. The matrix $B_n$ can also be used to compute an approximate singular system for $H$ as follows. Assume that $\{\phi_j\}_{j=1}^\infty$ is an orthonormal basis, so that $\tilde{\Phi} = \Phi$ is a unitary operator. Then, by (3.5.6), we have that

$$\mathcal{N} v_j = \lambda_j v_j \Leftrightarrow \Phi^\dagger B \Phi v_j = \lambda_j v_j \Leftrightarrow B \Phi v_j = \lambda_j \Phi v_j$$

Hence, if we can solve the eigenvalue problem

$$B z_j = \lambda_j z_j,$$  \hspace{1cm} (3.5.8)

we can express the corresponding eigenfunction of $\mathcal{N}$ as

$$v_j = \Phi^\dagger z_j = \sum_{k=1}^\infty z_{j,k}\phi_k$$

We can approximate the solution to (3.5.8) by using $B_n$ instead of $B$, that is, we can compute the solutions to the finite-dimensional eigenvalue problems

$$B_n z_j^{(n)} = \lambda_j^{(n)} z_j^{(n)}$$

and set

$$v_j^{(n)} = \Phi^\dagger z_j^{(n)} = \sum_{k=1}^n z_{j,k}^{(n)} \phi_k$$  \hspace{1cm} (3.5.9)

In particular, this suggests a method to approximate the null functions of a system operator $H$: by finding the eigenvectors corresponding to zero or near-zero eigenvalues of $B_n$, we can estimate
null functions of $\mathcal{H}$ by (3.5.9). This approach to approximating the singular values and singular functions was also suggested by Hansen [126] [125]. As $n \to \infty$, both $\lambda_j^{(n)}$ and $v_j^{(n)}$ converge to the true singular values and singular functions, with a rate that depends on the decay of the singular values of $\mathcal{H}$. In Section 3.8, we will employ this method to estimate null functions and singular functions for a simple ECT system.

### 3.6 Statistical Approaches to Inverse Problems

As we discussed in the previous section, the typical situation in imaging is that we have a dataset $g|f \sim P_{g|f}$ which is indirectly related to the object $f \in \mathcal{X}$. Given knowledge of $g$, we would like to extract information about $f$. We discussed how certain deterministic properties of the system operator, such as null space and singular values, imply that only certain features of the object can be reconstructed. We will now return to the statistical viewpoint, whereby we treat $g$ as a realization of a generalized random vector, i.e. $g|f \sim \text{Poi}(\mathcal{H}f)$. Because each imaging dataset arises from a unique object $f$, and we also treat objects as realizations of generalized random vectors, we can state that, for a fixed object, the imaging data is a sample from the generalized random vector $g|f$; overall, we have that $(f,g) \sim P_{(f,g)}$ forms a generalized random vector in $\mathcal{X} \times \mathcal{Y}$.

In the language of Section 2.7.1 we can consider a statistical model $\mathcal{M}$ as follows:

$$\mathcal{M} = \{\text{Poi}(\mathcal{H}f) : f \in \mathcal{X}\}$$  \hspace{1cm} (3.6.1)

In other words, this statistical model for ECT data is nonparametric, where the set $\mathcal{X}$ – which is the vector space of realizations of the random process $f$ – plays the role of the (infinite-dimensional) parameter set $\Theta$. In many cases, we have prior knowledge that $f$ lies in some proper subset of $\mathcal{X}$. For instance with object components that represent concentrations or densities, we must have $f \geq 0$, and so we could replace $\mathcal{X}$ in (3.6.1) with $\mathcal{X}_+ \subset \mathcal{X}$ where $\mathcal{X}_+$ is the convex cone of non-negative functions [26] [59]. If the model $\text{Poi}(\mathcal{H}f)$ is physically correct – meaning that image data truly follows this distribution – then (3.6.1) is a correctly specified statistical model, meaning that there exists $f^* \in \mathcal{X}$ such that the measured data $g$ is ‘truly’ a sample from $\text{Poi}(\mathcal{H}f^*)$. In
practice, however, the model (3.6.1) is impractical for two reasons:

1. The set $\mathcal{X}$ is infinite-dimensional, meaning that we have ostensibly infinitely many parameters to estimate. While this is not necessarily an issue in theory, as discussed in Section 2.7.3, nonparametric estimation can lead to over-fitting if one is not careful. Furthermore, it is impossible to estimate infinitely many parameters in a computer with finite resources, data and time;

2. The operator $\mathcal{H}$ may have null functions or rapidly decaying singular spectrum, meaning that the full object $f$ may not be estimable. Recall from section 2.7.3 that a parameter $\theta \in \Theta$ is said to be non-estimable with respect to a statistical model $\mathcal{M}$ is there exists a $\theta' \in \Theta$ such that $P(\theta) = P(\theta')$. If $\mathcal{H}$ has a nontrivial null space, then we have $\mathcal{H}f = \mathcal{H}(f + f_n)$ for any $f_n \in \mathcal{V}_n$, and hence $\text{Poi}(\mathcal{H}f) = \text{Poi}(\mathcal{H}(f + f_n))$, and thus $f$ is not estimable.

Obviously, these two facts appear to be discouraging (in essence, we haven’t fixed ill-posedness by simply considering a statistical approach), but as we discussed previously, there are ways to deal with both of the issues above: namely, we will only try to estimate estimable parameters (meaning only seek to estimate $f_m$), and we choose finite-dimensional representations of the object $f$ that lead to efficient estimation strategies.

3.6.1 Maximum Likelihood Image Reconstruction

Given the statistical model (3.6.1), and in spite of the issues raised above, we would still like to compute an estimator of $f$ from the data $g|f \sim \text{Poi}(\mathcal{H}f)$. As we discussed in Section 2.7.3 one approach is to consider the Maximum Likelihood Estimator. Recall that for Poisson data, the likelihood is defined in one of the following ways, corresponding to if $g \in \mathbb{R}^M$ (photon-counting
detectors) or \( g \in \mathcal{F}' \) (photon-processing detectors):

\[
\mathcal{L}_{pc}(f|g) = \prod_{j=1}^{M} \frac{\exp(-(\mathcal{H}_{pc} f)_j)((\mathcal{H}_{pc} f)_j)^{g_j}}{g_j!} \tag{3.6.2}
\]

\[
\ell_{pc}(f|g) = \sum_{j=1}^{M} (g_j(\mathcal{H}_{pc} f)_j - (\mathcal{H}_{pc} f)_j) \tag{3.6.3}
\]

\[
\mathcal{L}_{pp}(f|g) = \frac{\exp(-\bar{N}(f)) n!}{n!} \prod_{j=1}^{n} (\mathcal{H}_{pp} f)(\hat{a}_j) \tag{3.6.4}
\]

\[
\ell_{pp}(f|g) = \sum_{j=1}^{n} \ln((\mathcal{H}_{pp} f)(\hat{a}_j)) - \bar{N}(f) \tag{3.6.5}
\]

where in the first two, \( g = [g_1, \ldots, g_M] \) is the Poisson random vector of counts, and in the second two,

\[
g \equiv g(\hat{a}) = \sum_{j=1}^{n} \delta(\hat{a} - \hat{a}_j)
\]

and

\[
\bar{N}(f) = \int_{A} (\mathcal{H} f)(a) \, da
\]

We define the \textit{idealized} maximum likelihood estimator of \( f \) to be the solution to

\[
\hat{f} = \arg \max_{f \in \mathcal{X}} \ell(f|g), \tag{3.6.6}
\]

where \( \ell(\cdot) \) is given by either (3.6.3) or (3.6.5). As we discussed in Section \ref{sec:2.7.3}, the solution to (3.6.6) may not exist. Due to the possible presence of null functions, it may also be the case that the solution to (3.6.6) is not unique; there may be infinitely many \( f \in \mathcal{X} \) that maximize \( \ell \).

To deal with these problems, we will choose a representation of \( f \) that uses only finitely many \textit{estimable} parameters; then, we will replace the idealized MLE (3.6.6) with an \textit{approximate} maximum likelihood procedure, and under certain additional conditions, the solution to this modified problem will be guaranteed to exist; we are not guaranteed that the approximate MLE is an un-
biased estimate of $f$, again because of null functions, but it will be an unbiased estimate of $f_m$, the measurement component of $f$. So, suppose that $f = f_m + f_n$ as in (3.5.3), where $f_m \in \mathcal{V}_m$ is the measurement component and $f_n \in \mathcal{V}_n$ is the null component. Then, suppose that $\{v_j\}_{j=1}^r$ is an orthonormal basis for the measurement space $\mathcal{V}_m$ (the rank $r$ may be infinite). We consider an expansion of $f_m$ into this basis:

$$f_m = \mathcal{V}_m^\dagger \mathcal{V}_m f_m = \sum_{j=1}^{r} \theta_j v_j, \quad \theta_j = (f_m, v_j)_{L^2(\mathcal{X})}$$

Then, we define the *measurement likelihood* as the function(al)

$$\ell^{(m)}(\theta|g) = \ell(\mathcal{V}_m^\dagger \theta|g) \quad (3.6.7)$$

where $\ell(\cdot|g)$ is either of the likelihoods (3.6.3) or (3.6.5). In words, the measurement likelihood considers a sequence $(\theta_1, \theta_2, \ldots, \theta_r)$ (where $r$ may be infinite); it first synthesizes a function $f_m = \mathcal{V}_m^\dagger \theta \in \mathcal{X}$, then evaluates the ‘true’ likelihood on this synthesized function. We then define the *measurement MLE* as the solution to the maximization problem

$$\hat{\theta}_m = \arg \max_{\theta \in \Theta} \ell^{(m)}(\theta|g), \quad (3.6.8)$$

where $\Theta$ is an appropriate set of expansion coefficients, for instance on which enforces positivity.

Because we want an estimate of $f_m$ and not its expansion sequence $\theta$, we can synthesize $f_m$ via

$$\hat{f}_m = \mathcal{V}_m^\dagger \hat{\theta}_m$$

Note that our choice of the full orthonormal basis $\{v_j\}_{j=1}^r$ for $\mathcal{V}_m$ is theoretically appealing, but this may still lead to an optimization problem (3.6.8) that is intractable: to resolve this, we can choose any orthonormal collection $\{\phi_j\}_{j=1}^p \subset \mathcal{V}_m$ of estimable functions, where $p$ is finite, then define the measurement likelihood via (3.6.7) and the approximate MLE (3.6.8).

Note that by making this approximation, we have broken the main assumption necessary to prove most of the appealing properties of the MLE: the model $\mathcal{M}$ is no longer well-specified, because
unless \( f \in \mathcal{Y}_m \) to start with, there will not exist an \( f_m \in \mathcal{Y}_m \) such that \( \text{Poi}(\mathcal{H} f_m) = \text{Poi}(\mathcal{H} f) \); the model has become misspecified. However, an appealing fact of maximum likelihood for misspecified models is that the resulting estimator minimizes the Kullback-Leibler divergence to the true solution \(^{[295]}\), which is optimal in an information theoretic sense.

As we have mentioned before, any estimator \( \hat{\theta} \) is a generalized random vector, because it has been written as a function of a random quantity, namely the data \( g \). Thus, \( \hat{f}_m \) is a random quantity: the corresponding sampling distribution corresponds to a random process, and this inspires the definition of the patient-specific virtual MLE ensemble as the ensemble (i.e. generalized random vector) with probability distribution \( P_{\hat{f}} \). If a Bayesian solution is employed, where \( \tilde{f}_m \) is treated as a sample from the posterior \( f_m | g \), an analogous virtual ensemble is defined; see Section \( 5.2 \).

### 3.7 Parallel Computation of Ballistic Transport: Ray Transforms on the GPU

In Section \( 3.3 \), we discussed the analytical solution of the forward problem for the stationary RTE, leading ultimately to the propagation operator \( \mathcal{P} \) defined in \( (3.3.4) \). In this section, we discuss a GPU-based parallel numerical solver for the ballistic RTE, i.e. the case where \( \mathcal{K} \equiv 0 \). This method will be employed as part of a system simulation package which computes deterministic properties (such as null functions, singular functions, singular values and crosstalk matrices) of the system operators \( \mathcal{H} \) as discussed in Section \( 3.4 \). For our immediate purposes, we only require the numerical solution of the forward operator \( \mathcal{H} = \mathcal{D}\mathcal{P} \); a numerical reconstruction method would also typically require an adjoint operator method, i.e. an implementation of \( \mathcal{H}^\dagger = \mathcal{P}^\dagger \mathcal{D}^\dagger \). It is also common for real imaging systems to measure an ‘\( H \) matrix’ by physically moving a source through the field of view and measuring the mean detected signal \( \bar{g} \) \(^{[26]}\), but we will not consider this approach here.

To reiterate, the objective is to analyze the deterministic properties of \( \mathcal{H} \) so that we can better understand uncertainties in \( \bar{q} = \mathcal{M}(\tilde{f}) \), some of which are due to these deterministic properties. As part of this endeavor, we require an efficient simulation method for the system operator \( \mathcal{H} \). We have developed a novel parallel method for accurately computing attenuated X-Ray transforms,
which is the appropriate propagation operator for the scatter-free case, and have implemented parallel quadrature methods for computing the action of the detection operators $\mathcal{D}_{\text{int}}$ and $\mathcal{D}_{pp}$. The combination of these two schemes provides an efficient parallel method for computing the system operator $\mathcal{H}$.

Numerical methods for the RTE can broadly be categorized as stochastic (i.e. based on Monte Carlo integration) or deterministic (the integrodifferential equation is discretized and the resulting system solved). Monte Carlo methods are discussed in e.g. [285] [179] while deterministic methods are discussed in e.g. [149] [203] [152] [83]. In general, solving the full RTE with a deterministic method is a challenging numerical problem, primarily due to domain dimensionality: the monoenergetic optical phase space $\Gamma$ is five-dimensional, so a typical numerical discretization scheme (e.g. finite difference or finite element) with $n_d$ nodes per dimension will require $n_d^5$ total nodes; if $n_E$ energies are considered, we require a grid of size $n_E \cdot n_d^5$. Even a modest discretization of 512 nodes per dimension and $n_E = 1$ would lead to $2^{45} \sim 10^{13}$ total nodes, a nearly impossible number to manage without a supercomputer; adding energy resolution only worsens the problem. Monte Carlo methods, while extremely effective for many problems, are inherently statistical and hence less naturally applicable when we wish to study deterministic properties of the system operator $\mathcal{H}$. For our purposes, we are interested in computing the mean detected data $\bar{g} = \mathcal{H} f$, or some functionals of it, for either the photon-processing case or the binned photon-counting case. For our purposes, a numerical method which computes $w(r, \hat{s}, E)$ for arbitrary $(r, \hat{s}) \in \partial_+ \Gamma$ on-the-fly is sufficient, and if $\mathcal{K} \equiv 0$, this problem is trivially parallelizable over rays. We thus avoid impossibly large arrays [22] by taking a ‘grid- and matrix-free’ approach.

Recall that the stationary ballistic RTE was described as the following operator equation,

$$
\begin{cases}
\mathcal{T}_\mu w = \Xi \\
w \big|_{\partial_- \Gamma} = 0
\end{cases}
$$

(3.7.1)

where $\Xi = \Xi(r, \hat{s}, E)$ is the photon source density, $\mathcal{T}_\mu = c_m \hat{s} \cdot \nabla_r + c_m \mu$. One of the conclusions of Section 3.3 was that (3.7.1) had an explicit solution, given by the attenuated X-Ray transform
of \( \Xi \):

\[
w(r, \hat{s}, \mathcal{E}) = (\mathcal{T}^{-1}_{\mu}(\Xi))(r, \hat{s}, \mathcal{E}) = (\mathcal{X}_{\mu}(\Xi))(r, \hat{s}, \mathcal{E})
\]

\[
= \frac{1}{c_m} \int_{0}^{r-} \Xi(r - \ell \hat{s}, \hat{s}, \mathcal{E}) \exp \left( - \int_{0}^{\ell} \mu(r - \ell' \hat{s}, \mathcal{E}) d\ell' \right) d\ell
\]

\[
= \frac{1}{c_m} \int_{0}^{r-} \Xi(r - \ell \hat{s}, \hat{s}, \mathcal{E}) E_{\mu}(r, \hat{s}, \ell, \mathcal{E}) d\ell.
\]

(3.7.2)

The function \( E_{\mu}(r, \hat{s}, \ell, \mathcal{E}) \) is defined as the exponential weighting factor

\[
E_{\mu}(r, \hat{s}, \ell, \mathcal{E}) = \exp \left( - \int_{0}^{\ell} \mu(r - \ell' \hat{s}, \mathcal{E}) d\ell' \right).
\]

(3.7.3)

If we assume that the source is isotropic and monoenergetic (a reasonable assumption in many ECT modalities), then we have

\[
w(r, \hat{s}) = \frac{1}{4\pi} (\mathcal{T}^{-1}_{\mu}(\mathbb{I}))(r, \hat{s}) = \frac{1}{4\pi c_m} \int_{0}^{r-} f(r - \ell \hat{s}) E_{\mu}(r, \hat{s}, \ell, \mathcal{E}) d\ell
\]

(3.7.4)

where \( E_{\mu}(r, \hat{s}, \ell, \mathcal{E}) \equiv E_{\mu}(r, \hat{s}, \ell, \mathcal{E}_0) \) is the weight \( (3.7.3) \) for \( \mathcal{E} = \mathcal{E}_0 \).

If \( K \neq 0 \) satisfied certain conditions, we found a solution of \( (3.7.1) \) in terms of a Neumann series \( 3.3.6 \). In either case, we defined the propagation operator \( \mathcal{P} \) to be the operator that maps a source (either \( \Xi \) or \( f \)) to the phase-space density on the outflow boundary, i.e.

\[
(\mathcal{P}\Xi)(r, \hat{s}, \mathcal{E}) = w(r, \hat{s}, \mathcal{E})\big|_{(r, \hat{s}) \in \partial_{+}\Gamma}
\]

Thus in order to simulate the propagation operator \( \mathcal{P} \) for \( K \equiv 0 \), we must be able to evaluate \( w(r, \hat{s}, \mathcal{E}) \) for \( (r, \hat{s}) \in \partial_{+}\Gamma \), hence be able to apply the operator \( (3.7.2) \) (or \( (3.7.4) \) in the monoenergetic, isotropic case). If \( K \neq 0 \), we may wish to employ an iterative scheme for the computation of the Neumann series \( 149 \), so it will still be necessary to compute \( (3.7.2) \). Furthermore, in Section \( 3.8 \) we will use a basis expansion for the function \( f \) to define the so-called crosstalk matrix, so we will also require an implementation of \( (3.7.2) \) or \( (3.7.4) \) that can compute the projection of arbitrary basis functions quickly and accurately.
Computing ray transforms such as (3.7.2) and (3.7.4) numerically is a standard routine in imaging and integral geometry, however most methods begin by making a voxel approximation of both \( \Xi \) (or \( f \)) and \( \mu \); see for instance Siddon [254] or [141], [95]; refer also to Figure 3.9. Other methods first make a voxel approximation in \( r \) then use domain rotation to address the angular variable \( \hat{s} \) [203], which converts the line integrals to simple matrix row or column sums. Methods based on voxels are both memory intensive and inaccurate, so we propose a grid-free method for computing (3.7.2), which assumes only that implementations of the functions \( \Xi \) and \( \mu \) are available, i.e. we have methods \( X_i(r,s,e) \) and \( \mu_i(r,e) \) which return the value of these functions for an arbitrary input. We then employ a numerical ODE solver to quickly compute (3.7.2) to arbitrary accuracy for any desired \((r, \hat{s}, \mathcal{E})\). Because (3.7.2) is trivially parallelizable over separate rays, we implement the method in CUDA so that it can be run on GPU machines (each thread will process a single \((r, \hat{s}, \mathcal{E})\)).

**Figure 3.9:** On the left, a 2D illustration of the voxel-based approach to computing (3.7.1) via ray-tracing. A voxel grid is chosen to overlay the domain \( V \). Then, for a given evaluation point \( \gamma \in \partial_+ \Gamma \), a line is drawn through the domain until it exits at \( r - \tau_{-}\hat{s} \). The integral along this line is then approximated by a Riemann-type sum of the form \( \sum w_j f(r_j) \), where \( w_j \) are some weights (usually taken to be the length of the ray-box intersection) and \( f(r_j) \) are function evaluations at some nodes (usually taken to be the center of the ray-box intersection). Such approximations are memory intensive and inaccurate; if an increase in accuracy is necessary for any evaluation point \( \gamma \), the entire grid must be made finer. On the right is the proposed voxel-free method whereby each ray is discretized individually.

We now briefly discuss the numerical solution of ODE on the GPU.
3.7.1 Numerical Solution of ODEs on the GPU

Suppose that we wish to compute \( w(r, \hat{s}, \mathcal{E}) \) via (3.7.2) for some \((r, \hat{s}) \in \partial_+ \Gamma\). Denote the reverse exit time of \((r, \hat{s})\) by \( \tau_- = \tau_-(r, \hat{s}) \) (refer to Figure 3.9). While it is possible to estimate both integrals in (3.7.2) using a numerical quadrature method, we recall that (3.7.2) was derived as the solution of the following ODE:

\[
\begin{align*}
\frac{d\tilde{w}}{dz}(z) + \tilde{\mu}(z)\tilde{w}(z) &= \frac{1}{c_m}\hat{\Xi}(z) \\
\tilde{w}(0) &= 0
\end{align*}
\]  

(3.7.5)

where \( \tilde{w}(z) = w(r + (z - \tau_-)\hat{s}, \hat{s}, \mathcal{E}) \), and \( \tilde{\mu} \) and \( \hat{\Xi} \) defined similarly via \( r \to r + (z - \tau_-)\hat{s} \). Then, we noted that \( \tilde{w}(\tau_-) = w(r, \hat{s}, \mathcal{E}) \). Thus to compute (3.7.2), it is equivalent to solve the ODE (3.7.5) until \( z = \tau_- \). We propose to use a numerical ODE method to solve (3.7.5).

There is a significant literature on solving ODEs numerically, and the general problem is considered to be completely understood theoretically; see e.g. [138], [120], [40], [166], [9]. In brief summary, methods for solving a general system of ODEs of the form

\[
\begin{align*}
\frac{dy}{dt} &= F(t, y) \\
y(t_0) &= y_0
\end{align*}
\]  

(3.7.6)

where \( y \in \mathbb{R}^d \) are usually classified as being either single-step or multi-step, explicit or implicit, and can sometimes be adaptive meaning that they modify their step size depending on the estimated error. The output of a numerical method for solving (3.7.6) will typically consist of two arrays, \( t = [t_0, \ldots, t_{N-1}]^\top \in \mathbb{R}^N \) and \( Y = [y_0, \ldots, y_{N-1}]^\top \in \mathbb{R}^{N \times d} \), where \( y_j \approx y(t_j) \) is the approximated solution to the true value \( y(t_j) \) at time \( t = t_j \). If the system (3.7.6) is decoupled, that is,

\[
F(t, y) = \left[ F^{(1)}(t, y^{(1)}(t)), \ldots, F^{(m)}(t, y^{(m)}(t)) \right]^\top
\]
where each $y^{(j)} \in \mathbb{R}^{d_j}$, then it admits a trivial parallelization by solving each problem

$$\begin{cases}
\frac{dy^{(j)}}{dt} = F^{(j)}(t, y^{(j)}) \\
y^{(j)}(t_0) = y_0^{(j)}
\end{cases} \tag{3.7.7}$$

separately (each such sub-problem maps to a thread). In the best-case scenario, the system is fully decoupled so that $d_j = 1$ and each of (3.7.7) is a one-dimensional problem. This will be the case for computing (3.7.2), because we will wish to evaluate $w(r, \hat{s}, \mathcal{E})$ for some discrete set of $(r_j, \hat{s}_j, \mathcal{E}_j)$, and each $j$ gives rise to an ODE of the form (3.7.7) (see equation (3.7.5)). Another case in which we have decoupled problems (though not necessarily fully decoupled) arises when we wish to solve the same ODE system for many initial conditions or parameters, such as during a Monte Carlo simulation.

Note that on the GPU, it is most convenient to use explicit, non-adaptive solution methods, because both implicit and adaptive methods require some form of control flow branching (in the form of if statements), which can cause slowdowns [53]. Some workarounds to this issue are discussed in e.g. [211].

To illustrate the parallel implementation of a decoupled problem of the form (3.7.7), we consider the venerable Runge-Kutta 4th order method (abbreviated RK4) [243] [170] [40], which is an example of a single-step, explicit, non-adaptive method with reasonably high approximation order and good efficiency [7]. To begin, suppose we wish to estimate (3.7.7) at $N$ fixed time points $t_0 < t_1 < \cdots < t_{N-1}$, where each time point is equally spaced i.e. $t_n - t_{n-1} = \Delta t$. We then have $\Delta t = (t_{N-1} - t_0)/(N - 1)$, and $t_n = t_0 + n \Delta t$. The RK4 method is an explicit single-step method, which means it iteratively updates $y_n^{(j)}$ with a rule of the form

$$y_{n+1}^{(j)} = y_n^{(j)} + \Delta t \varphi(y_n^{(j)}, t_n, \Delta t), \quad 0 \leq n \leq N - 2. \tag{3.7.8}$$

\[ \text{we consider this simple method for illustration purposes only; in practice, more sophisticated methods usually prove more effective} \]
The explicit form of the update $\varphi(\cdot)$ for RK4 is

$$
\varphi(y_n^{(j)}, t_n, \Delta t) = \frac{1}{6} \left( k_1^{(j)} + 2k_2^{(j)} + 2k_3^{(j)} + k_4^{(j)} \right)
$$

where

$$
k_1^{(j)} = F^{(j)}(t_n, y_n^{(j)})
$$

$$
k_2^{(j)} = F^{(j)} \left( t_n + \frac{\Delta t}{2}, y_n^{(j)} + \frac{\Delta t}{2} k_1^{(j)} \right)
$$

$$
k_3^{(j)} = F^{(j)} \left( t_n + \frac{\Delta t}{2}, y_n^{(j)} + \frac{\Delta t}{2} k_2^{(j)} \right)
$$

$$
k_4^{(j)} = F^{(j)} \left( t_n + \Delta t, y_n^{(j)} + \Delta t k_3^{(j)} \right)
$$

The RK4 method has global error that is fourth order in $\Delta t$, which means that if $y^{(j)}(t)$ is the true solution to (3.7.7), there exist constants $C > 0$ and $H > 0$ such that $\|y^{(j)}(t_n) - y_n\| \leq C(\Delta t)^4$ for all $\Delta t < H$ [120]. Thus if we wish to obtain IEEE double precision accuracy, so that $\|y^{(j)}(t_n) - y_n\| \leq \varepsilon_{mach} = 2^{-53} \approx 10^{-16}$, we would require a time step on the order of $\Delta t = O(10^{-4})$, where the constant implied by the big-oh depends on $t_{N-1} - t_0$ and the function $F^{(j)}$.

To implement the RK4 method on the GPU, we define a CUDA kernel that performs the update rule (3.7.8), as shown in Listing 3.1. In this kernel, thread $j$ acts on the portion of the solution vector $y$ corresponding to $[y_0^{(j)}, \ldots, y_{N-1}^{(j)}]$.

To demonstrate the effectiveness of the parallel RK4 method, we solve a simple 1D logistic ODE $y' = ry(1 - y/\theta)$ for a large number of random initial conditions $y(0) = y_0$, random rate constants $r$ and random carrying capacities $\theta$. We display a histogram of the resulting $y_{N-1}$ values for one such simulation in Figure 3.10a and the resulting trajectories in Figure 3.10b. This sort of simulation would be useful in a Monte Carlo computation of some expected value $\langle f(y) \rangle$. This example is relevant to our discussion of treatment efficacy prediction in Chapter 4.

In Figure 3.11, the computation time of the CPU and GPU methods are compared.
Figure 3.10: The parallel RK4 method applied to the solution of a system of fully decoupled logistic ODE $y' = ry(1 - y/\theta)$ with random parameters $(y_0, r, \theta)$. The initial condition $y_0$ is drawn I.I.D. from $U(0, 1)$, the rate $r$ is drawn I.I.D. from $U(0, 2)$, and the carrying capacity $\theta$ is drawn I.I.D. from $U(0.5, 1)$. $d = 2^{20}$ samples were computed, and the simulation took approximately 0.25 seconds.

Figure 3.11: A comparison of the parallel and serial RK4 methods, applied to the same problem discussed in Figure 3.10. The number of samples $(d)$ varies from $2^{15}$ to $2^{20}$. The runtimes $T$ in seconds are displayed on the left graph, while the speedup factor $T_{CPU}/T_{GPU}$ is shown in the right graph.
struct MyRHSFun{
    double operator()(double t, double y){
        return y*(1-y);
    }
};

void initializeODE(double* t, double* y, int N, int d){
    // Initialize t and y.
}

template<class RHS>
__global__ void RK4Kernel(double* t, double* y, int N, int d, RHS F)
{
    tid = ThreadIdx.x;  // Thread index
    if(tid<d){yj = &y[tid*N];}  // Part of solution for this thread

    double k1, k2, k3, k4, yn, tn;
    double dt = t[1] - t[0];  // Assumes constant step size

    for(int n=0; n<N-1; ++n)
    {
        yn = yj[n];  tn = t[n];
        k1 = F(tn, yn);
        k2 = F(tn + dt/2.0, yn + k1*dt/2.0);
        k3 = F(tn + dt/2.0, yn + k2*dt/2.0);
        k4 = F(tn + dt, yn + k3*dt);
        yj[n+1] = yj[n] + (dt/6.0)*(k1+ 2*k2 + 2*k3 + k4);
    }
}

int main()
{
    MyRHSFun F;
    int N = 1024;  // Number of time steps
    int d = 1024;  // Number of ODEs (and threads)
    double *t, *y;
    initializeODE(t, y, N, d);
    RK4Kernel<<<d,1>>>(t, y, N, d, F);
}

Listing 3.1: CUDA code for a parallel RK4 implementation for \(d\) one-dimensional problems. Note the usage of a template parameter RHS, which allows passing a ‘function’ class. The __global__ flag is a CUDA language device that indicates that this function is a parallel kernel (see section 2.3). The vectors \(y\) and \(t\) that are passed to the kernel store, respectively, the vector of times and the solution vector. The initial conditions are assumed to be stored in \(Y[j*N]\) for 0 \(\leq j < d\).
3.7.2 Voxel-free Method for Computing Ray Transforms on the GPU

Instead of making a voxel assumption to approximate the functions $\Xi$ and $\mu$ in (3.7.2), we assume only that objects $X_i$ and $\mu$ have been defined which can evaluate these functions for an arbitrary input $(r, \hat{s}, \hat{e})$. While such methods may be interpolatory-type, so that indeed there is some underlying $n$-D grid and thus $X_i(r, s, e)$ computes an interpolation between voxels, such a method is inherently memory intensive and inaccurate. As we discussed in Section 2.6, one can also represent a function with a Galerkin-type approximation of the form

$$f(r) \approx \sum_{j=1}^{m} \alpha_j \phi_j(r), \quad (3.7.9)$$

and such approximations tend to be of much higher accuracy for a lower storage overhead: we need only store the $m$ numbers $\alpha_1, \ldots, \alpha_m$, instead of a 3D grid (usually $m \ll n^3$). As discussed in Sections 2.2.5 and 2.5, the functions $\phi_j(r)$ could be Karhunen-Loève basis functions, Fourier modes, wavelets, lump functions or spline-type functions. In Section 3.8, for instance, we will require the computation of the operator (3.7.2) for an arbitrary basis function of the form $\phi_j(r)$, for instance $\phi_j(r) = S_V(r) \exp(2\pi i r \cdot k_j)$; such a function can be implemented as a simple function class `FourierMode` as discussed in Section 2.3.

Assume for convenience that $\Xi$ is isotropic and monoenergetic, and that the corresponding $f(r)$ is approximated by a Galerkin-type sum of the type (3.7.9). Then, to evaluate $w(r_0, \hat{s}_0, \mathcal{E}_0) = (Pf)(r_0, \hat{s}_0, \mathcal{E}_0)$ for $(r_0, \hat{s}_0, \mathcal{E}_0) \in \partial_+ \Gamma \times [0, \infty)$, we perform the following steps:

1. Find the reverse exit time $\tau_-=\tau_-(\gamma_0)$. Recall that $\tau_-(\gamma_0)$ is the unique parameter such that $(r_0 - \tau_- \hat{s}_0, -\hat{s}_0) \in \partial_+ \Gamma$ - see Figure 3.2

2. Set up the initial value ode

$$\begin{cases}
\frac{d\tilde{w}}{dz} = F(z, \tilde{w}; \gamma_0) \\
\tilde{w}(0) = 0
\end{cases}, \quad (3.7.10)$$

177
where

\[ F(z, \tilde{w}; \gamma_0) = -\mu(r_0 + (z - \tau_-)\hat{s}_0, \hat{s}_0, \mathcal{E}_0)\tilde{w}(z) + \frac{1}{cm}\Xi(r_0 + (z - \tau_-)\hat{s}_0, \hat{s}_0, \mathcal{E}_0) \]

3. Solve the ODE (3.7.10) using a standard numerical integration method such as RK4 until \( z = \tau_- \).

4. The final value of \( \tilde{w} \) is \( w(\gamma_0, \mathcal{E}_0) \). Note that the full solution history from Step 3 is not needed - only the final value \( \tilde{w}(\tau_-) \) is required.

This algorithm is embarrassingly parallel since, given a vector of evaluation points \([ (r_j, \hat{s}_j, \mathcal{E}_j) ]_{1 \leq j \leq n} \), steps 1-4 can be solved independently of each other, with each thread being responsible for one ‘ray’ defined by \( (r_j, \hat{s}_j, \mathcal{E}_j) \).

### 3.7.3 Object Oriented Implementation

The numerical method presented in Section 3.7.2 is fairly elementary; the primary challenge for implementing it is programmatic. We need to be able to specify the necessary input functions, i.e. \( f, \mu \) etc., in a somewhat arbitrary manner. In Matlab, this would be done rather simply using the function handle notation e.g. \( @\mu \) for a function \( y = \mu(r, s, e) \) or the inline notation \( \mu = @(r, s, e) (...) \). However, from a performance perspective, it is better to use compiled functions instead of functions defined at runtime; we thus employ the use of templates, which are a C++ programming device intended to be able to generate polymorphic code at compile-time.

Effectively, a template is a way to parameterize a class or method by another class or type: the user then selects the template parameter at compile-time, and the compiler generates a function with the templated object replaced by the specific object provided by the user. We will not discuss templates at length; refer to e.g. [265].

The template design we developed to solve the programming problem outlined above is called the operator-helper-integrator, and this design is shown in Figure 3.12.

In this design, a user first declares 3 supporting objects: a \( \mu \) object of type \texttt{Mufun}, a \( \Xi \) object of type \texttt{Xifun}, and a domain object of type \texttt{Domain}. The domain object will contain, among other
Figure 3.12: The basic operator-helper-integrator design. The goal is to allow function calls like \( u = K(f,x) \) for arbitrary function objects \( f \) and inputs \( x \). The operator object \( K \) encapsulates everything needed to perform the specified operation: any helper functions such as kernels, and references to the appropriate numerical methods to perform the operation. Note that this is only a conceptual diagram; actual implementations (like the RTE object) may be more or less complicated.

methods, a method to compute the forward and reverse exit times, \( \tau_{\pm}(r,\hat{s}) \), as shown in Figure 3.2. Then, an RTE object is declared using \( \text{RTE } T(\mu,\xi,D) \), after which a user can evaluate \( w(r,\hat{s},E) \) simply using the function notation \( y = w(r,s,e) \) as shown in Listings 3.2 and 3.3.

```
Mufun mu;  % Attenuation function
Xifun xi;   % Source function
Domain D;   % Domain function (defines exit times, boundary, etc.)

RTE w(mu,xi,D);  % Declare RTE evaluator object
vec3 r(1.0,0.0,0.0);  % A sample position
vec3 s(0.0,1.0,0.0);  % A sample direction
double e = 1;        % A sample energy
double y = w(r,s,e); % Compute w(r,s,e)
```

Listing 3.2: Example script for the RTE simulator.
template<class Domain, class MuFun, class XiFun>
class RTEBallistic
{
  public:
    // ODE right hand side, passed to the ode integrator
  struct OdeRHS
  {
    numtype taup, taum; // Domain entry & exit times
    numtype *r0, *s0; // Initial point & direction
    numtype e0, cm; // Energy & light speed
    MuFun mu;
    XiFun xi;
    __dhp__ OdeRHS(MuFun mu_in, XiFun xi_in, numtype cm_in)
    {
      mu = mu_in; xi = xi_in; cm = cm_in;
    }
    __dhp__ complexnum operator()(numtype t, complexnum y)
    {
      numtype tempr[2];
      tempr[0] = r0[0] + (t−taup−taum)*s0[0];
      tempr[1] = r0[1] + (t−taup−taum)*s0[1];
      return −mu(tempr, s0, e0)*y + (xi(tempr, s0, e0));
    }
  };
  // Constructor
  __dhp__ RTEBallistic(MuFun mu_in, XiFun xi_in, Domain D_in)
  {
    mu = mu_in; xi = xi_in; D = D_in; F = OdeRHS(mu, xi, cm);
  }
  // Evaluation function. Allows for simple w(r,s,e) evaluation.
  __dhp__ complexnum operator()(numtype *r, numtype *s, numtype e)
  {
    numtype tau[2]; // Exit times
    D.GetExitTimes(tau, r); // Get exit times
    F.r0 = r; F.s0 = s; // Set up ode RHS
    F.taup = tau[0]; F.taum = tau[1]; // Entry & exit times
    int nstep = ceil((tau[1]−tau[0])/hmin);
    if(nstep>0)
    {
      numtype h = (tau[1]−tau[0])/(numtype)nstep−1.0;
      return rk4(tau[0], 0.0, nstep, h, F)/cm; // Solve the ode
    }
    else{
      return complexnumtype(0.0);
    }
  }
  private:
    MuFun mu; XiFun xi; OdeRHS F; Domain D;
    numtype cm = 299792458; // Unless otherwise specified
};

Listing 3.3: Class definition of the RTE CUDA class. Note the use of template parameters <class Domain, class MuFun, class XiFun>. This enables fully generic usage: the user can define any desired domain, attenuation and source function, so long as they have the necessary evaluation functions (parentheses operators). Some class methods removed for clarity.
3.7.4 Code Verification and Performance Tests of Ray Transform Code

To verify that our RTE code is functioning correctly and to test performance, we use a known solution of the RTE for a particular 2D geometry, attenuation and source. Namely, we consider the X-Ray transform with isotropic source, zero attenuation and speed of light $c_m = 1$:

$$(\mathcal{P}_0 f)(r, \hat{s}, \mathcal{E}) = \frac{1}{4\pi} \int_0^{\tau-(r, \hat{s})} f(r - \ell \hat{s}) \, d\ell$$

The domain $V$ is taken to be a disc of radius $R$ in $\mathbb{R}^2$. We then consider the parallel-beam Radon geometry, as shown in Figure 3.13. Each point $(r, \hat{s}) \in \partial_+ \Gamma$ is parameterized by an $(\alpha, \beta)$, where $0 \leq \alpha \leq \pi$ and $-1 \leq \beta \leq 1$.

![Diagram of parallel-beam Radon geometry](image)

Figure 3.13: Illustration of the parallel-beam Radon geometry used to verify the ballistic RTE code.

With this geometry, we can write $\mathcal{P}_0$ as

$$(\mathcal{P}_0 f)(\alpha, \beta) = \frac{1}{4\pi} \int_{-\tau-(\alpha, \beta)}^{\tau-(\alpha, \beta)} f(t \theta(\alpha) + \beta \theta^\perp) \, dt \quad (3.7.11)$$
Here $\mathbf{\theta}(\alpha) = [\cos(\alpha), \sin(\alpha)]^T$ is a unit vector making angle $\alpha$ with the $x$-axis, and $\mathbf{\theta}^\perp = [-\sin(\alpha), \cos(\alpha)]^T$ is its perpendicular. For our verification test, we use the method of exact solution [216] to compare the numerical output to a known, analytic solution of the problem. We’ll apply (3.7.11) to a Fourier mode $f(r) = \Phi_k(r)$:

$$
\phi_k(r) = \exp(2\pi i \rho_k \cdot r) S_V(r)
$$

(3.7.12)

With the domain $V$ given by a disc of radius $R$, have

$$
(3.7.11) = \frac{1}{4\pi} \int_{-\sqrt{R^2-\beta^2}}^{\sqrt{R^2-\beta^2}} f(t\mathbf{\theta} + \beta \mathbf{\theta}^\perp) dt
$$

In particular, if we let $f(r) = \phi_k(r)$ as in (3.7.12), we have

$$
(P_0 \phi_k)(\alpha, \beta) = \frac{1}{4\pi} \int_{-\sqrt{R^2-\beta^2}}^{\sqrt{R^2-\beta^2}} \exp(2\pi i \rho_k \cdot (t\mathbf{\theta} + \beta \mathbf{\theta}^\perp)) dt
$$

$$
= \frac{\exp(2\pi i \beta \rho_k \cdot \mathbf{\theta}^\perp)}{4\pi} \int_{-\sqrt{R^2-\beta^2}}^{\sqrt{R^2-\beta^2}} \exp(2\pi i \rho_k \cdot \mathbf{\theta}) dt
$$

$$
= \frac{\sqrt{R^2-\beta^2}}{2\pi} \exp(2\pi i \beta \rho_k \cdot \mathbf{\theta}^\perp) \text{sinc} \left(2\sqrt{R^2-\beta^2} \rho_k \cdot \mathbf{\theta}\right)
$$

(3.7.13)

In Figure 3.14 we display a comparison of the exact solution (3.7.13) and the numerical solution computed for a fixed $\alpha$ and $n = 512$ $\beta$ values; specifically, we let $\beta_j = -1 + j \Delta \beta$ with $\Delta \beta = 2/(n-1)$ and $0 \leq j \leq n - 1$. The ODE integrator used was RK4 (see Section 3.7.1) with a fixed step length of $h$.

In Figure 3.15 we demonstrate the relative error between the numerical solution vector $\mathbf{w}_n$ and exact solution vector $\mathbf{w}_e$ as a function of the RK4 step length $h$. 

182
3.8 Numerical Computation of Fourier Crosstalk Matrices and the SVD

Let \( f \in L^2(V) \) be an object function with support set \( V \subset \mathbb{R}^d \). Then, given any basis or frame \( \{\phi_k\}_{k=1}^\infty \) for \( L^2(V) \), we can write \( f \) using the analysis and synthesis operators (see Section 2.2.5).
as follows:

\[ f(r) = \Phi^\dagger \tilde{\Phi} f = \Phi^\dagger F = \sum_{k=1}^{\infty} F_k \phi_k(r), \quad F_k = (f, \tilde{\phi}_k)_{L^2(V)} \]  

(3.8.1)

where the series converges in the \( L^2 \) sense. We gave several examples of such expansions in Sections 2.2.5 and 2.5 – one such basis is given by the standard orthonormal Fourier basis when \( V = Q_\ell = [0, \ell]^d \) is a cube of length \( \ell \) in \( \mathbb{R}^d \). Then,

\[ \phi_k(r) = \frac{1}{\ell^{d/2}} \exp(2\pi i \rho_k \cdot r) 1_{Q_\ell}(r) \]  

(3.8.2)

is an orthonormal basis for \( L^2(Q_\ell) \). The modes \( \rho_k \) are a re-ordered version of the usual wavenumbers \( k/\ell \), numbered for instance by increasing modulus: \( |\rho_{k+1}| \geq |\rho_k| \). The indicator function \( 1_V \) is defined as

\[ 1_V(r) = \begin{cases} 
1 & r \in V \\
0 & \text{else} 
\end{cases} \]

For convenience, we will only consider expansions of the sort (3.8.1) where \{\phi_k\} is an orthonormal basis for \( L^2(V) \), so that \( \tilde{\Phi} = \Phi^\dagger \). Now consider a bounded linear operator \( \mathcal{H} : L^2(V) \to \mathcal{Y} \), where \( \mathcal{Y} \) is the appropriate vector space of mean data; in Sections 3.4.4 and 3.4.5 we considered \( \mathcal{Y} = \mathbb{R}^M \) (we can also consider \( \mathcal{Y} = \mathbb{C}^M \) to allow complex data; this is only for the purpose of Fourier analysis - real detector data is real) and \( \mathcal{Y} = L^2(\mathcal{A}) \) for respectively photon-counting and photon-processing detectors. As we discussed in Section 3.5.2, we can write

\[ \mathcal{H} f = \mathcal{A} \tilde{\Phi} f = \sum_{j=1}^{\infty} (f, \tilde{\phi}_j)_{L^2(V)} \mathcal{H} \phi_j. \]

The normal operator \( \mathcal{N} = \mathcal{H}^\dagger \mathcal{H} \) then has a matrix representative with respect to the \( \Phi \) given by the crosstalk matrix \( \mathcal{B} \):

\[ \mathcal{B} = \mathcal{A}^\dagger \mathcal{A}, \quad \mathcal{B}(k, k') = (\mathcal{H} \phi_k, \mathcal{H} \phi_{k'})_{\mathcal{Y}} \]  

(3.8.3)
Thus assuming that we can pre-compute the action of $\mathcal{H}$ on the basis elements $\phi_k$, we can compute the crosstalk matrix (3.8.3). Recall from Section 3.5.2 that if we truncate (3.8.3), we obtain a finite-dimensional matrix

$$B_n = (B(k, k'))_{k, k'=1}^n$$

and by computing the spectral decomposition of $B_n$, i.e.

$$B_n = Z_n \Lambda Z_n^\dagger,$$

where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$,

we obtain approximations of the first $n$ singular values and singular functions via

$$v_j(r) \approx (\Phi_n^\dagger z_j)(r) = \sum_{k=1}^n z_{jk} \phi_k(r), \quad \sigma_j^2 \approx \sqrt{\lambda_j}.$$ (3.8.5)

In particular, if $\lambda_j \approx 0$, then $v_j(r)$ is an approximate null function of $\mathcal{H}$. We have given the expression for the right-singular functions $v_j(r)$, which span the domain of $\mathcal{H}$; if the left-singular functions (or vectors) $u_j$ are desired, one must compute $u_j = \mathcal{H}v_j/\sigma_j$.

For photon-counting detectors, $\bar{\mathcal{Y}} = C^M$, and so the inner product (3.8.3) is just the usual $M$-dimensional dot product:

$$(v, w)_{C^M} = \sum_{j=1}^M v_j w_j^*$$

In the photon-processing detector, $\bar{\mathcal{Y}} = L^2(\mathcal{A})$, and so the inner product in (3.8.3) is given by

$$(f, g)_{L^2(\mathcal{A})} = \int_{\mathcal{A}} f(\hat{a}) g^*(\hat{a}) d\hat{a}$$

To compute the crosstalk matrix, we require numerical methods to compute $\mathcal{H}\phi_k = D\mathcal{P}\phi_k$ for an arbitrary basis function $\phi_k$; we have already provided in Section 3.7.2 a numerical method to compute the action of the propagation operator $\mathcal{P}$ on an arbitrary basis element $\phi_k$. To compute
the action of the detector models, recall from equations (3.4.7) and (3.4.18) that
\[
(D_{\text{int}} w)_m = c_m \Delta t \int_0^\infty \eta(\mathcal{E}) \int_{\mathbb{R}^2} \int_{d_m} w(r, \hat{s}, \mathcal{E}) \hat{s} \cdot \hat{\nu} \: dr d\hat{s} d\mathcal{E} 
\] (3.8.6)
\[
(D_{pp} w)(\hat{a}) = c_m \Delta t \int_0^\infty \int_{\mathbb{R}^2} k_e(a, \hat{a}) \lambda(a) \: da, 
\] (3.8.7)

To compute either (3.8.6) or (3.8.7), we can employ numerical quadrature methods [166] [267].

To demonstrate the computation of $B$ for both photon-counting and photon-processing systems, we consider a simplified case where $V \subset \mathbb{R}^2$ is the unit disc and the system is a fixed-head 2D SPECT system as described in Section 3.4.2; we assume a simple 3-detector setup as displayed in Figure 3.16. We assume further that the photon-processing detector performs only estimation of the interaction position on the detector face; in the photon-counting detector case, each pixel has a width of $w_p$, while in the photon-processing case, we assume that the estimation blur in (3.8.7) is a Gaussian with Full-Width-Half-Max (FWHM) given by $w_e$, as displayed in Figure 3.17.

The Fourier crosstalk matrices for a particular choice of $w_p$ and $w_e$ are displayed in Figure 3.18.
Figure 3.17: Description of the variables $w_p$ and $w_e$ for the crosstalk matrix simulation. For the photon-processing detector, $D_{pp}$ is a convolution with a Gaussian estimation blur with Full-Width-Half-Max (FWHM) given by $w_e$. For the photon-counting detector, each pixel has a width of $w_p$.

Note that we have selected a photon-processing blur FWHM $w_e = 235\mu$m that is comparable to the pixel width $w_p = 234\mu$m. In the pixelated case, the crosstalk matrix demonstrates more significant off-diagonal aliasing, an indication of severe illposedness [25] [19].

Figure 3.18: Comparison of the Fourier crosstalk matrix for a photon-processing system (left) and an photon-counting system (right).
In Figure 3.19, we show the first 36 estimated singular functions for the same system settings as in Figure 3.18. Note that up to re-arrangement, the singular functions are effectively the same; the decay of the singular values are different, however (refer to Figure 3.20), demonstrating that the pixelated detector is more severely ill-posed than the photon-processing detector.

Figure 3.19: A comparison of the resulting right-singular functions \( v_j(r) \) estimated using the Fourier crosstalk matrix and the relation (3.8.5). On the left are the first 36 right-singular functions for \( H_{int} \), while on the right are the first 36 right-singular functions for \( H_{pp} \).

A comparison of the singular values for various values of \( w_e \) and \( w_p \) is shown in Figure 3.20. These results demonstrate that the photon-processing system tends to outperform a pixelated system with pixel size comparable to the estimation blur FWHM, at least in terms of singular value decay rate, which is related to the degree of ill-posedness of the resulting image reconstruction problem.
Figure 3.20: Comparison of the rate of decay of the singular value sequence for the photon-processing and photon-counting systems (log scale for the vertical). Note that in each case, a photon-processing system with FWHM approximately equal to the pixel width results in a much better conditioned system. In fact, the worst performing photon processing system nearly outperforms the best performing pixelated system.
Chapter 4

Model-based Precision Chemotherapy

Note: portions of this chapter appear in the submitted article [128]. This author acknowledges the contributions of his co-authors for portions of this work, in particular, the original conception of several of the models presented here was due to Barrett, Myers and Clarkson. Our novel contribution is to consider generalizations of this model and to place it within the larger framework of model-based precision medicine presented in this dissertation.

4.1 The Paradigm of in silico Model-based Precision Medicine

In this chapter, we discuss a class of image science tasks that are inspired by a new paradigm of medical decision making that we refer to as in silico model-based precision medicine, or \( \mathcal{M} \)-PMED for short. In this paradigm, a virtual patient is built using imaging and other data. The virtual patient is then used to make predictions about the real patient, using mathematical and computational models of diseases and treatments. With these predictions, a decision can be made regarding the appropriate course of treatment; refer to Figure 4.1 for a diagrammatic illustration of \( \mathcal{M} \)-PMED.

By virtual patient, we mean simply a computational object, abstracted say by a class \texttt{Patient} (refer to Section 2.3). An instance \( P \) of the class \texttt{Patient} has data elements \( P.f1 \), \( P.f2 \), and so on, which represent certain physiological random processes for this patient. If explicit data is available about such processes, it is assumed that this data is stored in \( P \) as well, say under elements...
Figure 4.1: A conceptual framework for $\mathcal{M}$-PMED, whereby patient-specific data is used to generate a virtual patient ensemble $\tilde{f}$, which can be used to make patient-specific model-based predictions $\tilde{q} = \mathcal{M}(\tilde{f})$. The entire framework can be iterated as treatment progresses over time, with new patient-specific data updating the existing ensemble. Treating this entire framework as a treatment option, classical clinical trials can be performed to assess its efficacy.

$\mathbb{P}.g_1$, $\mathbb{P}.g_2$ etc. More explicitly, imagine that the patient has undergone an $^{18}$F-FDG study and an MRI; $\mathbb{P}.g_1$ could contain the raw or processed PET data, while $\mathbb{P}.g_2$ could contain the raw or processed MRI data. The ‘virtual patient’ need not be any more complicated than this; it is simply a computational object which abstracts all the available data about the patient and provides all the desired computational methods we wish to perform with such data. It is imaginable that such a class Patient could become rather sophisticated as data collection and processing methods advance.  

As we have discussed in previous chapters, there are uncertainties in the data used to form the patient object $\mathbb{P}$, and so any data elements (e.g. $\mathbb{P}.f_1$, $\mathbb{P}.g_1$, etc) must be treated as random; in the most general sense, each data element in $\mathbb{P}$ can be treated as a generalized random vector in the sense of Chapter 2. For example, if $\mathbb{P}.g_1$ represents raw $^{18}$F-FDG PET data collected using a photon processing system, it represents a realization of a Poisson point process $g$. If then $\mathbb{P}.f_1$ represents, say, the patients’ standardized glucose uptake value (SUV) as a function of $(r, t)$, it

---

1 For instance, the company Heartflow® has several FDA-approved computational methods that employ complicated patient-specific finite element models of the heart. Such a Patient object is certainly rather sophisticated (and proprietary!)
is appropriate to treat $P.f1$ as a sample from either the posterior distribution $P_{f|g}$ or from the sampling distribution of an estimator such as the MLEM-estimator $\hat{f}$ as discussed in Chapter 3. In either case, we assume that $P$ possess a method $P.Randomize$ which is able to draw samples from the appropriate distributions of all its data elements; we discussed several such methods to do so in Sections 2.4 and 2.5. The possibility of randomizing the virtual patient is what motivates the terminology Virtual Patient Ensemble, or VPE as discussed in Section 1.1.

If some data element (call it $P.fn$ to stand for ‘null’) is not directly estimable via patient-specific measurements, but is otherwise necessary to estimate quantities of interest $q$ or make decisions about the patient, we could choose a random process model, for instance one of the standard choices discussed in Section 2.6, to stand in for the unmeasured data. Of course, this selection must be made with great caution, as the choice of random field model may affect the ultimate decision made about the patient – the decision is conditional on the choice of model $[146]$, and this is a delicate issue; our recommendation is that such a model, should it be deemed necessary, should always be based on some measured population calibration data and not chosen simply for convenience.

The virtual patient object $P$ will also have available methods, say $P.M1$, $P.M2$, etc, which compute quantities of interest, that is, $P.M1$ is a computational implementation of the model:

$$\tilde{q}_1 = M_1(\tilde{f})$$

(4.1.1)

where $\tilde{f}$ is parameterized by the data contained in $P.f1$, etc. Again, to be explicit, imagine that $P.f1$ contains a reconstructed $^{18}$F-FDG PET scan, normalized to quantify spatial SUV; a common (albeit simplistic) computational task is to compute a maximum SUV in a region of interest, the so-called SUVmax value $[158]$; the method $P.SUVmax$ would compute this quantity of interest for patient $P$. If $P.n$ represents a tumor cell density $n(r,t_0)$, another quantity of interest might be to predict the total number of tumor cells $N(t_1) = \int_V n(r,t_1) \, dr$ at some time $t_1 > t_0$; we discuss this type of problem later in this chapter.

Recall that in our notation, the quantity of interest $\tilde{q}$ is predicted from the approximated inputs
\( \tilde{q} = \mathcal{M}(\tilde{f}_m, \tilde{f}_n, \tilde{f}_c) \) as follows:

Recall that \( \tilde{f}_m \) is the observable component, meaning it can be estimated for a particular patient from (say) imaging data \( g \); \( \tilde{f}_c \) is a control component, which will represent the treatment and provides a pathway to affect the outcome \( \tilde{q} \); and \( \tilde{f}_n \) is a ‘null’ component that represents unobserved features of the patient. Recall that in Section 1.1 we defined the model error or residual \( \epsilon = q - \tilde{q} \), where \( q \) is the ‘true’ (gold-standard) value of the quantity of interest. We can thus write

\[
q = \mathcal{M}(\tilde{f}_m, \tilde{f}_n, \tilde{f}_c) + \epsilon
\]

The precise definition of the components \( \tilde{f}_m, \tilde{f}_n \) and \( \tilde{f}_c \) are part of the modeling process, and will depend on the available data and treatment scenario. As we mentioned in Section 1.1 we view (4.1.3) as the precision medicine equivalent to the \( g = \mathcal{H}f + n \) model from image science [26]. Unlike in image science, where the statistics of the noise term \( n \) usually follow from the physics of imaging (for instance, Poisson statistics), the statistics of the residual \( \epsilon \) can be fully determined only from a model validation and calibration procedure. We will not address this issue here (see Future Work, Section 5.2).

Because \( \tilde{f} \) is treated as random, \( \tilde{q} \) is random and hence our prediction of the quantity of interest is uncertain; in the context of precision medicine, \( \tilde{q} \) represents a predicted effect of a treatment, and so we will be interested in knowing the probability that the predicted effect satisfies some criterion of ‘successful’ treatment; this motivates our definition of patient-specific treatment efficacy as a probability of the form

\[
\mathcal{P} = \mathbb{P}(\tilde{q} \in E) = \mathbb{P}(\mathcal{M}(\tilde{f}) \in E). \tag{4.1.4}
\]

We thus assume that the virtual patient object \( \mathbb{P} \) has methods to perform Monte Carlo simulation to estimate expectations \( \langle F(\tilde{q}) \rangle \), in particular treatment efficacy probabilities of the form (4.1.4);
in Chapter 2, we discussed several such methods, in particular when \( \tilde{f} \) is a random process.

It is hoped that by judicious selection of \( \tilde{f}, M, \tilde{q} \) and \( E \), a very wide range of patient-specific figures of merit can be derived using (4.1.4): we discuss a particular instance of the modeling framework (4.1.3) and (4.1.4) in this chapter, identifying a particular \( \tilde{f}, M, \tilde{q} \) and \( E \) for the case of chemotherapy.

Suppose furthermore that the control input \( \tilde{f}_c \) in (4.1.2) depends on a set of tunable treatment parameters, say \( \tilde{f}_c = \tilde{f}_c(\zeta) \) where \( \zeta = (\zeta_1, \ldots, \zeta_p) \) – for example drug mass, injection times, injection sites, and drug types. The output \( \tilde{q} \), and hence the distribution \( P_{\tilde{q}} \) and the probability (4.1.4) all thus depend on \( \zeta \), and so we can consider the function

\[
P(\zeta) = P_{\tilde{q}(\zeta)}(E) = P(\tilde{q}(\zeta) \in E)
\]

where

\[
\tilde{q}(\zeta) = M(\tilde{f}_m, \tilde{f}_n, \tilde{f}_c(\zeta))
\]

Now, we can consider the following optimization problem, which defines patient-specific optimal treatment selection under the \( M \)-PMED paradigm:

\[
\zeta^* = \arg \max_{\zeta \in P} P(\zeta).
\]

In words, the optimization problem (4.1.6) selects the treatment (as parameterized by \( \zeta \)) which maximizes the probability of desired effect (as defined by \( P(\tilde{q} \in E) \)). We will explain later how we may need to consider multiple objective functions of the form (4.1.5) which may have competing interests, then develop a composite objective which balances these tradeoffs. For example, suppose \( q_1 \) measures the predicted effect of a drug on neoplastic tissue, while \( q_2 \) measures the predicted effect of a drug on normal tissue; we would then wish to maximize \( P_1(\zeta) \) while simultaneously minimizing \( P_2(\zeta) \). We view the successful implementation of optimization problems of the form (4.1.6) as one of the central problems in \( M \)-PMED; see also Section 5.2 (Future Work).
Of course, treatment selection need not be so mechanical: patient preference to risk aversion and informed consent should also play a role. We will later discuss briefly how to display a visualization of predicted tradeoffs so that a personal decision can be made by the patient and clinician, but we otherwise do not discuss these delicate (and deeply interesting) issues.

The paradigm presented above is, of course, completely breaking with the norm of medical treatment. As we discussed in Section 1.1, historically, medical decisions have been made by categorizing a patient into a well-known class of disease (the diagnostic step), and this step is usually performed by a human, perhaps with the aid of some data or diagnostic tools. Then, for each disease type, prognostic information is available via population statistics, and treatment selection options are again largely based on population statistics as approximated by a clinical trial: the probability that a given treatment will succeed is defined in terms of a population, not the individual. For many simple diseases, this paradigm has proved effective for thousands of years, but as we progress towards treating more complex, systemic conditions such as cancer, a more sophisticated approach to medical decision making is called for. It is certainly appreciated that medical decisions should always be patient-specific to whatever extent possible, and that complex diseases demand a more precise approach, but exactly how to implement personalized, precision medicine is not yet entirely clear. More extensive disease subtyping is one approach [64], whereby current diagnostic categories are further split into subcategories (the number of diagnostic bins is increased). Taken to its extreme, however, we arrive at the conclusion that each patient is entirely unique from every other patient. If the population statistics perspective is taken, this is unacceptable because we cannot make predictions with a sample of \( n = 1 \), namely we cannot estimate probability of success because we have no population to compare our patient to. In this work, we have outlined a strategy which defines patient-specific probability of treatment success via the objective function (4.1.5), which is defined in terms of the VPE. In other words, whereas a classical treatment would define probability of treatment success in terms of a real clinical trial, we have defined probability of treatment success in terms of a virtual clinical trial - the computation of (4.1.5) may require the application of a Monte Carlo method, where we sample \( \tilde{f}_1, \ldots, \tilde{f}_N \sim P \tilde{f} \); this would be called a virtual cohort. Testing the treatment on each patient in the virtual cohort
amounts to applying the model $\tilde{q}_j = \mathcal{M}(\tilde{f}_j)$; probability of treatment success is then approximated by

$$P(\zeta) \approx \frac{1}{N} \sum_{j=1}^{N} \chi_E(\mathcal{M}(\tilde{f}(\zeta))) = \frac{M}{N}$$

where $M$ is the number of successful treatments out of the $N$ virtual patients and $\chi_E(q)$ is the indicator function of the set $E$, i.e. $\chi_E(q) = 1$ if $q \in E$ and zero otherwise.

To illustrate, suppose that we wish to predict the size of a patient’s tumor over time. The tumor is parameterized by a scalar function $N(t)$ which gives the total number of tumor cells at time $t$ (in say, days). Assume that for our patient, we can estimate $N_0 = N(0)$, and we wish to predict $N(100)$. A purely statistical approach would, for instance, assume that we have access to a collection $\mathcal{N} = \{(N^{(1)}(t_{11}), N^{(1)}(t_{21})), \ldots, (N^{(m)}(t_{1m}), N^{(m)}(t_{2m}))\}$ of true tumor sizes for $m$ other patients at some times $t_{1j}, t_{2j}$; we could then fit a regression model (say) to predict $N(100)$ from $N_0$. Such a strategy requires no understanding or modeling of the physiology underlying tumor growth. A mechanistic or phenomenological model, on the other hand, will make use of the measured $N_0$ and some explicit growth model for $N(t)$, making minimal use of the dataset $\mathcal{N}$ if at all. For example, we could model $N(t)$ using the Gompertz ODE [109]:

$$\begin{cases}
\frac{d\tilde{N}}{dt} = -k \tilde{N} \log(\tilde{N}/\theta) \\
\tilde{N}(0) = N_0
\end{cases} \tag{4.1.7}$$

Solving (4.1.7) gives an explicit model,

$$\tilde{N}(t) = \mathcal{M}(N_0, k, \theta, t) = \theta \exp(\log(N_0/\theta) \exp(-kt)) \tag{4.1.8}$$

Note that we have used the tilde to indicate that (4.1.7) is a model of the true $N(t)$. Of course, being a model, (4.1.8) is subject to comparison to real tumor growth curves; assuming that a gold standard value for $N(t)$ can be established\footnote{which can be difficult, especially in humans}, one can compare the model value predicted by (4.1.8)
to the actual value, i.e. we can compute the error $\epsilon(t) = N(t) - \tilde{N}(t)$, and say that

$$N(t) = M(N_0, k, \theta, t) + \epsilon(t)$$

In this case, the virtual patient has been parameterized by $\tilde{f} = [N_0, k, \theta]$; these parameters can certainly be patient-specific – we have already assumed that $N_0$ is available, for instance from in vivo imaging. The parameters $k$ and $\theta$ may be more difficult to obtain from a single time point $t = 0$, however: $\theta$, for instance, is an asymptotic carrying capacity, which in principle can only be determined if we know $N(t)$ for large $t$ (which defeats the purpose of the model). The parameter $k$, on the other hand, could potentially be measured for a specific patient if multiple time points were known (say $N(0)$ and $N(1)$). An extensive discussion of parameter estimation in models such as (4.1.8) is given in [27]. If $k$ and $\theta$ turn out to be impractical to estimate for a particular patient, we could treat both as nuisance parameters, i.e. set $\tilde{f}_n = [k, \theta]$ and assume them to be random samples from a known population distribution, calibrated off-line. In this example, there is no control component $\tilde{f}_c$; we could modify the model (4.1.7) to incorporate the effect of a drug, say, for instance by considering

$$\frac{d\tilde{N}}{dt} = -k\tilde{N} \log(\tilde{N}/\theta) - \alpha C(t)$$

where $C(t)$ is a time-varying amount of drug and $\alpha$ is a drug susceptibility; the drug is presumably under the physician’s control, hence we would model $\tilde{f}_c \equiv C(t)$. By modifying $C(t)$, we can control the output of $\tilde{N}$.

This example illustrates one of the fundamental difficulties in mechanistic modeling: it is entirely possible that (4.1.7) cannot correctly predict $N(t)$ for all times $t$, no matter how accurately we estimate $N_0$, $\theta$ and $k$. In fact, this is observed for real tumors: the growth model (4.1.7) has been shown to be consistent with certain types of tumor growth, but not others [167]. The process of establishing how well a model prediction fits reality is referred to as validation in the computational science community [216]. We discuss spatially resolved variants of (4.1.7) in Section 4.2.4.

With a validated mechanistic model, we can potentially begin to make extremely precise state-
ments about an individual patient without resorting to population data. Of course, biological systems are immensely complex; there is not, at the time of this writing, a set of foundational mathematical principles that govern biological systems such as we have with fluids (Navier-Stokes), mechanical systems (Newton and Hamilton), electromagnetism (Maxwell), and quantum systems (the standard model and quantum electrodynamics). It is perhaps unreasonable to assume that such a model could even exist in principle: biological systems are inherently built out of complex, interrelated networks of processes, so any such model is either inherently multiscale, or only valid at one scale [76]. The best we can hope for, at the moment, is to develop reasonable models which are then validated experimentally. Experimental results can then inspire further refinement of the model, which then inspires further experiments, \textit{ad infinitum} until we are satisfied. Whether this process will eventually converge in biology is unclear. Thankfully, our goal is only to assess \textit{efficacy} of such methods, not to provide full model validation for complex physiological processes: the definition of a precise task (in our case, optimization of treatment efficacy) greatly narrows the scope of study.

Before proceeding, we briefly reiterate our suggested framework for \( \mathcal{M} \)-PMED. We suggest that a good approach consists of the following tasks:

1. Identification of Physiological Random Processes (PRPs), \( f \), which are suspected to influence outcomes in therapy;

2. Identification of clinically relevant quantities of interest \( q \) whose value, if known, would provide a useful marker of success, and a set of outcomes \( E_{\text{success}} \) which defines successful treatment with regard to the quantities \( q \);

3. Identification of the measurement processes by which the PRPs \( f \) can be measured for patient \( j \), leading to a statistical model for the data \( g|f_j \) collected for the individual patient;

4. Building of a Virtual Patient Ensemble model \( \tilde{f} \), which is an approximation of the PRPs \( f \) for patient \( j \), using the measured data \( g|f_j \) and possibly externally validated population models for the null component \( f_n \);
5. Building of mathematical models $\mathcal{M}$ which seek to predict $q$ given a virtual patient $\tilde{f}$, culminating in the fundamental relationship

$$q = \mathcal{M}(\tilde{f}) + \epsilon$$

6. Determination of the controllable parameters of the treatment, parameterized by a vector $\zeta$;

7. Determination of a figure of merit for treatment success; we postulate that the form

$$\mathcal{P}(\zeta) = \mathbb{P}[\tilde{q}(\zeta) \in E_{success}]$$

is sufficiently general, though other objective functions, for instance of the form $\mathcal{P}(\zeta) = \mathbb{E}[\phi(\tilde{q}(\zeta))]$, where $\phi(\cdot)$ is a convex function, may prove more tractable;

8. Optimization of the above objective functional over the class of admissible treatment parameters.

For the remainder of this chapter, we demonstrate the above methodology by presenting a spatial model for tumor growth and response to chemotherapy, identifying the relevant PRPs, biomarkers $q$, and a specific model $\mathcal{M}$. We then discuss how the parameters in this model can be treated as random processes, using the technology developed in Chapter 2. Then, using Chapter 3, we discuss how ECT imaging data can be used to create patient-specific random process models $\tilde{f}$ for those parameters, and discuss how to perform patient-specific Virtual Clinical Trials (VCTs) to test treatment response (i.e. to evaluate the objective function $\mathcal{P}$). We also discuss how the deterministic and statistical properties of the imaging system relate to the properties of the virtual patient ensemble, and hence the statistics of the model output $\tilde{q}$.
4.2 Mathematical Models in Cancer

4.2.1 The Hallmarks of Cancer and the problem of Heterogeneity

Cancer, in essence, is a breakdown of homeostatic cellular behavior, due primarily to genetic mutation [289]. In two papers, the first in 2000 and the second in 2011, Hanahan and Weinberg [123] [124] presented eight traits that are hallmarks of most if not all cancers. These hallmarks distinguish malignant cell growth from both normal and abnormal but non-malignant cell growth:

1. Cancers must be able to proliferate (i.e. grow) without external growth factors;
2. Cancers must evade growth suppressors, i.e. chemicals which would normally prevent abnormal growth;
3. Cancers must resist programmed cell death;
4. Cancers must replicate indefinitely;
5. Cancers must be capable of recruiting new blood vessels;
6. Cancers must invade surrounding tissue and metastasize (break away and reform away from the primary site);
7. Cancers must be able to reprogram energy metabolism (the way cells use nutrients);
8. Cancers must be able to avoid destruction by immune cells.

The interaction with the tumor microenvironment – that is, the surrounding tissues and physiological processes – is also appreciated as playing a major role [289].

Furthermore, each of these hallmarks is associated with a complex set of genetically regulated physiological processes [281]. For example, the state of oxygenation (as quantified by the partial pressure, pO$_2$, or the oxygen saturation SaO$_2$) within a tumor is a crucial factor in the development and treatment of cancer. In tumors, the protein HIF-1-α (hypoxia-inducible-factor 1-α) regulates metabolism and promotes angiogenesis in response to hypoxia. Hypoxia itself, on the other hand, leads to invasion and resistance to radiation or chemotherapy [289].
It is generally agreed in the literature that the spatial and genetic heterogeneity of these physiological processes plays a critical role in initiating the Hanahan-Weinberg hallmarks and determining their magnitudes \cite{39, 188, 214, 91}, but there has been less effort towards defining physiological heterogeneity in rigorous mathematical terms or describing the interactions between different heterogeneous physiological processes. The mathematical modeling tools discussed in this work, namely the random process models discussed in Chapter 2, provide a framework for filling this gap.

Without detailed information about the specific nature and degree of heterogeneity for a given patient, prediction of clinical outcomes is difficult. However, we have also provided, in Chapter 3, a theoretical framework for making patient-specific measurements of certain heterogeneous physiological processes via ECT imaging. There, we saw that measurement devices are inherently noisy and their data incomplete, and so argued that the estimation of patient-specific processes and parameters must be accompanied by quantification of uncertainty.

We make the fundamental modeling assumption that every spatially heterogeneous physiological process - in particular those processes associated with the Hanahan-Weinberg hallmarks - is described by a physiological random process, and the collection of all relevant spatiotemporal physiological processes is described by a vector random process \( \mathbf{f} \sim \mathcal{P}_f \). Each component of \( \mathbf{f} \), i.e. \( \mathbf{f} = [f^{(1)}, \ldots, f^{(k)}] \) corresponds to a spatiotemporal physiological trait, such as oxygen saturation, tumor cell density, and drug concentration. The goal of \( \mathcal{M} \)-PMED is then to learn as much as possible about our patient’s \( \mathbf{f} \), namely using imaging to construct a VPE \( \tilde{\mathbf{f}} \). This provides a statistically rigorous way to quantify uncertainties in patient-specific estimates of heterogeneous physiological processes and hence any clinically relevant decision parameters derived from such estimates via a model of the form \( \tilde{\mathbf{q}} = \mathcal{M}(\tilde{\mathbf{f}}) \). In this sense, we have provided a unique statistical framework for precision medicine in the context of spatiotemporal physiological quantities and noisy imaging data, where we define precision medicine as the general tactic of combining patient-specific data, mathematical and statistical modeling and computational techniques to assist in the clinical decision-making process; refer again to Figure 4.1.
4.2.2 Treatment Strategies, Virtual Patient Ensembles and Patient-Specific Efficacy Figures of Merit

There are three general strategies to treating cancer: surgical approaches (including both resection and so-called interventional approaches which are less invasive), radiation-based approaches (both external and internal), and drug-based approaches [130]. Drug-based approaches include chemotherapeutic agents, which seek to chemically destroy cells, immunotherapies, which seek to aid immune response to cancer, and molecular approaches which seek to interrupt biomolecular pathways, usually by binding to particular receptors to inhibit certain cancer-specific pathologies such as proliferation and angiogenesis [130].

No matter the treatment approach, if we aim to employ the $\mathcal{M}$-PMED paradigm as discussed above, it is necessary to develop figures of merit for the predicted efficacy of treatments which are defined in terms of an individual patient. As we discussed in the introduction to this chapter, a discrete view of disease, whereby patients are placed into diagnostic bins, ultimately leads to a population-level description of efficacy: a certain treatment is deemed effective if it can be shown to be effective in the (real) population which defines the disease bin, and patient-specific probability of treatment success is defined in terms of the population probability. This mindset leads to the current standard clinical trial methodology. In the $\mathcal{M}$-PMED paradigm, we have proposed that each patient gives rise to a virtual population of ‘similar’ patients who would have given rise to similar measured data – this is the Virtual Patient Ensemble or VPE. This virtual ensemble is then used to define patient-specific efficacy: a proposed treatment is deemed effective if, when tested on the virtual population using a virtual clinical trial, it is effective. For our purposes, it is the random process $\tilde{f} = [\tilde{f}_m, \tilde{f}_n, \tilde{f}_c]$ which represents the virtual patient population: this process has been constructed using patient-specific data (to form $\tilde{f}_m$), random process models or population data (to form $\tilde{f}_n$) and models of the proposed treatment (to form the ‘control’ $\tilde{f}_c$). From the virtual patient ensemble, we make patient-specific predictions using models of the form $\tilde{q} = \mathcal{M}(\tilde{f})$.

An efficacy figure of merit should incorporate tradeoffs between treatment success and deleterious effect, and should also incorporate uncertainties in the predicted effect of the treatment [21] [18]. We have already claimed that a very general class of figures of merit are given by probabilities
of the form

\[ P = \mathbb{P}(\tilde{q} \in E_{\text{success}}) \]  

(4.2.1)

where \( \tilde{q} = \mathcal{M}(\tilde{f}) \) is the predicted effect of the treatment using model \( \mathcal{M} \), and \( E_{\text{success}} \) is a set of outcomes which has been selected to define an efficacious treatment; this set could quantify certain clinical endpoints, for instance. The probability (4.2.1) is how we define patient-specific probability of treatment success.

4.2.3 Integrated Log-kill: A Quantity of Interest for Chemotherapy

To illustrate the efficacy figure of merit (4.2.1), we will now discuss a particular example where we identify some patient-specific parameters \( f \) and a quantity of interest \( q \). In the next section, we will discuss a particular model \( \mathcal{M} \) that predicts \( q \) from a VPE \( \tilde{f} \), and later, make the connection to patient-specific ECT imaging data.

We consider the case where \( q = [q_m, q_n] \) is a predicted effect of a chemotherapy drug on respectively malignant and normal cells. For \( q_m \), we will define a quantity of interest called the integrated log cell-kill. Suppose that \( n(r, t) \) is a (real) tumor cell density in a patient, where \( r \in V \subset \mathbb{R}^3 \) is a tissue region of interest. We assume as usual that \( n \sim \mathbb{P}_n \) is a random process, and that for each realization of \( n \) we have a \( V \) such that \( n \geq c > 0 \) in \( V \) (in other words, cell density is bounded away from zero in the tumor region). Then, we define the integrated log-kill after treatment time \( T \) to be the scalar

\[ q_m = \int_V \ln \left( \frac{n(r, T)}{n(r, 0)} \right) dr \]  

(4.2.2)

Note that we only integrate over the region where \( n \geq c > 0 \), so \( q_m \) is finite. We posit that (4.2.2) is a useful quantity of interest for chemotherapy, because it averages the change in local cell density due to a treatment. Note that if \( q_m < 0 \), we have – on average – \( n(r, T) < n(r, 0) \), so that the local cell density has been reduced. Of course, other scalar quantities of interest can also be derived from \( n(r, t) \): one could also consider the maximum \( q_m = \max_{r \in V} n(r, t) \), for instance. For the
predicted effect of the drug on normal tissue, we assume only that a convenient scalar $q_n$ has been
defined that measures the degree of normal tissue complication. Predicting normal tissue effects
for individual patients as a result of chemotherapy is challenging, so it may be the case that $q_n$
measures a simple clinical endpoint such as survival time or time until first side effect, and we may
need to resort to population statistics (‘big data’) in order to estimate the distribution of $q_n$.

Note that (4.2.2) is a random variable, because we have assumed $n$ to be a random process.
With these choices, we will define the set $E_{\text{success}}$ as having $q_m \leq k_m$ and $q_n \geq k_n$, with $k_m$ and
$k_n$ some constants. Note that a more negative log cell kill implies more cells are killed; it is thus
logical to have $k_m < 0$. We choose $q_n \geq k_n$ under the assumption that $q_n$ measures something like
survival time or time until first side effect (measured in years, say), so that we define success in this
case as having no visible normal-tissue complications in under $k_n$ years. This selection of $E_{\text{success}}$
is a modeling choice; we make this one for convenience.

Thus, $q \in E_{\text{success}}$ represents reaching a desired level of tumor control (as defined by $q_m \leq k_m$)
while maintaining an acceptable level of normal tissue damage (as defined by $q_n \geq k_n$). We will
use a model of the form $\tilde{q} = \mathcal{M}(\tilde{f})$ to predict the actual $q$ defined via (4.2.2) (recall that variables
with a tilde represent an estimated or predicted, while the un-tilded version is the actual idealized
‘ground truth’ value), using a combination of a mathematical model of tumor growth and response
to chemotherapy, measured ECT data for the patient, and a random process model for a nuisance
input $\tilde{f}_n$.

With a model $\tilde{q} = \mathcal{M}(\tilde{f})$ to predict (4.2.2), and a fixed time $T$ (usually taken to be the time
for a chemotherapy drug to clear the patient), we can define the probability of tumor control as

$$P_{\text{TC}} = \mathbb{P}[\tilde{q}_m \leq k_m]$$

and the probability of normal tissue complication as

$$P_{\text{NTC}} = \mathbb{P}[\tilde{q}_n \geq k_n]$$
Treating \( \tilde{q} = [\tilde{q}_m, \tilde{q}_n] \in \mathbb{R}^2 \) as a joint random vector, we can also define the joint efficacy probability

\[
P_E = P[\tilde{q} \in E_{\text{success}}], \quad E_{\text{success}} = (-\infty, k_m] \times [k_n \times \infty)
\]

(4.2.3)

### 4.2.4 Mathematical models of tumor growth and treatment response

There is an extensive literature on mathematical models for tumor growth going back at least 50 years (though, some common models have roots that go much deeper); see for example [294, 213, 245, 275, 239, 6, 187, 98, 302]. See also [167] for an accessible introduction to mathematical modeling in cancer.

While there are certainly other distinctions to be made, we immediately distinguish two kinds of tumor growth model: models which are spatially resolved, and models which are ‘lumped parameter’. The classical examples of the latter are the exponential, Gompertz, Logistic and von Bertalanffy models, and various coupled ODE models [167]. Spatially resolved models are discussed in e.g. [69] [228] [302].

For simplicity we will consider only avascular and premetastatic growth; incorporating both angiogenesis and metastasis requires more complex mathematical models; see e.g. [69].

The simplest models of tumor growth are based on ordinary differential equations for the time dependence of the total number of malignant cells in a tumor, denoted \( N(t) \). It is common to regard \( N(t) \) as a deterministic quantity that can be found by solving the differential equations, with no spatial inhomogeneity, uncertainty or stochastic effects considered. The differential equations may involve a small number of free parameters, which can be fixed by fitting the predicted behavior of \( N(t) \) to measurements on tumors in humans or animals.

These models typically take the generic form

\[
\frac{dN(t)}{dt} = \frac{dN(t)}{dt} \bigg|_{\text{growth}} + \frac{dN(t)}{dt} \bigg|_{\text{drug}},
\]

(4.2.4)

where the first term describes the growth of the tumor in the absence of a therapeutic intervention and the second accounts for the presence of a drug.

The simplest form for the growth term is \( dN(t)/dt|_{\text{growth}} = \beta N(t) \) (where \( \beta \) is a positive
constant), which leads to exponential growth, \( N(t) \propto \exp(\beta t) \). It is widely observed, however, that the growth rates of solid tumors decrease as the tumors get larger, and hence several common models take the form \( \frac{dN(t)}{dt}|_{\text{growth}} = N(t)\Phi[N(t)] \), where \( \Phi(\cdot) \) is a monotonically decreasing function. The resulting differential growth rates for several common models are shown in Figure 4.2. For example, the commonly used Gompertzian model assumes that the tumor size reaches a “carrying capacity” \( N_{\text{max}} \) limited by blood supply and nutrients. The Gompertz model enforces this condition by taking \( \Phi[N(t)] \propto -\ln(\frac{N(t)}{N_{\text{max}}}) \). Mechanistic explanations for Gompertzian tumor growth have been given by Gyllenberg [115, 116], Hahnfeldt [119] and Norton [213].

There is an extensive literature on the response of cancer cells to chemotherapy drugs, radiation or other interventions [140, 75, 99, 235, 248, 1, 195]. A common form for the drug effect in (4.2.4), derived from the classical law of mass action [210], assumes that the ‘reaction rate’ is proportional to the product of the concentrations, i.e. \( \frac{dN(t)}{dt}|_{\text{drug}} = -\alpha C(t)N(t) \), where \( C(t) \) is the drug concentration and \( \alpha \) is the positive rate constant. Equivalently,

\[
\frac{1}{N(t)} \left. \frac{dN(t)}{dt} \right|_{\text{drug}} = \frac{d}{dt} \ln N(t) = -\alpha C(t) . \tag{4.2.5}
\]
This form, often called *linear log-kill*, says that the fractional rate of cell killing is a linear function of the drug concentration and is usually attributed to Skipper [256]. Linear log-kill is a reasonable model for certain cell-cycle-independent drugs such as doxorubicin, at least *in vitro* [82]. More generally, cytotoxic drugs can affect cell populations non-linearly with respect to $C(t)$, for instance due to saturation effects and cell cycle inhomogeneities. We discuss several nonlinear models later.

With the general growth model $dN(t)/dt|_{\text{growth}} = N(t)\Phi[N(t)]$ and the mass-action assumption for the drug effect, the overall model (4.2.4) becomes

$$
\frac{d}{dt} \ln N(t) = \Phi[N(t)] - \alpha C(t). \tag{4.2.6}
$$

With the choice of Gompertzian growth kinetics,

$$
\frac{d}{dt} \ln N(t) = -\mu \ln \left( \frac{N(t)}{N_{\text{max}}} \right) - \alpha C(t). \tag{4.2.7}
$$

Equation (4.2.7), paired with an initial condition of $N(0) = N_0 \geq 0$ total cells, describes the deterministic evolution of $N(t)$. Again, neither spatial effects nor uncertainty are addressed in such a model. We will address both by replacing $N(t)$ with a corresponding random process $n(r,t)$.

We define a continuum density of cells $n(r,t)$, which has units of number of cells per unit volume. We treat $n(r,t)$ as a random process, and hence the total number of tumor cells $N(t)$ is also random; the two entities are related by

$$
N(t) = \int_{V(t)} n(r,t) \, dr, \tag{4.2.8}
$$

where $V(t)$ is the evolving tumor volume.

We postulate that a reasonable spatial counterpart of equation (4.2.6) is the following reaction-diffusion equation [167] [98] [89] [268] [151] [69]:

$$
\begin{aligned}
\frac{\partial n}{\partial t} &= \nabla \cdot (D_c(r) \nabla n(r,t)) + n(r,t)\Phi[n(r,t)] - \alpha(r)c(r,t)n(r,t) \\
n(r,0) &= n_0(r)
\end{aligned}, \tag{4.2.9}
$$
where $D_c(r)$ is a cell diffusion rate, $\alpha(r)$ and $c(r, t)$ are random processes representing, respectively, drug susceptibility and drug concentration. We also assume that the initial condition $n_0(r)$ is a random process. The growth nonlinearity $\Phi$, which usually takes one of the same forms as in the scalar case, may also depend on some additional random process parameters; for example, with a Gompertzian model, we would have

$$\Phi[n(r, t)] = -\mu(r) \ln \left( \frac{n(r, t)}{n_{\text{max}}(r)} \right)$$

where the growth rate $\mu(r)$ and carrying capacity $n_{\text{max}}(r)$ are allowed to be spatial random processes (we assume these to be independent of time, but this assumption could of course be generalized). With this assumption and vector notation, the reaction-diffusion model (4.2.9) becomes

$$\begin{cases}
\frac{\partial n}{\partial t} = D_c n - \mu \ln \left( \frac{n}{n_{\text{max}}} \right) n - \alpha c n \\
n(0) = n_0
\end{cases}
\quad (4.2.10)$$

where $D_c = \nabla \cdot D_c(r) \nabla$ is the diffusion operator.

Because each function in (4.2.9) and (4.2.10) is treated as a random quantity, we must specify the sense in which the equality holds. We will assume throughout that a differential equation such as (4.2.9) holds in the weak (in the sense of generalized functions) sense for each fixed realization of the random parameters. This assumption allows us to solve (4.2.9) using standard differential equation techniques, then derive statistical distributions for the resulting solution and any quantities of interest related to it. Such an equation is frequently called a \textit{random} differential equation, to distinguish from \textit{stochastic} differential equations which require more sophisticated solution strategies such as the Itô formalism (see discussion in [257], section 4.7).

Assuming that the equation (4.2.9) is well-posed, we can solve it for $n(r, t)$; this results in a model of the form

$$\tilde{n} \equiv \tilde{n}(r, t) = \mathcal{G}(n_0, D_c, \alpha, c, \mu, n_{\text{max}}) = \mathcal{G}(\tilde{f}).
\quad (4.2.11)$$
Figure 4.3: A simulation of Gompertzian growth of the spatial random field $n(r,t)$, as modeled in equation (4.2.10) with $c(r, t) \equiv 0$, is displayed on the left. In this simulation, both the local growth constant $\mu(r)$ and the local carrying capacity $n_{\text{max}}(r)$ are spatially constant but random; the initial condition $n(r,0)$ is taken as a lumpy background random process, and the cell diffusion rate $D_c$ is taken to be zero. Three realizations (rows) are shown at three times. The total cell number $N(t)$ as defined in (4.2.8) is shown on the right for 16 realizations of the process.

where the virtual patient $\tilde{f}$ is defined by

$$\tilde{f} = [n_0, D_c, \alpha, c, \mu, n_{\text{max}}]$$  \hfill (4.2.12)

The growth model (4.2.11) illustrates that the solution $\tilde{n}(r,t)$ to (4.2.9) depends on all the parameters that define the model; because all of these parameters are treated as random processes, the output $\tilde{n}$ is random as well. Thus, the integrated log kill (4.2.2) can be estimated as

$$\tilde{q}_m = \mathcal{M}(\tilde{f}) = \int_V \ln \left( \frac{\tilde{n}(r,T)}{\tilde{n}_0(r)} \right) \, dr$$  \hfill (4.2.13)

where $\tilde{n}(r,t)$ is given by (4.2.11).

The components of (4.2.12) that are measurement components ($\tilde{f}_m$) and null components ($\tilde{f}_n$) will depend on the available imaging data; we will argue in the next section that both $\alpha$ and $c$ are at least partially measurable in vivo. The drug concentration $c$ can also be treated as a control
component \((\tilde{f}_c)\) so that we can maximize predicted efficacy.

In the next section, we will show that a simplification of the model \((4.2.11)\) leads to an estimate of \((4.2.13)\) that only makes use of only two spatial random processes, both of which we propose can be measured \textit{in vivo} using ECT imaging data. We discuss how to use such images to estimate integrated log-kill in Section 4.3.

### 4.2.5 A Model for Computing Patient-Specific Integrated Log-kill

Chemotherapy drugs remain in a patient’s circulation for a few hours or days at most, and it is reasonable to assume that the tumor growth rate is negligible over this period. If we begin with the reaction-diffusion model \((4.2.9)\) but ignore the diffusion and growth terms (more rigorously, we could perform a timescale analysis), we have the simplified model:

\[
\begin{cases}
\frac{\partial \tilde{n}}{\partial t} = -\alpha(r)c(r,t)\tilde{n}(r,t) \\
\tilde{n}(r,0) = \tilde{n}_0(r)
\end{cases}
\quad (4.2.14)
\]

The equation \((4.2.14)\) has a convenient explicit solution given by

\[
\tilde{n}(r,T) = \tilde{n}_0(r) \exp \left( -\alpha(r) \int_0^T c(r,t') \, dt' \right) = \tilde{n}_0(r) \exp \left( -\alpha(r) \beta(r;T) \right)
\quad (4.2.15)
\]

where

\[
\beta(r;T) = \int_0^T c(r,t') \, dt'
\]

is the drug area-under-the-curve or AUC. Thus, dividing both sides of \((4.2.14)\) by \(\tilde{n}_0(r)\), taking the logarithm and integrating over the volume \(V\), the solution \((4.2.15)\) gives a model for the integrated log kill \((4.2.2)\):

\[
\tilde{q}_m = M(\tilde{f}) = \int_V \ln \left( \frac{\tilde{n}(r,T)}{\tilde{n}_0(r)} \right) \, dr = -\int_V \alpha(r) \beta(r;T) \, dr
\quad (4.2.16)
\]
Note that if $T$ is large enough for complete drug clearance from the tumor, $\beta(r; T) \equiv \beta(r)$ is independent of $t$ and can be interpreted as the total exposure of tumor tissue at point $r$ to the drug. We have thus expressed a model for the integrated log kill $q_m$ in terms of two random processes, $\alpha(r)$ and $\beta(r)$, and this model takes the form of an $L^2(V)$ inner product:

$$\tilde{q}_m = -(\alpha, \beta)_{L^2(V)}$$ \hspace{1cm} (4.2.17)

Because both $\alpha$ and $\beta$ are random processes, $\tilde{q}_m$ is a random variable (ref Chapter 2). In order to compute the patient-specific probability of success $\tilde{q}_m$, we require a statistical description of $\tilde{q}_m$ in the form of a PDF, probability distribution or characteristic function. Indeed, the statistics of the random variable $q_m$ for a specified drug distribution $c$ can be derived from the conditional characteristic function, which is given in terms of the characteristic functional of $\alpha$ (refer to Section 2.2.8) by

$$\psi_{\tilde{q}_m|c}(\xi) = \left\langle \exp \left( -2\pi i \xi \tilde{q}_m \right) \right\rangle_{\tilde{q}_m|c} = \left\langle \exp \left( 2\pi i \xi \int_V \alpha(r) \beta(r) \, dr \right) \right\rangle_{\alpha} = \Psi_{\alpha}[ -\xi \beta(r)] .$$ \hspace{1cm} (4.2.18)

We can then write the conditional PDF for $\tilde{q}_m$ as

$$p_{\tilde{q}_m|c}(x) = \int_{-\infty}^{\infty} \psi_{\tilde{q}_m|c}(\xi) \exp(2\pi i \xi x) \, d\xi = \int_{-\infty}^{\infty} \Psi_{\alpha}[ -\xi \beta(r)] \exp(2\pi i \xi x) \, d\xi ,$$ \hspace{1cm} (4.2.19)

This equation relates the time-integrated drug distribution ($\beta(r)$) to the estimated integrated log-kill $\tilde{q}_m$ via the spatially varying statistics of the cellular sensitivity to the drug ($\alpha(r)$). Because $p_{\tilde{q}_m|c}(x)$ is conditional on $c$, (4.2.19) is directly applicable only when the drug distribution is known exactly, but its extension to the practical clinical case where the drug distribution is either unknown and random or estimated from noisy molecular imaging data is discussed later; we claim that $\beta$ can be measured in vivo using molecular imaging, making the estimation of the quantity of interest $\tilde{q}_m$ for a particular patient practical if a characteristic functional $\Psi_{\alpha}[\phi]$ for the drug sensitivity is known. We claim that this characteristic functional can also be determined using molecular
imaging, though perhaps not for a specific patient; it may need to be estimated from population statistics by fitting one of the models discussed in Section 2.6 to a calibration database [168].

With (4.2.19) in hand, we can compute the efficacy of a drug treatment in terms of degree of short-term tumor control via a probability of the form (4.2.3). If there is successful death of tumor cells immediately following therapy, \( q_m \) is a negative number, so as discussed above, we define tumor control as achieving a \( q_m \) that is less than some chosen threshold \( k_m \) (e.g., \( k_m = -3 \) for tumor reduction by a factor of \( e^3 \approx 20 \)); similarly, we define normal tissue complication as having a \( q_n \) that is greater than some chosen threshold \( k_n \). The conditional probability of tumor control for a specified drug concentration is then given by

\[
P_{TC|c} = \mathbb{P}(q_m \leq k_m | c) = \int_{-\infty}^{k_m} p\tilde{q}_m(x) \, dx = \int_{-\infty}^{k_m} \int_{-\infty}^{\infty} \Psi_{\alpha} [-\xi \beta(r)] \exp (2\pi i \xi x) \, d\xi dx,
\]  

(4.2.20)

Both of these integrals are one-dimensional, so they are readily performed numerically if both \( c(r,t) \) and an analytic form for \( \Psi_{\alpha} [\phi] \) are known.

To extend the result (4.2.20) formally to an ensemble of drug distributions, say \( c \sim \mathbb{P}_c \), one can perform an average over \( c \) using the law of total expectation (equation (2.1.30)):

\[
P_{TC} = \langle P_{TC|c} \rangle_c = \int_{-\infty}^{k_m} \int_{-\infty}^{\infty} \langle \Psi_{\alpha} [-\xi \beta(r)] \rangle_c \exp (2\pi i \xi x) \, d\xi dx.
\]  

(4.2.21)

To put these results into the context of clinical chemotherapy optimization, we assume that mass \( M \) of the drug has been administered to a patient. Within the confines of the linear model for the drug distribution used thus far, \( c(r,t) \) and hence the drug AUC \( \beta(r) \) are proportional to \( M \), so we can define \( \beta(r) = M \cdot \beta_1(r) \), where \( \beta_1(r) \) is the drug AUC for \( M = 1 \). Then we can write

\[
P_{TC|c}(M) = \mathbb{P}(q_m \leq k_m | c, M) = \int_{-\infty}^{k_m} \int_{-\infty}^{\infty} \Psi_{\alpha} [-\xi M \cdot \beta_1(r)] \exp (2\pi i \xi x) \, d\xi dx.
\]  

(4.2.22)

Assuming that a similar model for \( P_{NTC|c}(M) \) for an administered mass \( M \) is available, we can thus perform optimization over \( M \) to find the optimal injected mass; by plotting \( P_{TC|c}(M) \) versus \( P_{NTC|c}(M) \) for different values of \( M \) results in the Therapy Operating Characteristic or TOC Curve [18], as shown in Figure 4.4. The TOC curve provides a graphical display of the tradeoff between
probability of tumor control and probability of complications.

Figure 4.4: An example TOC curve, which is a graphical display of the tradeoff between probability of tumor control and probability of normal tissue complication.

If multiple treatment parameters are available, for instance treatment schedules, we can parameterize the drug distribution $c$ and hence the function $\beta$ with a vector $\zeta$ as discussed in (4.1.6).

If we define the successful treatment set as $E_{\text{success}} = (-\infty, k_m] \times [k_n, \infty)$ as in (4.2.3), we can also measure the joint probability of successful treatment,

$$\mathcal{P} (\zeta) = \mathbb{P} [\tilde{q}(\zeta) \in E_{\text{success}}]$$

where $\tilde{q} = [\tilde{q}_m, \tilde{q}_n]$.

4.2.6 Other treatment response models

The formulas for tumor response developed above are all linear, in the sense that the log-kill is a linear function of the drug concentration, and local, in the sense that the response of the tumor at some point $r$ depends on the drug concentration only at that point. There are situations, however, where the response can be nonlinear, nonlocal or both.
Nonlocal processes with negligible time delay

As an example of nonlocal drug action, consider targeted radionuclide therapy, which uses radioactive drugs that bind to a specific receptor on tumor cells. While bound, the radioisotope in the drug undergoes radioactive decay, emitting gamma rays and charged particles. Depending on the isotope, the charged particles might include beta particles (high-energy electrons), positrons, conversion electrons or alpha particles. The energetic radiations kill cancer (and other) cells by damaging DNA, but for present purposes all we need to know is that the damage can be at some distance from the cell to which the drug molecule is bound; the range is tens of microns for alpha particles in tissue, millimeters for beta particles and positrons, and centimeters for gamma rays.

Because of the speed of the nuclear decay products, there is negligible time delay between the decay and the DNA damage. In that case the general linear but nonlocal model is obtained by the replacement

\[ \tilde{q}_m = -\int_V \alpha(r) \beta(r) dr \rightarrow -\int_V \alpha(r) \int_V \ell(r, r') \beta(r') dr' dr \]

In scalar-product form, the righthand side is \(-\langle \alpha, \mathcal{L} \beta \rangle\), where \(\mathcal{L}\) is a linear integral operator with kernel \(\ell(r, r')\).

We can thus retrace the steps to (4.2.18) and write

\[ \psi_{\tilde{q}_m|c}(\xi) = \Psi_{\alpha|c} \{ -\xi [\mathcal{L} \beta] (r) \} . \quad (4.2.23) \]

where \([\mathcal{L} \beta] (r) = \int_V \ell(r, r') \beta(r') dr'\). We then then proceed to derive \(p_{\tilde{q}|c}(x)\) in the same manner, and compute the probability of tumor control \(\mathcal{P}_{TC}\).

Drug distribution at the microscale

So far we have considered the drug distribution \(c(r, t)\) within a tumor to be a general random process. A more detailed treatment considers that the drug molecules may be in the capillaries, diffusing in the interstitial space, bound to surface receptors or internalized into the cytoplasm of malignant cells [143], so that the drug distribution can be decomposed as [199] [18].
\[ c(r, t) = c^{\text{cap}}(r, t) + c^{\text{diff}}(r, t) + c^{\text{bound}}(r, t) + c^{\text{int}}(r, t). \] 

(4.2.24)

See Figure 4.5.

Figure 4.5: The decomposition of a drug concentration into capillary, diffusing, bound and internalized components [137] [11] [143]

The first two terms in the concentration decomposition (4.2.24) can be linked by the time-dependent diffusion equation for an inhomogeneous medium, given by [18] [142] [251]

\[ \frac{\partial}{\partial t} c^{\text{diff}}(r, t) - \nabla \cdot [D(r) \nabla c^{\text{diff}}(r, t)] = s(r, t), \] 

(4.2.25)

where \( D(r) \) is the diffusion coefficient and \( s(r, t) \) is the source of the diffusing species (e.g. drug molecules). For our purposes we will treat both \( D(r) \) and \( s(r, t) \) as random processes, and the initial condition \( c^{\text{diff}}(r, 0) \) is taken to be zero.

In many cases drug molecules escape from tumor capillaries through small pores called fenestrations, typically 60-80 nm diameter. Most drug molecules are a few nm across, so they can easily escape from the capillaries, as can antibodies, which are 10-15 nm. Because the pores are small compared to other relevant dimensions in a tumor, emission of drug molecules can be modeled as
a spatiotemporal point process, with one sample of the random process \( s(r, t) \) being given by

\[
s(r, t) = \sum_{j=1}^{J} \delta(r - r_j) \delta(t - t_j),
\]

(4.2.26)

where the \( j \)th drug molecule \((j = 1, ..., J)\) escapes into the extravascular space at point \( r = r_j \) and time \( t = t_j \). The units of \( s(r, t) \) are drug molecules per unit volume per unit time. If we assume that the emissions satisfy the Poisson postulates outlined in section [3.4.3], then \( s(r, t) \) is a Poisson point process. The characteristic functional of a spatiotemporal Poisson point process is fully determined by its mean function \( \bar{s}(r, t) \) (see Section [2.6.2])

\[
\Psi_{s|\bar{s}}[\phi] = \exp \left[ \int_V \int_0^T \bar{s}(r, t) \left( e^{-2\pi i \phi(r, t)} - 1 \right) dt \, dr \right]
\]

(4.2.27)

where \( T \) is the time interval over which the emissions can occur. The notation \( \Psi_{s|\bar{s}}[\phi] \) indicates that \( \bar{s}(r, t) \) is held constant in the average over the sets \( \{r_j\} \) and \( \{t_j\} \); if we allow \( \tilde{s} \) to be a random process, the result would be a Cox point process (see Section [2.6.2]).

Physiologically, \( \bar{s}(r, t) \) depends on the concentration of the drug in the capillaries \( c_{\text{cap}}(r, t) \) and the vascular permeability \( v(r) \). If there are no nonlinearities in the secretion process, this dependence is a simple product, \( \bar{s}(r, t) = c_{\text{cap}}(r, t)v(r) \), and we can write

\[
\Psi_{s|c_{\text{cap}},v}[\phi] = \exp \left[ \int_V \int_0^T c_{\text{cap}}(r, t)v(r) \left( e^{-2\pi i \phi(r, t)} - 1 \right) dt \, dr \right]
\]

(4.2.28)

If \( c_{\text{cap}} \) and \( v \) are also treated as random processes, we must average over them to obtain the overall characteristic functional for the source term:

\[
\Psi_{s}[\phi] = \left\langle \exp \left[ \int_V \int_0^T c_{\text{cap}}(r, t)v(r) \left( e^{-2\pi i \phi(r, t)} - 1 \right) dt \, dr \right] \right\rangle_{c_{\text{cap}}} \bigg|_v .
\]

(4.2.29)

These two averages can be approximated with sample averages if one has a constructive model for the tumor capillaries (e.g., the fractal model of Baish and Jain [13] or the Anderson-Chaplain model [1]).
We can write the diffusion equation above in vector-space form as

$$\frac{\partial c^{\text{diff}}(t)}{\partial t} + Dc^{\text{diff}}(t) = s(t), \quad (4.2.30)$$

where

$$D \equiv -\nabla \cdot D \nabla. \quad (4.2.31)$$

It can be shown that the characteristic functional for the diffusing component of the drug is related to the characteristic functional of the source term by

$$\Psi_{c^{\text{diff}}}(\phi, t) = \left\langle \Psi_s \left[ \int_0^t \exp[(t' - t)D^\dagger] \phi \, dt \right] \right\rangle_D, \quad (4.2.32)$$

where \(\exp(tD^\dagger)\) is the operator exponential of \(D^\dagger\) (equivalently, is the adjoint of the operator exponential \(\exp(tD)\)) [226]. The expectation in (4.2.32) is over diffusion coefficients \(D\); performing this expectation may require a Monte Carlo numerical method such as that considered in [61] because of the nonlinear dependence on \(D\); see also Section 5.2 (Future Work).

**Binding to receptors in targeted therapy**

To link \(c^{\text{diff}}\) to \(c^{\text{bound}}\), we consider ligand-receptor binding kinetics [210] [167]. Binding of a ligand to a receptor is conventionally parameterized by the receptor concentration \(B_{\text{max}}\), the dissociation constant \(K_d\) and the binding potential \(BP = B_{\text{max}}/K_d\) [201]. At tracer levels \(BP\) is the ratio of specifically bound ligand concentration to free concentration. In the usual approach where the concentrations are not considered to vary with position and are not treated as random, the equilibrium concentration of bound ligands is given by the Michaelis-Menten kinetic equation:

$$C^{\text{bound}} = \frac{B_{\text{max}} C^{\text{free}}}{K_d + C^{\text{free}}}. \quad (4.2.33)$$

To adapt this result to the formalism of this work, we interpret the receptor density \(B_{\text{max}}\) as the random density of tumor cells \(n(\mathbf{r}, t)\) times the random number of receptors per cell, \(N_{\text{rec}}\), and we can identify \(C^{\text{free}}\) as the random process \(c^{\text{diff}}(\mathbf{r}, t)\). If we assume that \(c^{\text{diff}}(\mathbf{r}, t)\) varies sufficiently
slowly that dynamic equilibrium is maintained at each point, we can write:

\[
c_{\text{bound}}(\mathbf{r}, t) = n(\mathbf{r}, t)N_{\text{rec}} \frac{c_{\text{diff}}(\mathbf{r}, t)}{K_d + c_{\text{diff}}(\mathbf{r}, t)}. \tag{4.2.34}
\]

Because tumor cells are genetically very similar (probably all clones of a single parent cell), a reasonable statistical model for the number of receptors per cell is that it is a Poisson random variable with mean \(\bar{N}_{\text{rec}}\), which is independent of \(\mathbf{r}\). It may also be valid to assume that \(K_d\) is the same for all cancer cells with receptors for a particular targeted drug. In that case \(K_d\) can be taken as nonrandom and independent of \(\mathbf{r}\).

In clinical practice it is useful to maintain \(c_{\text{diff}}(\mathbf{r}, t) \ll K_d\) so that the binding potential is small; otherwise the drug molecules may all be bound on a thin layer of cells at the periphery of a large tumor, leaving few to treat malignant cells in the interior; Weinstein at al. refer to this effect as the binding-site barrier \([290]\). It is avoided in the weak-binding limit, where we linearize \((4.2.34)\) about zero to write

\[
c_{\text{bound}}(\mathbf{r}, t) \approx \frac{n(\mathbf{r}, t)N_{\text{rec}}}{K_d} c_{\text{diff}}(\mathbf{r}, t). \tag{4.2.35}
\]

With this approximation, \(c_{\text{bound}}\) is proportional to \(c_{\text{diff}}\), which in turn is a linear transform of the source \(s\) in the diffusion equation. The corresponding characteristic functional is

\[
\Psi_{c_{\text{bound}}} (\phi) = \left\langle \Psi_{c_{\text{diff}}} \left[ \frac{N_{\text{rec}}}{K_d} n(\phi) \right] \right\rangle_{N_{\text{rec}}, n}. \tag{4.2.36}
\]

**Internalization**

A general term for internalization is endocytosis. Specific mechanisms include phagocytosis, pinocytosis, receptor-mediated endocytosis and clathrin-mediated endocytosis. For a review, see \([81]\).

A simple model of endocytosis presented by Wiley in 1982 \([297]\) defines an endocytotic rate
constant $K_e$ through the equation (in our notation),

$$\frac{dC^{int}}{dt} = K_e C^{bound}. \tag{4.2.37}$$

If we replace the overall concentrations with the corresponding random processes and integrate over time, assuming a chemotherapy administration at $t = 0$, we obtain

$$c^{int}(r, t) = K_e \int_0^t c^{bound}(r, t') dt'. \tag{4.2.38}$$

With (4.2.34), we have

$$c^{int}(r, t) = K_e N_{rec} \int_0^t n(r, t') \frac{c^{diff}(r, t')}{K_d + c^{diff}(r, t')} dt'. \tag{4.2.39}$$

The events that occur after internalization of a cytotoxic drug are complex, and mathematical models are largely lacking. For a qualitative review, see [193].

Incorporating the models discussed above into the framework for predicting treatment response is a subject of future work; see Section 5.2. We now discuss the application of the integrated log-kill model to precision medicine by incorporating patient-specific ECT data.

### 4.3 Application to Precision Cancer Therapy: Incorporation of Patient-Specific Data

Information about the drug distribution for a particular patient can be obtained by using radiolabelled drug molecules that can be imaged with ECT. The sensitivity of these imaging modalities is high, so the drug can probably be administered at subtherapeutic levels, but if this is a worry, then a surrogate molecule with approximately the same molecular weight as the drug but with no therapeutic effect can be used to study the extravasation of the drug from the capillaries (the source term) and its diffusion through the extracellular space. For a bolus injection the extravasation and diffusion processes have very different time dependences (the former being more rapid), so in principle we can separate $c^{cap}(r, t)$ from $c^{diff}(r, t)$ on the basis of dynamic PET or SPECT
images, for either the actual drug or the surrogate.

Assume we have used a radiolabeled drug with long half life to observe the overall drug concentration \( c_j(r, t) \) for patient \( j \) over a long time period \( T \). As we discussed in Chapter 3, imaging data \( g \) and the radiolabeled drug \( c \) form a pair of statistical covariates \( [c, g] \sim P_{c,g} \). For patient \( j \), the raw imaging data \( g_j = g|c_j \) is a statistical quantity whose distribution is indirectly related to \( c_j \) via a linear operator \( \mathcal{H}_c : L^2(V) \to \mathcal{Y} \), where \( \mathcal{Y} \) is either \( \mathbb{R}^M \) or \( L^2(A) \) as discussed in Chapter 3. Recall that in both the photon-counting and photon-processing ECT case, we can model \( g|c_j \) as a sample from \( \text{Poi}(\mathcal{H}f) \), i.e.

\[
g|c_j \sim \text{Poi}(\mathcal{H}c_j) \quad (4.3.1)
\]

The explicit expressions for the distribution of \( g|c_j \) are given in Chapter 3. As we discussed there, the imaging system also induces a decomposition of \( c_j \) into measurement and null components:

\[
c_j = c_{m,j} + c_{n,j}
\]

One can then perform statistical estimation to reconstruct \( c_{m,j} \); such an estimate is denoted generically by a hat. In molecular imaging, the most common reconstruction methods are variants on classical Maximum Likelihood (ML) estimation, that is we choose \( \hat{c}_j \) to be (at least approximately) a maximizer of \( \ell(c|g) \) (see Chapter 3 or [26]). For instance, from \( g|c_j \), an estimate \( \hat{c}_j \) can be formed by assuming a voxelized approximation for \( c \), then using the EM algorithm to perform approximate maximum likelihood estimation of the voxel values. We can also assume a Bayesian formulation, in which we take \( c_j \) to be distributed according to the posterior \( P_{c|g} \). In either instance, the resulting object can be taken as the patient-specific virtual ensemble \( \tilde{c}_j \).

Suppose further that we have used some tracer to measure drug sensitivity \( \alpha(r) \). The data for this study are denoted \( g|\alpha_j \), and the linear operator for this step is denoted \( \mathcal{H}_\alpha \). We can use the data sets \( g|c_j \) and \( g|\alpha_j \) to perform ML reconstructions of (the measurement components of) the drug distribution and sensitivity, denoting the results as results as \( \hat{c}_{j,m} \) and \( \hat{\alpha}_{j,n} \), respectively. Note that as discussed in both \( \hat{c}_{j,m} \) and \( \hat{\alpha}_{j,m} \) can now be considered as patient-specific physiological
random processes, that is, we can form a measurement VPE as

\[ \tilde{f}_m = [\tilde{\alpha}_{j,m}, \tilde{c}_{j,m}] \]

For the null components of both \( \alpha \) and \( c \), we assume that a random process model has been chosen. In some circumstances, the null component will not contribute significantly to the model output \( \tilde{q} \); this is the case in the integrated log kill model (4.2.17) if either \( \alpha \) or \( \beta \) is very smooth, for instance as a result of diffusion.

Note that, as we emphasize in Chapter 3, the randomness in \( \tilde{f}_m \) now is a result of the noise in the imaging system (which is well characterized for emission imaging). The statistical properties of \( \tilde{c}_j \) and \( \tilde{\alpha}_j \) are related to the estimation procedure used to construct them. In particular, if Maximum Likelihood estimation with the iterative EM algorithm is employed [78, 79], we can use a combination of asymptotic statistics and Monte Carlo simulation to compute the properties of these estimates [24]. In particular, in the asymptotic regime, \( \tilde{c} \) and \( \tilde{\alpha} \) with both be accurately modeled by Gaussian random processes (see Section 2.6 and Chapter 3).

We will now employ the proposed treatment response model (equation (4.2.17)), with and the patient-specific PRPs \( \tilde{c} \) and \( \tilde{\alpha} \) in place of the generic PRPs \( c \) and \( \alpha \) in order to personalize the parameter \( q_m \) and hence the probability of tumor control \( P_{TC} \) given in . Plugging \( \tilde{\alpha}_j \) and \( \tilde{c}_j \) into (4.2.17), we obtain

\[ \tilde{q}_{m,j} = -\int_V \tilde{\alpha}_j(r) \tilde{\beta}_j(r) \, dr \]  \quad (4.3.2)

where \( \tilde{\beta}_j(r) \) is the area under the curve of the estimated drug concentration at point \( r \) vs. time:

\[ \tilde{\beta}_j(r) \equiv \int_0^T dt \, \tilde{c}_j(r,t) . \]  \quad (4.3.3)

Note that as (4.3.2) is a functional of an approximate maximum likelihood estimate, it is guaranteed to be asymptotically normal by an application of the (functional) delta method; see [277, 300] for a technical discussion. One can also interpret (4.3.2) as an application of the maximum likelihood invariance theorem [26], but the pair of functions \( [c, \alpha] \) must be considered infinite-dimensional nuisance parameters in this context, and so the definition of the likelihood for the parameter \( q \)
requires care 222.

To estimate patient-specific $P_{TC}$ using the definition $P_{TC} = P(\tilde{q}_m \leq k_m)$, we require the PDF of the scalar random variable $\tilde{q}_j$; as just discussed, this PDF is approximately Gaussian. If we can assume that $\hat{\alpha}_j$ and $\hat{c}_j$ are unbiased estimates of $\alpha_j$ and $c_j$, the mean of $\tilde{q}_j$ is the true value $\tilde{q}_j$, that is,

$$\langle \tilde{q}_j \rangle_{\tilde{q}_j \mid \alpha_j} = \tilde{q}_j = -\int_V \alpha_j(r) \beta_j(r) \, dr$$  \hfill (4.3.4)

In current practice, $\hat{\alpha}_j$ and $\hat{c}_j$ will usually be obtained by an iterative algorithm called MLEM (Maximum-Likelihood Expectation Maximization), which returns voxelized approximations to the estimates of the functions $\alpha_j(r)$ and $c_j(r, t)$, respectively. The image data will also be sampled in time with a resolution $\Delta t$.

With these interpretations and the properties of ML estimators (including the statistical independence of the two ECT data sets), we can show that the variance of $\tilde{q}_j$ is approximately given by

$$\text{Var}(\tilde{q}_j) \approx \epsilon^3 \Delta t \sum_{n,n'} \sum_{m,m'} K_{\hat{\alpha}_j \mid \alpha_j}(r_n, r_{n'}) K_{\hat{c}_j \mid c_j}(r_n, t_m; r_{n'}, t_{m'}) ,$$  \hfill (4.3.5)

where $\epsilon^3$ is the volume of a voxel, $\Delta t$ is the time between images in a dynamic sequence, and $K_{\hat{\alpha}_j \mid \alpha_j}$ and $K_{\hat{c}_j \mid c_j}(r_n, t_m; r_{n'}, t_{m'})$ are elements of the covariance matrices for the estimates of the drug sensitivity and drug concentration, respectively. An efficient algorithm for computing these covariance matrices is given in 24.

We can thus assume that $\tilde{q}_j$ is approximately normal with mean 4.3.4 and variance 4.3.5, meaning that an estimate of the tumor control probability for patient $j$ is given by

$$\hat{P}_j = \int_{-\infty}^{k_m} p(\tilde{q}_j) \, d\tilde{q}_j ,$$  \hfill (4.3.6)

Note that the mean and variance of $\tilde{q}_j$ are unknown; their values may be replaced by their sample equivalents, then confidence intervals constructed by standard means. With 4.3.5 and 4.3.6, the estimated TCP for patient $j$ is an error function as in 18.
Chapter 5

Conclusions and Future Work

5.1 Conclusions and Summary of Contributions

In this work, we have provided a mathematical framework for the analysis of uncertainties in imaging- and model-based precision medicine. We have defined model-based precision medicine as the usage of an \textit{in silico} virtual patient $\tilde{f}$ and a mathematical model $\mathcal{M}$ which predicts treatment efficacy quantities of interest (biomarkers) via $\tilde{q} = \mathcal{M}(\tilde{f})$. The virtual patient $\tilde{f}$ is built using patient-specific imaging data, and we account for uncertainties due to noisy and incomplete data by defining a decomposition of $\tilde{f}$ into a measurement component $\tilde{f}_m$, a null, nuisance or noise component $\tilde{f}_n$, and a control component $\tilde{f}_c$. We then defined the notion of a \textit{Virtual Patient Ensemble}, or VPE, which models the virtual patient $\tilde{f}$ as a sample from a generalized random vector with distribution $\mathbb{P}_{\tilde{f}}$ or characteristic functional $\Psi_{\tilde{f}}$. Then, we provided a rigorous definition of patient-specific probability of treatment success via $\mathcal{P} = \mathbb{P}[\tilde{q} \in E_{success}]$; this probability is defined in terms of the virtual patient ensemble $\tilde{f}$, and such a probability can be computed via a patient-specific virtual clinical trial. These definitions frame model-based precision medicine in a rigorous probabilistic setting. To optimize a treatment for a specific patient, we proposed to parameterize the treatment by a vector $\zeta$, then optimize the objective function $\mathcal{P}(\zeta) = \mathbb{P}[\tilde{q}(\zeta) \in E_{success}]$.

To our knowledge, the definitions provided here of model-based precision medicine, patient-specific virtual patient ensemble and the decomposition of objects into measurement, null and
control components, the definition of patient-specific probability of treatment success and patient-specific virtual clinical trial all constitute novel contributions to the precision medicine and mathematical oncology literature. While extensive previous work (for example \cite{302} \cite{303} \cite{191} \cite{77} \cite{76} \cite{237} \cite{268}) has discussed the possibility of employing rigorous mathematical modeling in precision oncology, to date a complete, mathematically rigorous statistical theory of patient-specific probability of treatment success has been lacking. We have proposed such a framework in this work.

In Chapter 2 we discussed both mathematical and computational strategies for working with VPEs; while much of this material is well-traveled background, the application of rigorous statistical modeling of spatiotemporal \textit{physiological} processes via random fields is a largely untapped frontier. In other fields, namely the earth sciences \cite{261} \cite{57} \cite{68}, spatiotemporal statistical modeling is ubiquitous, but to our knowledge, no similar framework has been suggested for mathematical oncology; our rigorous definition of inter- and intra-patient physiological heterogeneity via Physiological Random Processes (PRPs) is, to our knowledge, a novel contribution to mathematical oncology.

One of the other key contributions of this work is to incorporate rigorous statistical image science into the modeling of the virtual patient and the quantification of uncertainties in M-PMED. While previous work (e.g. \cite{218} \cite{63} \cite{127}) has investigated sources of uncertainty in mathematical oncology, to our knowledge these works make convenient but unrealistic assumptions about the statistics of objects and image data (typically modeling both as Gaussian). In Chapter 3 we provided a rigorous analysis of molecular emission imaging data and the resulting virtual patient ensemble, emphasizing the role of both deterministic properties of the imaging system via the singular value decomposition, statistical properties of the imaging system via Poisson statistics, and noise properties of statistical image reconstruction algorithms. We discussed how a particular molecular emission imaging system gives rise to a decomposition of a drug concentration $c(r, t)$ into measurement and null components, which contributes to the overall decomposition of the object $f$ into measurement and null components. Our application of rigorous mathematical forward and inverse transport theory to photon-processing molecular imaging systems is entirely unique,
and builds on the work of CGRI (e.g. [43] [147] [148]) and mathematical inverse transport (e.g. [260] [132] [16]). We have also discussed several high-performance parallel simulation methods for ECT systems, providing novel parallel numerical methods for the solution of the scatter-free radiative transport equation and the computation of Fourier crosstalk matrices and singular value decompositions for both photon-counting and photon-processing systems.

In Chapter 4, we discussed general mathematical models $\mathcal{M}$ that seek to predict biomarkers of interest $\tilde{q}$ from the virtual patient $\tilde{f}$. This approach is partially inspired by quantity-of-interest modeling from the uncertainty quantification community [257] [182], but this level of generality is to our knowledge entirely novel in the context of precision medicine. For the particular case of chemotherapy efficacy, we have provided a particular example which defines a biomarker $q$ which is not directly measurable in vivo, and hence must be predicted using imaging data and a mathematical model $\mathcal{M}$. We then presented a novel mathematical model $\mathcal{M}$ that relies on knowledge of a spatially resolved bound-or-internalized drug AUC $\beta(r)$ and drug susceptibility $\alpha(r)$ to predict $q = -(\alpha, \beta)_{L^2(V)}$; we suggested how this quantity could be calculated for a particular patient using in vivo ECT imaging data. We discussed particular assumptions that lead to an explicit form for the PDF of $q$, allowing for the explicit computation of patient-specific probability of treatment success, and ultimately for the computation of figures of merit that incorporate both uncertainty and trade-offs between successful tumor kill and the possibility of normal tissue complications; we suggested the TOC curve as a way to display these tradeoffs graphically. We also discussed extensions to this model that take into account the fine-scale spatiotemporal distribution of the drug within the tumor microenvironment.

### 5.2 Future Work

As we have discussed, one of the primary goals of this work was to formulate a rigorous mathematical framework for the usage of image data and mathematical models to predict treatment outcomes in precision medicine. We have provided a glimpse into a large array of mathematical, statistical and computational tools and discussed several examples of explicit models that fit within this framework, but much more work, on the near and far horizon, is necessary to fully validate both
the specific models and the general modeling framework presented. The general framework proposed was presented in Section 4.1 and briefly summarized above, but to reiterate, the framework consists of the following components:

1. Identification of Physiological Random Processes (PRPs), $f$, which are suspected to influence outcomes in therapy;

2. Identification of clinically relevant quantities of interest $q$ whose value, if known, would provide a useful marker of success, and a set of outcomes $E_{\text{success}}$ which defines successful treatment with regard to the quantities $q$;

3. Identification of the measurement processes by which the PRPs $f$ can be measured for patient $j$, leading to a statistical model for the data $g|f_j$ collected for the individual patient;

4. Building of a Virtual Patient Ensemble model $\tilde{f}$, which is an approximation of the PRPs $f$ for patient $j$, using the measured data $g|f_j$ and possibly externally validated population models;

5. Building of mathematical models $M$ which seek to predict $q$ given a virtual patient $\tilde{f}$, culminating in the fundamental relationship

$$q = M(\tilde{f}) + \epsilon = \tilde{q} + \epsilon$$

6. Determination of the controllable parameters of the treatment, parameterized by a vector $\zeta$;

7. Determination of a figure of merit for treatment success; we postulate that the form

$$\mathcal{P}(\zeta) = \mathbb{P}[\tilde{q}(\zeta) \in E_{\text{success}}]$$

is sufficiently general, though other objective functions, for instance of the form $\mathcal{P}(\zeta) = \mathbb{E}[\phi(\tilde{q}(\zeta))]$, where $\phi(\cdot)$ is a convex function, may prove more tractable.

8. Optimization of the above objective functional over the class of admissible treatment parameters.
Each of the items outlined above is ripe for further investigation under the guises of theory, experiment or simulation. We will outline several specific projects, both short and medium term, that will seek to further demonstrate, validate and understand these core elements.

5.2.1 Specific Aims for 2018-2020

As a postdoctoral researcher at the University of Arizona, funded under NIH R01-EB000803 (P.I. H.H. Barrett), I aim to work on the following specific projects. For each, I will highlight how they will contribute to the specific aims of that grant.

Publication of the M-PMED framework in the medical literature One of the first papers I will write is a succinct description of the mathematical framework presented. Very recently (in fact, after this dissertation was defended), a paper was published in Translational Oncology by authors from Vanderbilt and U.T. Austin [191] which discusses, at a high level accessible to a general biomedical audience, many of the issues I have presented in this dissertation. However, the level of mathematical sophistication presented there is very light – in particular, while they do briefly mention uncertainties, patient-specific imaging data and tumor control/toxicity trade-offs, they provide no concrete mathematical strategies for addressing these issues and suggest a very informal treatment efficacy optimization problem of the form

Minimize \( F(\zeta) = f(\text{Tumor}(\zeta)) + g(\text{Toxicity}(\zeta)) \)

Subject to \( \text{Toxicity}(\zeta) \leq \text{Maximum tolerated Toxicity} \) \hspace{1cm} (5.2.1)

They do not propose specific forms for the ‘tumor’ and ‘toxicity’ objectives, nor do they suggest how to incorporate uncertainty, risk and probability, or how to parameterize patients and imaging data. In their conclusion, the authors of [191] ask:

...what is the \( F = ma \) for cancer? We have the means to measure tumor “mass” and “acceleration” (i.e., the multifactorial response of a tumor to therapy). Further, we can measure treatment “force” (i.e., drug pharmacokinetics). A modeling framework that relates these variables would offer the opportunity to adjust and optimize treatment...
regimens to maximize response. Mathematical models will form the foundation of this approach, and they will hasten the implementation and maximize the benefit of current (and future) therapeutics.

I believe that my work demonstrates explicitly how to implement a scheme such as (5.2.1), and while I cannot claim that I have provided an ‘F = ma’ for cancer (I and many others would argue that no such governing equation can exist, because cancer is a complex system), I would argue that \( q = \mathcal{M}(\tilde{f}) + \epsilon \) is, at least structurally, headed in the right direction: \( \mathcal{M} \) predicts relevant biomarkers \( q \) using measured and unmeasured patient-specific parameters as estimated by \( \tilde{f} \). I will thus aim to publish the key results of this dissertation - namely the concepts of model-based PMED, PRPs, VPEs, patient-specific VCTs, patient-specific probability of treatment success and the optimization problem

\[
\arg\min_\zeta P[\tilde{q}(\zeta) \in E_{success}]
\]

in a technically-oriented medical journal such as Translational Oncology, Science Translational Medicine, Statistical Methods in Medical Research, or similar.

**Parallel ECT system simulation** For the third chapter of this dissertation, I wrote a somewhat extensive set of simulation codes for ballistic radiative transport and ECT systems, both for photon-processing and photon-counting detectors. I will continue to develop, analyze and test this code, and anticipate several publications as a result. I have already begun work on the following publications:


In this paper, I will discuss the parallel ray transform method discussed in Section 3.7. I will compare the speed and accuracy of the method to existing ray transform codes, including Siddon’s method [254] and Gao’s method [95]. I may also investigate an adjoint transform method, and in a future related work, possibly geodesic ray transforms (joint work with F. Monard, UC Santa Cruz). I plan to submit to Physics in Medicine and Biology or a similar journal. This work contributes to Aim 1 of the R01 grant, ‘Imaging the radiance function’. 

In this paper, we will extend the results presented in Section 3.8 to provide a more complete crosstalk matrix and singular value analysis for photon-processing systems. A preliminary version of this work was presented at the 2017 IEEE Nuclear Science Symposium and Medical Imaging Conference, and will appear in the conference proceedings. We plan to submit to the IEEE Trans Med Imaging, SPIE J Med Imaging, Phys Med Biol, or a similar journal.


In this paper, I will discuss the analytical results presented in Chapter 3 regarding the null space of photon-processing systems. This paper will directly address Aim 3 from the R01 grant, ‘Characterizing the null space of ECT systems.’ I plan to submit to a mathematics-oriented journal such as SIAM J. Imaging Science or IOP Inverse Problems.

**Software for task-based assessment of image quality**  
Over the past several years, I have been working to produce a Matlab package to compute task-based measures of image quality for general imaging systems. The package includes the following modules:

1. A suite of 2D and 3D stochastic object simulation methods to draw from object distributions $\mathbb{P}_f$, including Gaussian, lognormal, generalized lumpy and Karhunen-Loève.

2. A basic system design and simulation module for ECT systems, where the user can specify the geometry of a system, including aperture and detector properties. The simulation uses the parallel RTE simulation methods presented in Chapter 3 to compute projection operators $\mathcal{P}$, and detector simulation methods to compute the operators $\mathcal{D}_{pc}$ and $\mathcal{D}_{pp}$.

3. Some basic image reconstruction modules such as MLEM will be available to simulate reconstructing objects from noisy ECT imaging data.

4. A module containing several model observers for both classification and parameter estimation tasks, including the Hotelling observer, the channelized Hotelling observer and the Bayesian ideal observer for classification tasks, as well as least squares and scanning linear estimators.
for estimation tasks. This module also contains statistical methods for estimating figures of
merit such as AUC, EMSE and Bayesian risk.

The first version of this code will be available on a code-sharing platform such as Github early
summer 2018, with published user documentation. This package contributes to Aim 5 of the R01
grant, ‘Task-based system evaluation and optimization’.

The concentration decomposition problem: extracting drug components from reconstructed images
In Chapter 4 we discussed how a drug concentration \( c(r, t) \) can be decomposed as

\[
c(r, t) = c^{\text{cap}}(r, t) + c^{\text{diff}}(r, t) + c^{\text{bound}}(r, t) + c^{\text{int}}(r, t)
\] (5.2.2)

When an image of a radiolabeled drug is reconstructed from data, what is produced is an estimate
of the complete \( c(r, t) \) (or more appropriately, the measurement component \( c_m(r, t) \)). However, for
the purposes of predicting treatment outcomes in chemotherapy, we would desire to have estimates
of each of the individual components in (5.2.2). In this project, we will investigate methods for
estimating these components either directly from data \( g \), or from a reconstruction \( c(r, t) \). Possible
strategies include the application of blind source separation techniques [41] [173] [308] and explicit
biochemical/PK/PD models that relate each of the components.

To be more precise, the concentration decomposition problem could be stated mathematically
as follows.

**Problem.** Let \( f \in \mathcal{F} \) be an object, where \( \mathcal{F} \) is a Hilbert space, and let \( g|f \sim P_{g|f} \) be a random
noisy imaging dataset, where \( P_{g|f} \) is, for instance, given by Poi(\( \mathcal{H} f \)). Assume further that we can
write

\[
f = \sum_{\ell=1}^{L} f_{\ell}, \quad f_{\ell} \in \mathcal{F},
\]

where there is a known implicit or explicit relationship between the \( f_{\ell} \), i.e. \( F(f_1, \ldots, f_L) = 0 \) for
some function \( F \). Then, the object is to reconstruct the individual functions \( f_{\ell} \) from \( g|f \).
Numerical simulation of PRPs relating to drug delivery and chemotherapy efficacy

In Chapter 4, we discussed a model for chemotherapy efficacy that requires knowledge of a drug concentration \( c(r, t) \), and in the above paragraph, we discussed how this drug concentration can be decomposed into components relating to different physiological regions. To further understand both the concentration decomposition problem and chemotherapy efficacy, we will develop numerical simulations that generate physiologically realistic drug concentrations \( c(r, t) \). In order to do so, we must consider the following issues:

1. To simulate a drug concentration \( c^{cap}(r, t) \) in a vascular network, we must be able to generate random vascular networks that are physiologically accurate. I have developed a preliminary 3D stochastic angiogenesis model similar to the Anderson-Chaplain model \([4]\) which assumes the presence of an angiogenic growth factor concentration \( v(r, t) \). Vessels are sprouted from an existing vessel and grow stochastically along the concentration gradient. Example realizations are shown in Figure 5.1. Given a vessel network, a drug concentration will be transported through the network according the biophysical laws of fluid transport \([137] [251]\).

2. To simulate the diffused drug concentration \( c^{diff}(r, t) \), we must consider drug extravasation
through the vessel wall and diffusion through the interstitial space; diffusion is governed by
the diffusion equation, as discussed in Chapter 4. We can simulate drug diffusion using a
Green’s function method similar to [252] [251]. Preliminary results of such a method, for the
vessel networks shown above, is shown in Figure 5.2.

Figure 5.2: Preliminary results for a Green’s function-based diffusion code to simulate
\(c_{\text{diff}}(r, t)\) given \(c_{\text{cap}}(r, t)\).

3. To the simulate bound and internalized components, we must employ chemical kinetic models
similar to those given in Chapter 4.

We will consider working with the Secomb group at Arizona to develop this project further,
perhaps extending their existing Green’s function diffusion code to incorporate the necessary uncer-
tainties. Another aspect we should investigate (in connection with the next item) is the appropriate
PRP model for the drug susceptibility field \(\alpha\).

This project will contribute to Aim 4 of the R01 grant, ‘Molecular imaging and random pro-
cesses’.

**Experimental Validation of PRP Theory** I plan to collaborate with H.H. Barrett and others
at the Center for Gamma Ray Imaging to push for an experimental validation of several of the
theories outlined in this work. In particular, we hope to validate the integrated-log-kill model, \( \tilde{q} = - (\alpha, \beta) \). This will involve gathering molecular images of cell proliferation and drug concentration, computing drug AUC and applying the concentration decomposition methods outlined above. This will contribute to Aim 4 of the R01 grant, ‘Molecular imaging and random processes’.

5.2.2 Medium-term Projects

The following projects I can envision producing publications in the next three to five years.

**Infinite dimensional statistical inverse problems with point process data** As we have presented in this work, emission imaging systems can give rise to data \( g \) which takes the form of a Poisson point process. Precisely, we have shown in Chapter 3 that for a photon processing system which images the object \( f \in L^2(I) \), we have

\[
g|f \sim \text{Poi}(\mathcal{H}f)
\]

where \( \mathcal{H} : L^2(I) \to L^2(A) \) is a compact linear operator. We denote the log-likelihood (defined in Chapter 3) as \( \ell(f) \) or \( \ell(f|g) \). One of the primary goals of imaging is to construct an estimate of the object \( f \) – or more generally a ‘virtual ensemble’ \( \tilde{f} \). We have mentioned several approaches:

1. Let \( \hat{f} \) be the maximum likelihood estimate of \( f \) over some set \( \Theta \), i.e.

\[
\hat{f} = \arg \max_{f \in \Theta} \ell(f)
\]

2. Let \( \hat{f} = \mathcal{O}(g) \) be defined in terms of some general reconstruction operator \( \mathcal{O} \), for instance a Backus-Gilbert type method, a regularized maximum likelihood, etc.

3. Define a prior \( \mathbb{P}_0 \) over the set \( \Theta \) (which, to be clear, is a subset of an infinite dimensional space, hence \( \mathbb{P}_0 \) must be thought of as a random process model). Then, let \( \tilde{f} \) be the Bayesian posterior, i.e.

\[
\tilde{f} \sim \mathbb{P}_{f|g}
\]
where $P_{f|g}$ is the appropriate posterior measure on $\Theta$. Again, because $\Theta$ is a subset of an infinite dimensional space of functions, $P_{f|g}$ must be treated as the distribution of a random process (and thus carefully!). Very recent work (e.g. [266] [73] [267] [204]) has made progress towards defining this notion rigorously. To illustrate briefly, it is shown in e.g. [73] [267] that, under some very restrictive assumptions on the prior $P_0$ and the likelihood $\ell(f|g)$, that the posterior $P_{f|g}$ is absolutely continuous with respect to the prior $P_0$, and that

$$\frac{dP_{f|g}}{dP_0} \propto \exp(\ell(f|g)) = \exp(-\Phi(f|g))$$

where $\Phi(\cdot)$ can be thought of as an energy in the Gibbs sense. This expression then gives rise to appropriate Markov chain Monte Carlo methods to sample from $P_{f|g}$, hence providing an avenue to perform statistical inference in this setting (for example, computing model outputs $\tilde{q} = \mathcal{M}(\tilde{f})$ and probabilities $\mathbb{P}[\tilde{q} \in E]$).

The goal of this project will be to analyze the statistical properties of the three inversion strategies outlined above in the particular case of Poisson point process data and develop efficient computational routines to sample from the corresponding virtual ensembles. More precisely, we have

**Problem.** Let $\mathcal{X}$ be a Hilbert space, and $\mathcal{H} : \mathcal{X} \to L^1_+(A)$ be a compact linear operator mapping $f \in \mathcal{X}$ to a non-negative integrable function $\bar{g}(a) = (\mathcal{H}f)(a) \in L^1_+(A)$. Define the Poisson point process with intensity function $\bar{g}$ via $\text{Poi}(\bar{g})$. Then,

1. Rigorously define the Bayesian posterior $P_{f|g}$ for this form of data; which prior measures $P_0$ on $\mathcal{X}$ give rise to well-defined posteriors?

2. Prove an appropriate Bernstein-von Mises-type asymptotic consistency and normality theorem. In this context, the appropriate asymptotic limit will be as the total intensity goes to $\infty$, i.e.

$$\bar{N} = \int_A f(a) \, da \to \infty$$

3. For the resulting posterior $P_{f|g}$ from 1., define an appropriate Monte Carlo method (Markov
chain or otherwise) which samples $P_{f|\Theta}$ efficiently.

### Simulating a random process when only $\Psi_f$ is known

For all the random process models considered in this work, we have presented methods for drawing samples which require either explicit knowledge of some finite-dimensional PDF $p(x)$, or otherwise some direct, generative method for drawing samples from $f$ (for instance in the case of generalized lumpy background processes). However, we also know that the characteristic functional $\Psi_f[\phi]$ provides a complete statistical characterization of the process $f$, and in many cases it is easier to derive a form for $\Psi_f$ than for any $p(x)$. Thus, it would be nice to be able to simulate a random process if only $\Psi_f$ is known. This inspires the following problem, which to our knowledge does not yet have a satisfactory solution in the literature:

**Problem.** Let $f \in \mathcal{H}$ be a generalized random process with known characteristic functional $\Psi_f[\phi]$, $\phi \in \mathcal{T}$, where $\mathcal{T}$ is a space of test functions. Define an efficient sampling algorithm (I.I.D., Markov, or otherwise) which produces a sequence $f_1, f_2, \ldots$ such that $f_j \sim \Psi_f$. In other words, we require that the empirical measure converges to $P_f$:

$$
\frac{1}{N} \sum_{j=1}^{N} \delta_{f_j} \to P_f
$$

where $P_f$ is the probability measure on $\mathcal{T}'$ induced by $\Psi_f$. This mode of convergence would imply that any expected value computed with the sample sequence would converge to the true expected value, i.e. (in some convergence sense)

$$
\lim_{N \to \infty} \frac{1}{N} \sum_{j=1}^{N} F(f_j) = \langle F(f) \rangle_f
$$

The reason this problem appears difficult is that while $\Psi_f$ can in principle be used to derive finite-dimensional PDFs, for instance the PDF $p_{\phi_1,\phi_n}(x)$ of the random vector $u_{\phi_1,\phi_n} = [X_{\phi_1}, \ldots, X_{\phi_n}]$†, evaluating this PDF or finding an explicit form for it requires performing an inverse Fourier transform in high dimension. To our knowledge, all explicit sampling methods that would produce samples of $u_{\phi_1,\phi_n}$ (accept-reject, Gibbs, Metropolis-Hastings, etc.) require at least
being able to evaluate $p_{\phi_1, \ldots, \phi_n}(x)$ for an arbitrary input $x \in \mathbb{R}^n$.

Possible strategies include:

- Using MCMC or Quasi Monte Carlo (QMC) to perform the inverse Fourier transform

- Approximating $\Psi_f$ with a known functional, for instance a Gaussian or a generalized lumpy background, then simulating the resulting process. More explicitly, suppose that $\Psi[\phi; \theta], \theta \in \Theta$ is a parametric family of characteristic functionals with known sampling procedure that depends on $\theta$ (an example would be $\mathcal{N}(\mu, C)$). Then, we could choose $\theta^* \in \Theta$ such that

$$
\Psi[\phi; \theta^*] \approx \Psi_f[\phi] \quad \forall \phi \in \mathcal{F}
$$

Then, by drawing samples from $\Psi[\phi; \theta^*]$, we approximate samples from $\Psi_f$.

- Another approach uses an approximate form of the (forward or inverse) Fourier transform of a non-negative function $\psi(x) = \exp(-F(x))$; this makes use of the Legendre transform and is inspired by large deviations theory \[38\].

Multi-pinhol ECT Collimator design One of the main features of the photon-processing framework is that individual photon interaction attributes can be estimated. Recall that a photon interacts with a detector with attribute $a = (r, \hat{s}, E, t)$; we have assumed that $r, E$ and $t$ can be estimated using either maximum likelihood or some other methodology. In principle, there is nothing preventing the estimation of photon interaction angle $\hat{s}$, though with most current setups, this estimation would likely prove difficult; as we mentioned in Chapter \[3\] calibrating a likelihood model that incorporates angle would be challenging – the current strategy is to measure an MDRF for each detector, which involves the positioning of a point source across the detector face, and this process is already very time consuming. To calibrate a likelihood model for angle estimation would require the variation of the source’s angle in addition to position, which would make the MDRF calibration combinatorially more expensive (if $n$ angles are to be calibrated, the calibration time would be multiplied by $n$).

One possible solution is to take inspiration from the plenoptic design for phase-space photograi-
phy, which employs arrays of ‘lenslets’ \cite{212} \cite{46}. In such a design, an array of apertures is placed in front of a detector; assuming that position can then be estimated to sufficiently high accuracy on the detector face, angle can be estimated by selecting the most likely pinhole, then estimating the angle formed by the ray drawn from the estimated interaction position and the estimated pinhole. Such a design is illustrated in \cite{5.3}. In theory, such a design should improve image quality because more information is gathered for each photon interaction; in a single-pinhole setup, angles are averaged over a cone instead of estimated.

Figure 5.3: Demonstration of the aperture cones for a multi-pinhole aperture setup. This arrangement acts to define a particular measurement set $\mathcal{A}_\text{det} \subset \partial_+ \Gamma$ and corresponding aperture function $T_{ap}(r, \hat{s})$. Compared to a single pinhole, a larger subset of $\partial_+ \Gamma$ is detectable due to the presence of more apertures; it is conjectured that such an arrangement would lead to improved image quality.

The goal of this project will be to work with others at CGRI to investigate the feasibility of implementing such a design in a real ECT imaging system. My role in this project will be to provide theoretical and simulation support.


Chapter 6

Appendix

6.1 Rigorous Definitions: Probability and Random Processes

In Section 2.1.1, we discussed some fundamental notions from probability theory. Here, we provide a few rigorous mathematical definitions that were left out.

Definition (Sigma Algebra). Let \( \Omega \) be a sample space and let \( \mathcal{F} \subset 2^\Omega \) be a collection of subsets of \( \Omega \). Then, \( \mathcal{F} \) is a sigma algebra if

1. \( \Omega \in \mathcal{F} \)

2. \( \Omega \setminus E \in \mathcal{F} \forall E \in \mathcal{F} \), where \( \Omega \setminus E \) is the complement of \( E \) in \( \Omega \).

3. \( E_1 \cup E_2 \cup \cdots \in \mathcal{F} \) if \( E_j \in \mathcal{F} \).

Intuitively, conditions (2) and (3) in the definition of a sigma algebra indicate that the basic operations ‘not’ and ‘or’ should lead to well-defined events. Combining (2) and (3) implies that countable intersections are events, and hence the ‘and’ operation is well-defined. Other logical conclusions about the construction events follow from these three definitions.

Definition (Probability Measure). Let \( \Omega \) be a sample space and \( \mathcal{F} \) a sigma algebra of subsets of \( \Omega \). Then, a function \( \mathbb{P} : \mathcal{F} \to \mathbb{R} \) is called a probability measure if

1. \( \mathbb{P}(E) \in [0,1] \) for all \( E \in \mathcal{F} \).
2. \( P(\emptyset) = 0 \) and \( P(\Omega) = 1 \).

3. \( P(\bigcup_{j=1}^{\infty} E_j) = \sum_{j=1}^{\infty} P(E_j) \) if \( E_j \cap E_k = \emptyset \) for \( j \neq k \).

One can combine the definitions of the sample space and the probability function to derive the basic calculus of probability. For instance, the probability of an event \( E \) not happening is \( 1 - P(E) \), which then follows from \( P(\Omega) = 1 \), \( \Omega = E \cup (\Omega \setminus E) \) and property (c) of \( P \).

A special type of sample space and collection of event sets arises when \( \Omega \) is initially not defined as a probability space (that is, no sigma algebra of events nor probability measure has been defined), but rather is only a topological space.

**Definition** (Borel Probability Space [84]). Suppose that \( \Omega \) is a topological space with topology \( \mathcal{T} \). Then, there exists a sigma algebra \( \mathcal{B} \supset \mathcal{T} \) which called the Borel sigma algebra. Together with a probability measure, \( (\Omega, \mathcal{B}, \mathbb{P}) \) forms a Borel probability space.

For instance, if \( \Omega \) is a metric space, Hilbert space, Banach space, or space of generalized functions, it possess a collection of open sets, denoted \( \mathcal{O} \), which provide a notion of convergence. Then, a ‘natural’ sigma algebra of events can be defined which contains \( \mathcal{O} \) by letting \( \mathcal{B} \) be the smallest sigma algebra containing \( \mathcal{O} \); this is a well-defined sigma algebra [84]. For example, the Borel sigma algebra on the Hilbert space \( L^2(V) \) is the smallest collection of subsets of \( L^2(V) \) which is simultaneously a sigma algebra, and which contains all sets of the form \( \{ \| u - v \|_2 < \epsilon : u, v \in L^2(V), \epsilon > 0 \} \). Such a definition provides an avenue to rigorously define a probability function on infinite-dimensional space such as \( L^2(V) \), that is, a measure \( \mathbb{P} \) such that it makes sense to ask ‘what is the probability that a random function \( f \) is in the (Borel) set \( E \subset L^2(V) \)?’ Such probability functions arise in the definition of random processes, though, they typically do not possess particularly ‘nice’ properties, and serve only as an abstraction of the concept of uncertainty for infinite-dimensional objects.

**Definition** (Measurable Function). Let \( (\Omega, \mathcal{F}) \) and \( (\Omega', \mathcal{F}') \) be two measurable spaces. Then, a function \( f : \Omega \to \Omega' \) is called measurable if for all \( E' \in \mathcal{F}' \),

\[
    f^{-1}(E') = \{ x \in \Omega : f(x) \in E' \} \in \mathcal{F}
\]
Note that the definition of a measurable function is indeed philosophically connected with the notion of a physical observable in the following sense. In the original formulation of measure theory (by Borel and Lebesgue, among others), the measures under consideration were intended to be mathematical idealizations of the geometrical notions of ‘length’, ‘area’ and ‘volume’, notions which certainly correspond to physical observables (i.e. physically ‘measurable’ quantities). The class of sets $E$ for which a reasonable length, area or volume could be assigned were deemed ‘measurable’; Borel defined a certain class which was connected to topology, while Lebesgue considered a slightly larger class which included sets of intuitively ‘zero’ volume. It was necessary to define such classes because paradoxes arose in which one could perform bizarre non-physical things to mathematical sets, the most famous example being due to Banach and Tarski who showed that one could ‘create two unit balls from a single unit ball’; the problem was that the sets employed had no natural definition of volume, so this magic could not be physically realized.

A complete treatment of probability on Banach spaces can be found in e.g. [276]; fairly general statements can also be made in ‘complete separable metric spaces’ [84]. The case of random variables taking values in spaces of generalized functions is treated in [100].

### 6.2 The Matérn-Whittle Covariance Class

In this appendix, we work through a brief derivation of the Matérn family of isotropic Gaussian random fields, discussed briefly in Section 2.6. This family provides a range of smoothness behaviors according to the choice of a parameter $\nu$. The class was introduced by Matérn in the context of geostatistics in his 1960 monograph [189]; Whittle had introduced a special case corresponding to $\nu = 1$ in 1954 [296]. See also Stein [201], Rasmussen [230] and [68], [57], [301].

We define, similar to Whittle [296], the stationary random field $u_\nu(\tau)$, where $\tau \in \mathbb{R}^n$ as the solution to the following linear stochastic pseudodifferential equation:

$$\mathcal{L}_\nu u_\nu \equiv (\mathcal{I} - \nabla^2)^{\nu/2} u_\nu = w$$

where $w$ is a Gaussian white noise process, defined as a generalized random process over test
functions $\mathcal{F} = C_0^\infty(\mathbb{R}^n)$, $s = n/2 + \nu$, $\mathcal{I}$ is the identity operator, and $\nabla^2 = \sum_{j=1}^n \frac{\partial^2}{\partial x_j^2}$ is the Laplacian. The parameter $\nu$ can take any positive real value, and hence we can have any $s \in (n/2, \infty)$; we define non-integer powers of the differential operator $(I - \nabla^2)$ via the Fourier transform:

$$\mathcal{F}[(I - \nabla^2)^p f](\xi) = (1 + 4\pi^2\|\xi\|^2)^p \hat{f}(\xi)$$

With this definition, we can formally solve (6.2.1) by inverse Fourier transform:

$$u_\nu(\tau) = L_\nu^{-1} w = \int_{\mathbb{R}^n} \frac{\hat{w}(\xi)}{(1 + 4\pi^2\|\xi\|^2)^{s/2}} \exp(2\pi i \xi^T \tau) \, d\xi = \int_{\mathbb{R}^n} w(\tau') \int_{\mathbb{R}^n} \frac{\exp(2\pi i \xi^T (\tau - \tau'))}{(1 + 4\pi^2\|\xi\|^2)^{s/2}} \, d\xi \, d\tau'$$

$$= \int_{\mathbb{R}^n} h_\nu(\tau - \tau') w(\tau') \, d\tau'$$

where the kernel $h_\nu(\tau - \tau')$ is defined via the inverse Fourier transform:

$$h_\nu(\tau - \tau') = \int_{\mathbb{R}^n} \frac{\exp(2\pi i \xi^T (\tau - \tau'))}{(1 + 4\pi^2\|\xi\|^2)^{s/2}} \, d\xi$$  \hspace{1cm} (6.2.2)

Because we have defined $u_\nu$ in terms of a linear operator applied to $w$ (which, for certain values of $\nu$ is bounded on $\mathcal{F}'$), we can immediately apply the linear transformation rule of characteristic functionals to see that

$$\Psi_{u_\nu}[\phi] = \exp \left(-2\pi^2 \|L_\nu^{-1} \phi\|_{L^2(\mathbb{R}^n)}^2 \right) = \exp \left(-2\pi^2 (\mathcal{C}_\nu \phi, \phi)_{L^2(\mathbb{R}^n)} \right)$$  \hspace{1cm} (6.2.3)

where the covariance operator $\mathcal{C}_\nu$ is given by (recall that $\mathcal{C}_w = \mathcal{I}$, the identity and note that $L_\nu^{-1}$ is self-adjoint):

$$\mathcal{C}_\nu = L_\nu^{-1} \mathcal{I} \left( L_\nu^{-1} \right)^\dagger = L_\nu^{-2}.$$

In other words, because the Fourier multiplier (aka transfer function) of $L_\nu^{-1}$ is given by $(1 + 4\pi^2\|\xi\|)^{-s/2}$, the Fourier multiplier of $L_\nu^{-2}$ is given by $(1 + 4\pi^2\|\xi\|)^{-s}$, and we can write the
covariance function of $u_\nu$ in terms of an inverse Fourier transform similar to (6.2.2):

$$c_\nu(\tau - \tau') = \int_{\mathbb{R}^n} \frac{\exp(2\pi i \xi^\top(\tau - \tau'))}{(1 + 4\pi^2\|\xi\|^2)^s} d\xi$$  \hspace{1cm} (6.2.4)

The covariance function (6.2.4) is the Matern covariance with parameter $\nu$. Using an integral table [112], we can compute (6.2.4) more explicitly in terms of a modified Bessel function $K_\nu$. Note first that (6.2.4) is an inverse Fourier transform of an isotropic function, i.e.

$$c_\nu(z) = \int_{\mathbb{R}^n} \exp(2\pi i \xi^\top z) p(\|\xi\|) d\xi.$$  \hspace{1cm} (6.2.5)

We can use $n$-dimensional polar coordinates to write (6.2.4) in terms of a Hankel transform [26]:

$$c_\nu(z) = \frac{2\pi}{\|z\|^{n/2-1}} \frac{1}{(2\pi)^{2s}} \int_0^\infty \frac{J_{n/2-1}(2\pi r\|z\|) r^{n/2}}{\left(\frac{1}{2\pi} + r^2\right)^s} dr.$$  \hspace{1cm} (6.2.5)

We can now apply [112]6.565.4 with $\nu = n/2 - 1$ (not the same as our $\nu$!), $\mu = s - 1$, $b = 2\pi\|z\|$, and $a = 1/2\pi$ to obtain

$$c_\nu(z) = \left(\frac{2\pi}{\|z\|^{n/2-1}} \frac{1}{(2\pi)^{2s}} \frac{2^{s-1} \Gamma(s)}{\Gamma(n/2+s)}\right) K_{n/2-s}(\|z\|) \frac{\|z\|^\nu K_\nu(\|z\|)}{2^n \pi^{n/2} 2^{\nu-1} \Gamma(n/2 + \nu)}.$$  \hspace{1cm} (6.2.6)

where $K_\nu(s)$ is the modified Bessel function of the second kind, and $\nu = s - n/2$. Note that we used the identity $K_\nu(s) = K_{-\nu}(s)$ to arrive at (6.2.7) [2]. By allowing arbitrary scaling factors for both the input and output, we arrive at the Matérn family as given in Table 2.1:

$$c(z; \sigma^2, \nu, \ell) = \frac{\sigma^2}{2^{\nu-1} \Gamma(n/2 + \nu)} \|z/\ell\|^\nu K_\nu(\|z/\ell\|)$$  \hspace{1cm} (6.2.8)

We thus have two equivalent definitions of this family; we can either state that a Matérn class random field is a Gaussian random process with mean $\mu$ and covariance function given by (6.2.8); equivalently we can provide the characteristic functional (6.2.3); or, we can describe it as a solution to the (in general fractional order) stochastic equation (6.2.1).
6.3 Philosophy of Probability and Uncertainty

To paraphrase Gelman and Shalizi [101], statisticians design statistical methods and should work to analyze their efficacy in different scenarios; methods that work are deemed useful, regardless of their philosophical motivation, and all models are subject to objective evaluation. The same is true in image science and precision medicine: scientific objectivity is applied to the analysis of systems, while the models and methods that make up a system are usually selected for efficiency, convenience, or some other (occasionally ad hoc) reasons.

An important philosophical question arises upon the introduction of a mathematical framework for uncertainty. What is the meaning of the number $P(E)$? In what sense is it a ‘chance’ or ‘probability’? As it turns out, this question is surprisingly (or perhaps unsurprisingly) subtle, and centuries of discussion in the philosophy of science have dealt with it [146] [101] [225] [222] [79]. The conclusion is usually that there are two interpretations. Assume that $\Omega$ represents a set of possible experimental outcomes for some well-defined experiment (say, measuring the number of clicks that a Geiger counter sounds in a period of time). The two interpretations are then:

1. If we repeat the experiment many times, the number $P(E)$ represents the limiting ratio of the number of times $\omega$ satisfies the condition of being in $E$, that is,

$$P(E) = \lim_{N \to \infty} \frac{M(N)}{N}$$

where $M(N)$ is the number of ‘successful’ trials, that is, trials for which $\omega \in E$. This interpretation of $P(E)$ is intended to be verifiable in the sense that if your friend claims, say, that $P(E) = 0.2$, you can go into the lab and perform many experiments to verify this claim.

2. To the best of our knowledge, we predict that the chance of $\omega$ satisfying the condition of lying in $E$ is $P(E)$. This claim is sometimes not verifiable in the same sense as the first in that the experiment is sometimes not repeatable. The most famous recent example is the 2016 presidential election: pundits made many claims about the chance of a Clinton victory; some of these claims were as high as 95%. However, the outcome of the experiment was indeed a Trump victory, and this is an experiment that we cannot repeat in order to validate the
claim that Clinton wins 95% of the time.

The argument then usually ensues that only one of the above interpretations is correct. However, in our view, this is an unproductive argument: in some cases, we are forced to admit the second interpretation by the conditions of an experimental setup, while in some other cases, we can indeed repeat many trials to compute $P(E)$. In particular, in cases where the goal is to predict the probability of some clinical outcome taking place, which is one of the central goals of this dissertation, we must admit that what we are computing is a probability of the second kind, that is, a prediction of chance which is computed to the best of our knowledge. The success of our prediction can be verified by repeated trials (in our case, real-life clinical trials), however the trials are admittedly not identical repetitions of the same experiment because each patient will be slightly different. The kind of probability that will be used to make clinical decisions is thus analogous to the way the weather is predicted: a ‘50% chance of rain tomorrow’ is not a verifiable probability claim in the classical (first) sense, because we cannot repeat tomorrow infinitely many times. The 50% is a prediction based on many simulations of tomorrow. In this work, patient-specific probability of treatment success is an example of a probability based entirely on simulations performed ‘to the best of our knowledge’. Further discussion of the idea that statistical quantities can be computed by any means, but then ultimately should be validated by repeated trials - is found in Barrett and Myers [26].

6.4 Adjoint Operators

Here we briefly provide a proof of equation 3.3.9, which was the key technical formula used for the derivation of the adjoint attenuated X-Ray transform. Recall that 3.3.9 states that if $f \in L^1(\Gamma)$, then

$$
\int_{\Gamma} f(r, \hat{s}) \, dr \, d\hat{s} = \int_{\partial_{+} \Gamma} \int_{0}^{\tau_- (r', \hat{s})} f(r - \ell \hat{s}, \hat{s}) \, d\ell \, |\hat{s} \cdot \hat{\nu}| \, dr' \, d\hat{s}
$$

(6.4.1)

where $\Gamma = V \times S^2$ is the optical phase space, $\tau_-$ is the reverse exit time, and $\hat{\nu}$ is the outward unit normal to the boundary of $V$. To prove (6.4.1), we parameterize $(r, \hat{s}) \in \Gamma$ from the boundary by
the change of variables

\[ \phi : \partial_+ \Gamma \times [0, \infty) \to \Gamma, \quad \phi(r', \hat{s}, \tau) = (r' - \tau \hat{s}, \hat{s}) \]

This mapping is one-to-one between \( \Gamma \) and \( \partial_+ \Gamma \times [0, \infty) \), since given \((r, \hat{s}) \in \Gamma\), we have

\[ \phi^{-1}(r, \hat{s}) = (r + \tau_+(r, \hat{s})\hat{s}, \hat{s}, \tau_+(r, \hat{s})) \]

Both \( \phi \) and \( \phi^{-1} \) are also continuously differentiable, so we can apply the change of variables theorem:

\[ dr d\hat{s} = |\det J_\phi| d\tau dr' d\hat{s} \tag{6.4.2} \]

Note briefly that (6.4.2) looks wrong (there appears to be too many variables on the right), but the \( r \) on the left is a 3D position while the \( r' \) on the right is a two-dimensional variable \((r' \in \partial V)\).

To complete the proof of (6.4.1), we simply need to prove that \(|\det J_\phi| = |\hat{s} \cdot \hat{\nu}|\). To do so, assume that the boundary is parameterized locally by \( r' = r'(\alpha, \beta) \); then we have

\[ \hat{\nu} = \frac{\partial r'}{\partial \alpha} \times \frac{\partial r'}{\partial \beta}. \tag{6.4.3} \]

Furthermore we parameterize \( \hat{s} \in S^2 \) with the usual spherical coordinates, \( \hat{s} = \hat{s}(\theta, \varphi) \). We can now write \( \phi \) in terms of \((\alpha, \beta, \theta, \varphi, \tau)\), and so the Jacobian of \( \phi \) is given by

\[
J_\phi = \begin{pmatrix}
\frac{\partial r'}{\partial \alpha} & \frac{\partial r'}{\partial \beta} & -\tau \frac{\partial \hat{s}}{\partial \beta} & -\tau \frac{\partial \hat{s}}{\partial \varphi} & -\hat{s} \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0
\end{pmatrix}
\]

Thus by simplifying the determinant, using the triple product formula for \( 3 \times 3 \) determinants and (6.4.3), we have

\[
|\det J_\phi| = \left| \det \left( \frac{\partial r'}{\partial \alpha} \frac{\partial r'}{\partial \beta} - \hat{s} \right) \right| = \left| \left( \frac{\partial r'}{\partial \alpha} \times \frac{\partial r'}{\partial \beta} \right) \cdot \hat{s} \right| = |\hat{\nu} \cdot \hat{s}|
\]
Bibliography


248


258


[218] J Tinsley Oden, Ernesto ABF Lima, Regina C Almeida, Yusheng Feng, Marissa Nichole Rylander, David Fuentes, Danial Faghhihi, Mohammad M Rahman, Matthew DeWitt, Manasa


