MASTER'S THESIS

Minimizing Variance in Monte Carlo Based Iterative PET Reconstruction

Mate Balogh

Supervisor: David Legrady
Associate Professor
BME Department of Nuclear Technology

BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS

2014.

To my parents

Diplomamunka feladat a Fizikus mesterképzési (MSc) szak hallgatói számára

A hallgató neve:	**Balogh Máté**	szakiránya:	**orvosi fizika**
A diplomamunkát gondozó (a záróvizsgát szervező) tanszék:			**Nukleáris Technika Tanszék**

A diplomamunka készítésének helye:	BME TTK		
A témavezető neve:	Légrády Dávid	A konzulens neve: (külső témavezető esetén kijelölt tanszéki munkatárs)	
– munkahelye:	BME NTI		
– beosztása:	Egyetemi docens	– beosztása:	
– e-mail címe:	legrady@reak.bme.hu	– e-mail címe:	

A diplomamunka címe:	ML-EM PET rekonstrukció konvergencia-analízise	azonosítója: **DM-2013-69**

A téma rövid leírása, a megoldandó legfontosabb feladatok felsorolása:

Az izotópdiagnosztikai tomográfiában, ezen belül a Pozitron Emissziós Tomográfiában egyeduralkodóvá válik napjainkban az Maximum Likelihood Expectation Maximization (ML-EM) módszer illetve variánsai az MAP-EM, OSEM, stb. Az emissziós ML-EM algoritmus iterációs képletének direkt Monte Carlo (MC) mintavételezése a szokásos sztochasztikus modell kibővítését igényli többek között a MC várhatóérték becslések torzítatlanságát illetően. További probléma a mintavételezés okozta zaj. A hallgató feladata a fenti témakörök matematikai tárgyalása különös tekintettel a következő kérdésekre: - optimális ML-EM MC mintavételezés a becslés torzítatlansága érdekében - optimális ML-EM MC mintavételezés a becslés a rekonstruált kép zajtartalmának csökkentésére adott mintaelemszám és általános céltárgy esetére - regularizációs tag (MAP-EM) tervezése és tesztelése a MC mintavételezés okozta konvergenciaproblémák figyelembe vételével - valamilyen értelemben ideális térfelbontású modell vizsgálata reguláris karteziánus ráccsal osztott voxeltér helyett A hallgató feladata továbbá, egy magasabb szintű programozási környezetben (pl. MATLAB) olyan legalább 2D szimulációs modell megépítése, mellyel a fenti problémák részletei demonstrálhatóak és a megoldások tesztelhetőek.

A feladat kiadásának időpontja:	**2013. február 5.**

Témavezető vagy tanszéki konzulens aláírása:	A diplomamunka témakiírását jóváhagyom (tanszékvezető aláírása):

Alulírott **Balogh Máté** a Budapesti Műszaki és Gazdaságtudományi Egyetem fizikus MSc szakos hallgatója kijelentem, hogy ezt a diplomamunkát meg nem engedett segédeszközök nélkül, önállóan, a témavezető irányításával készítettem, és csak a megadott forrásokat használtam fel.
Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból vettem, a forrás megadásával jelöltem.

Budapest,

aláírás

Contents

Kivonat

Diplomamunkámban iteratív PET (Pozitron Emissziós Tomográfia) rekonstrukció optimalizálásával foglalkozom. Korábbi kutatások során [1]-ben vizsgálták a mintavételezési bizonytalanságoknak a rekonstrukcióra kifejtett hatását. Az eredmények összehasonlíthatóságáért a módszer-specifikus számolásaim Monte Carlo alapú rekonstrukciókra fókuszálnak. Munkámban a rekonstrukció L_2 hibájának minimalizálására koncentrálok.

Először részletesen bemutatom az egyszerűsített rendszert, amelyet a munkám során alapul veszek. Ezután a rendszer viselkedését visszaadó matematikai modellt definiálok. Ez a matematikai modell a későbbi elméleti okfejtések alapja. Megmutatom, hogy tisztán elméleti eredmények levezetéséhez szükséges lenne pozitív binomiális véletlen változók reciprokának eloszlását kezelni. Ismert, hogy ennek a problémának nincs általános megoldása és még egyszerűbb esetekben is komoly nehézségekkel jár. Ezen akadály megkerülésére a véletlen változó reciprokát normál eloszlással közelítem.

A rendszer tulajdonságainak a rekonstrukció minőségére kifejtett hatásait is vizsgálom. Ennek érdekében kiszámítom a várható rekonstruált varianciát a rendszermátrix elemeinek függvényében. Az eredményekből az látszik, hogy a rekonstrukció rendszermátrixok széles skáláján optimális. Ebből következően az iteratív rekonstrukció robusztus képrekonstrukciós eljárás, amely nem érzékeny a rendszerparaméterek kis változásaira.

Ezután levezetek egy elméletileg optimális mintavételezést az előrevetítés lépéséhez. Az optimalizálást a teljes beütésszám varianciájára végzem, azonban ez nem feltétlenül eredményez minimális rekonstruált varianciát. Ezt mutatja a 2×2 rendszer viselkedését vizsgáló numerikus szimuláció eredménye is. A tesztelés eredménye nem tudja egyértelműen megadni, hogy az újonnan javasolt mintavételezés valóban optimális-e.

A visszavetítés lépésére is meghatározom az optimális mintavételezést. Ebben a számításban már a rekonstruált kép várható négyzetes hibáját minimalizálom, így ez nagyobb valószínűséggel vezet valódi képminőség javuláshoz. Emellett, a javasolt mintavételezés megvalósításához szükség van a detektált jelek szórásának ismeretére, amelynek meghatározása nem triviális feladat. Az általam írt 2×2 iteratív rekonstrukció eredménye szerint az új mintavételezés valóban csökkenti a rekonstrukciós hiba mértékét.

Az előzetes numerikus tesztelés szerint az előrevetítés lépéséhez levezetett mintavételezés az egyenletes mintavételezésnél jobban, míg az aktivitás arányos verziónál rosszabbul teljesít. Azonban a visszavetítéshez meghatározott optimális mintavételezés jobb képminőséget eredményez, mint az eddig használt módszerek.

Abstract

In my thesis I work on improving iterative PET (Positron Emission Tomography) image reconstruction. Previous research in [1] examined the effect of sampling uncertainties on reconstruction quality. My method-specific calculations also focus on MC based reconstruction to allow for comparison of results. Most of my work focuses on minimizing L_2 error of reconstructed activity concentrations.

First I give a detailed technical description of the slightly simplified system I consider in my research. Afterwards, a mathematical model is defined that replicates system behaviour. This mathematical model forms the basis of further theoretical inquires. I show that purely theoretical results would require handling the inverse distribution of a positive binomial random variable, which is known to be problematic even in the simplest cases. To avoid this problem, the inverse distribution is approximated by a Gaussian.

The effect of system properties on overall reconstruction quality is also investigated. This is done by determining the variation of expected reconstructed variance with system matrix elements. It is found that reconstruction is optimal for a wide range of system matrices, therefore iterative reconstruction is a robust approach that is not sensitive to slight variations in system details.

I proceed to derive a theoretically optimal distribution of Monte Carlo particles between individual voxels in the forward projection step. Optimization is carried out with respect to variance of total detection count, however this does not necessarily translate to minimum reconstructed variance. This is illustrated by the numerical simulations I wrote to test predictions of the model on 2×2 systems. Initial results can not differentiate clearly between the currently used and newly proposed samplings.

I then derive a similar optimal sampling for the back projection step. This optimization is performed over reconstructed variance, and is therefore a more promising candidate for improving reconstruction quality. However, this requires access to variance of individual LOR counts, which can be difficult to measure. I coded up an iterative reconstruction for 2×2 systems. The results of numerical tests indicate that the theoretically optimal sampling does reduce reconstruction error.

In conclusion, initial numerical testing shows that the derived forward projection sampling performs better than simple uniform sampling, but slightly worse than the currently used activity weighted method. However, simulated results also indicate that the proposed back projection sampling offers an improvement over activity weighted reconstruction.

Chapter 1

System Description

1.1 PET Imaging

In Positron Emission Tomography (or PET), a positron emitting isotope is introduced to the examined system. This isotope emits positrons, which are annihilated upon coming into contact with electrons of surrounding atoms. Such an annihilation produces two or three photons depending on the spin state of the electron-positron system. However, three photon decays are a factor of 370 less frequent when incident particles have small relative velocity, as shown in [2]. This condition is satisfied in PET imaging, so three photon emissions can be disregarded. This means that detected decays emit two photons, which travel in opposite direction along a line. Once again, the non-collinearity of photons is neglected, as this study focuses on statistical optimization of Monte Carlo sampling and not particle transport. Emitted photon pairs are then subjected to coincidence detection using an array of photon detectors.

The above described simplified setup can be broken down into two main parts. The physical system itself has an activity distribution, which is the quantity PET imaging seeks to recover. While the coincidence detected measurement data is taken in Line of Response (LOR from now on) space. If physical space is discretized into voxels (volumetric elements), then a simple mathematical model can be formulated that connects measured data to real quantities.

$\boldsymbol{A}$, the system matrix describes the behaviour of photon pairs emitted from the observed system. The definition of the individual matrix elements can be written in a simple form. A_{ij} is the probability that a particle emitted by the j-th voxel is detected by the i-th LOR. This mathematical description is illustrated in figure 1.1

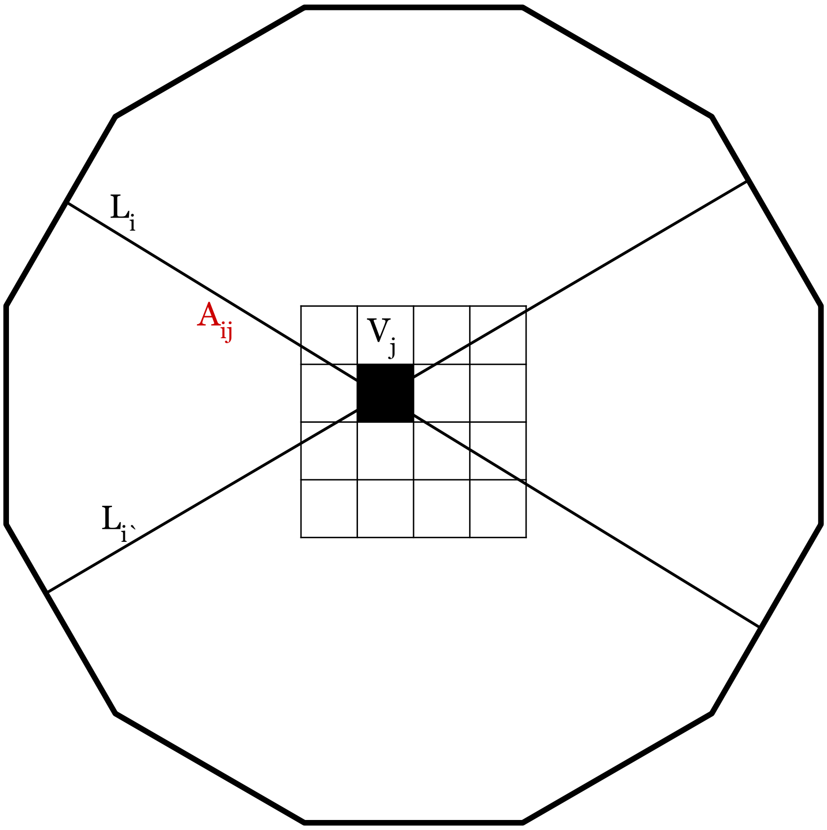

Figure 1.1: The real system is discretized into voxels (square grid in the centre), while the imaging device consists of detector panels (sides of the regular polygon). Each detector pair constitutes a LOR; and coincidence detection in the two detectors translates to a particle detection in their LOR. Two LORs are shown: L_i and $L_{i'}$. The system matrix element A_{ij} is the probability that a photon pair emitted from voxel V_j gets picked up by LOR L_i

1.2 Virtual Measurement Setup

The simplest possible system that allows for a meaningful reconstruction can be used as a starting point. This system consists of two voxels: V_A and V_B and two lines of response (LOR from now on) L_1 and L_2. The voxels have activities C_A and C_B respectively (however, to keep notation simple, from now on $c_A = C_A t$ and $c_B = C_B t$ decay counts will be used instead, where t is measurement time).

Particle transport is governed by a known system matrix $\boldsymbol{A}$, which in this simple case is a 2x2 square matrix.

To properly define the system, a physical model is also needed. Once again, the simplest possible one is sufficient. Each emission from a voxel produces exactly one photon pair. Furthermore, a photon upon interacting with a detector is removed from the system. Thus the possible outcomes of a photon pair emission are: the photons gets detected by one of the LORs or they leave the system unobserved.

Our simple system has now been completely defined, and it is now possible to formulate a mathematical description of its behaviour. A photon pair emitted from voxel j follows a multinomial distribution: with probability A_{1j} it is detected by L_1, with probability A_{2j} it is picked up by L_2 and with probability $1 - (A_{j1} + A_{j2})$ it escapes the system without detection. Therefore the whole system can be summarized in the following table:

	V_A	V_b
L_1	A_{11}	A_{12}
L_2	A_{21}	A_{22}
$\emptyset$	$1 - (A_{11} + A_{21})$	$1 - (A_{12} + A_{22})$

(1.2.0.1)

This model neglects one important property of the system: the inherent stochastic nature of particle emissions. This was left out to simplify the algebra involved in the derivations.

1.3 Simulation of the System

Many iterative reconstruction methods have to simulate the system which gives rise to the acquired data. To better understand the theoretical limits of such an approach, a simulation that reproduces the behaviour of our virtual setup will be used as the basis of our calculations. The implicitly assumed analogy between physical integrals and Monte Carlo sampled quantities is shown in [3].

The output of the measurement setup is the pair of LOR counts. Measured signals from L_1 and L_2 will be denoted by S_1 and S_2 respectively.

Assuming that upon detection each photon pair gives rise to a unit strength detection signal we can already write down the expected value (EV from now on) of our signals:

$$\mathbb{E}\left[\begin{pmatrix} S_1 \\ S_2 \end{pmatrix}\right] = \boldsymbol{A} \cdot \begin{pmatrix} c_A \\ c_B \end{pmatrix} \tag{1.3.0.2}$$

However, that in itself is not enough to run a simulation. Let us have N_A virtual particles emitted from V_A and N_B particles from V_B. To get the right expected LOR counts, these have to have weights w_A and w_B. The expected LOR counts from such a simulation are also simple to calculate:

$$\mathbb{E}\left[\begin{pmatrix} S_1 \\ S_2 \end{pmatrix}\right] = \boldsymbol{A} \cdot \begin{pmatrix} w_A N_A \\ w_B N_B \end{pmatrix} \tag{1.3.0.3}$$

The weights can be determined from the above equations, matching expected simulated and measured LOR counts:

$$\begin{pmatrix} c_A \\ c_B \end{pmatrix} = \begin{pmatrix} N_A w_A \\ N_B w_B \end{pmatrix} \tag{1.3.0.4}$$

and so $w_A = \frac{c_A}{N_A}$ and $w_b = \frac{c_B}{N_B}$

$N = N_A + N_B$, the total number of virtual particles in the simulation is a very important quantity that affects the quality of sample statistics.

Chapter 2

Problem Proposition

2.1 Overview

A general iterative reconstruction algorithm follows an iteration scheme such as the one in [4]:

$$x_j^{k+1} = x_j^k \frac{1}{\sum\limits_{s=1}^{l} A_{sj}} \sum_{i=1}^{l} A_{ij} \frac{y_i^m}{\sum\limits_{t=1}^{n} A_{it} x_t^k} \tag{2.1.0.1}$$

Where $\boldsymbol{A}$ is the system matrix (the first index runs over LORs and the second over voxels). l is the number of LORs and n the number of voxels in the system. x_j^k denotes the reconstructed activity in voxel m after k iterations. y_i^m is the measured count in LOR i.

The above expression is identical to a Richardson-Lucy deconvolution [5], as the measured LOR counts are given by the real activity concentrations convolved with the system matrix and some noise due to the statistical nature of nuclear decay.

The first major problem is the appearance of explicit system matrix elements in the expression. This is a problem, because $\boldsymbol{A}$ can be very large, and thus impractical to store. In a human PET system there can be $3 \cdot 10^8$ LORs and 10^4 voxels, as noted in [6]. The system matrix for such a device has $3 \cdot 10^{12}$ elements. Such a large matrix would be impractical to work with, so an alternative method to evaluate approximations to (2.1.0.1) is important.

One iteration in the reconstruction process can be separated into two steps: forward projection and backward projection. Forward projection is calculating expected LOR counts based on the latest estimates of voxel activities. Backwards projection is comparing these results to measured LOR counts and correcting the reconstructed activity values based on the differences in calculated and measured LOR counts.

In (2.1.0.1) the forward projection step is calculating

$$y_i^s = \sum_{t=1}^{n} A_{it} x_t^k \tag{2.1.0.2}$$

While the back projection is evaluating the whole expression:

$$x_j^{k+1} = x_j^k \sum_{i=1}^{l} \frac{A_{ij}}{\sum\limits_{s=1}^{l} A_{sj}} \frac{y_i^m}{y_i^s} \tag{2.1.0.3}$$

To avoid the necessity of storing the entire system matrix, a Monte Carlo approach may be used in both the forward and backwards projections. This is the method used in [7]. By taking a closer look at the expressions themselves, it can be seen that the forward projection poses a serious mathematical problem in a Monte Carlo simulation. In any realistic system, there is a non-zero probability that a photon gets absorbed, or leaves the system. This means that there is always a positive probability that a certain LOR measures zero particles in a Monte-Carlo simulation even if the physically measured count was non-zero. This means that the expression $z_i = \frac{y_i^m}{y_i^s}$, which is a random variable, can take on infinity as a value with positive probability. Such a random variable is impossible to work with numerically, so a different approach is necessary to perform the forward projection. It is interesting to note that the backwards projection never leads to this problem, as in the expression $\frac{A_{ij}}{\sum\limits_{s=1}^{l} A_{sj}}$ the denominator can only be zero if the numerator is zero as well; this avoids infinite values, but still allows for $\frac{0}{0}$ instances.

In short, the Monte-Carlo approach poses problems for both forward and back projections, but numerical treatment of the forward projection is more difficult due to the appearance of singular values.

2.2 Analytical Results

There are two common methods to avoid infinite values in the forward projection. LORs with zero simulated counts are either disregarded during the iteration step, or replaced by values from the previous step. The two cases are very similar, in that it simply disregards problematic results. The first approach simply replaces all singular z_i values by zeros, while the latter re-draws them according to the probability distribution until it finds a non-singular value.

If we consider a simple one voxel, one LOR system, some analytical results can be derived concerning the distribution of z_i values:

$$\begin{gathered}\overline{z_i} = y_i^m \overline{\left(\frac{1}{y_i^s}\right)} = y_i^m \sum_{k=1}^{n} \frac{1}{k} \binom{n}{k} p^k (1-p)^{n-k} = \\ y_i^m np (1-p)^{n-1} {}_pF_q \left(\{1,1,1-n\}; \{2,2\}; \frac{p}{p-1}\right)\end{gathered} \tag{2.2.0.4}$$

where p is the detection probability, n is the total number of virtual particles and ${}_pF_q$ is the generalized hypergeometric function defined in [8].

A similar expression can be derived for z_i^2:

$$\begin{gathered}\overline{z_i^2} = (y_i^m)^2 \overline{\left(\frac{1}{y_i^s}\right)^2} = (y_i^m)^2 \sum_{k=1}^{n} \left(\frac{1}{k}\right)^2 \binom{n}{k} p^k (1-p)^{n-k} = \\ (y_i^m)^2 np (1-p)^{n-1} {}_pF_q \left(\{1,1,1,1-n\}; \{2,2,2\}; \frac{p}{p-1}\right)\end{gathered} \tag{2.2.0.5}$$

Detailed derivation of the above results can be found in appendix A.2. From the above two equations, the expected variance of z_i could also be expressed using $\overline{\delta^2 z_i} = \overline{z_i^2} - (\overline{z_i})^2$. However, it is an unnecessarily complicated expression and therefore is omitted. Analogous formulae can be derived for the case when infinite values are re-generated instead of thrown away. These are almost identical to the above expressions and are therefore omitted.

While the existence of an analytical expression for the mean and variance are positive results, dealing with the generalized hypergeometric function is cumbersome and numerical evaluation is computationally intensive. Furthermore, even the simple two voxel, one LOR system proves to be too complicated for similar calculations; I was not able to find a closed analytical expression for mean and variance for such a setup. Negative moments of positive Binomial distributions are required in many studies, however as noted in [9], there are no known analytical methods for evaluating these moments.

In later chapters I will mostly deal with minimizing L2 errors, therefore the important expression is not total variance itself, but its various partial derivatives with respect to particle counts. However, these expressions are in no way simpler than the explicit formulae given above.

In conclusion, obtaining pure analytical results for iterative PET reconstruction is an unrealistic goal. Therefore either an approximation; or an alternative, more ad-hoc route has to be considered.

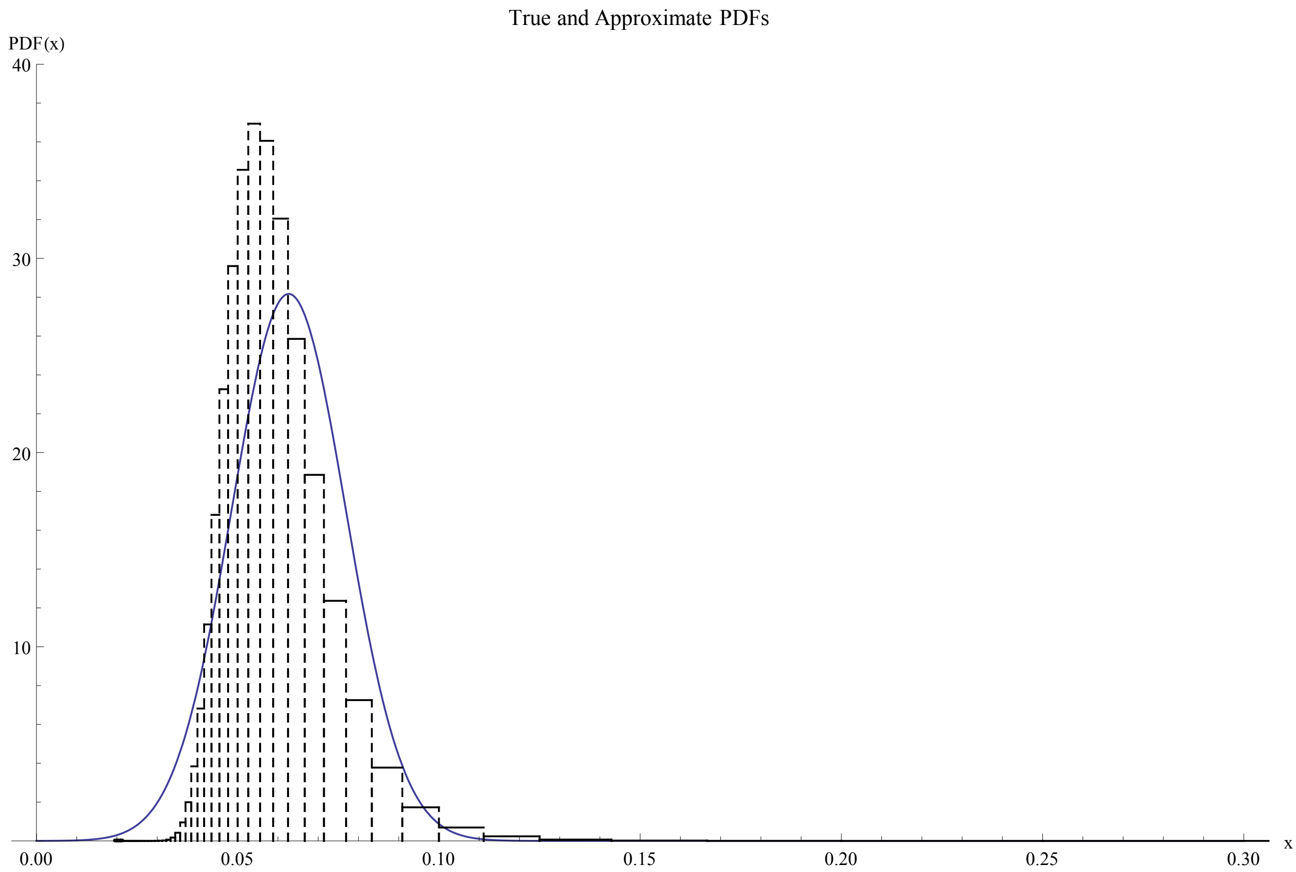

Figure 2.1: Comparison of probability distributions. The continuous line shows the normal approximation, while the dotted lines show a normalized histogram of the true distribution of values. In this case $n = 50$ and $p = \frac{1}{3}$

2.3 Normal Approximation

In the previous section, it was shown that general, analytical results are unreasonably difficult to obtain for the whole system.

While the distribution of z_i values is difficult to handle, once the singularities are removed, it has a finite mean and variance. There is a possibility of approximating such a distribution with a normal distribution with the same mean and variance. Such an approximation can work well in many cases, but can be very inaccurate in case of ill-behaved distributions. Figure 2.1 shows this approximation in case of a 1x1 system. The PDF of a normal distribution provides a reasonable fit to the idealized histogram, which indicates that such an approximation could lead to useful results.

Chapter 3

L_2 Optimal System Matrix

The conclusion of the previous chapter was that it is very difficult to derive purely analytical results for an iterative reconstruction scheme. In this chapter we investigate the effect of the system matrix on total reconstructed variance. If reconstructed variance is a slow-varying function of individual system matrix elements, that means that iterative reconstruction is a robust approach. However, if variance has a sharp minimum then performance is heavily dependent on minor details of the system.

3.1 Introduction

As discussed in the previous chapter, the general form of an iterative reconstruction algorithm is:

$$x_j^{k+1} = x_j^k \frac{1}{\sum\limits_{s=1}^{l} A_{sj}} \sum_{i=1}^{l} A_{ij} \frac{y_i^m}{\sum\limits_{t=1}^{n} A_{it} x_t^k} \tag{3.1.0.1}$$

Where $\boldsymbol{A}$ is the system matrix (the first index runs over LORs and the second over voxels). l is the number of LORs and n the number of voxels in the system. x_j^k denotes the reconstructed activity in voxel m after k iterations. y_i^m is the measured count in LOR i.

The above can be re-cast in a more convenient form:

$$x_j^{k+1} = x_j^k \sum_{i=1}^{l} \frac{A_{ij}}{\sum\limits_{s=1}^{l} A_{sj}} \frac{y_i^m}{\sum\limits_{t=1}^{n} A_{it} x_t^k} = x_j^k \sum_{i=1}^{l} w_{ij} z_i \tag{3.1.0.2}$$

where w_{ij} is the abstract weight factor which is given by the reduced system matrix element:

$$w_{ij} = \frac{A_{ij}}{\sum\limits_{s=1}^{l} A_{sj}} \tag{3.1.0.3}$$

and y_i is the ratio of measured and expected counts in LOR i:

$$z_i = \frac{y_i^m}{\sum\limits_{t=1}^{n} A_{it} x_t^k} \tag{3.1.0.4}$$

The weights satisfy the following normalization property:

$$\sum_{i=1}^{l} w_{ij} = 1 \tag{3.1.0.5}$$

To achieve optimal reconstruction in L_2 norm, the total variance reconstructed of quantities has to be minimized. Assuming previous reconstructed values and abstract weights to be constants without variance, total variance can be expressed as:

$$\overline{\delta^2 x_j^{k+1}} = \left(x_j^k\right)^2 \sum_{i=1}^{l} w_{ij}^2 \overline{\delta^2 z_i} \tag{3.1.0.6}$$

We know that the distribution of y_i can not be handled analytically, but we can assume that the distribution has a known variance and proceed with that. From now on we will use the values:

$$\overline{\delta^2 z_i} = \sigma_i^2 \tag{3.1.0.7}$$

So now we have

$$\overline{\delta^2 x_j^{k+1}} = \left(x_j^k\right)^2 \sum_{i=1}^{l} w_{ij}^2 \sigma_i^2 \tag{3.1.0.8}$$

This has to be minimal to achieve L2 optimal reconstruction.

There are two possible ways to proceed. Inter-voxel correlations can either be ignored or taken into account. The total variance neglecting correlations is:

$$D_n^2 = \sum_{j=1}^{n} \overline{\delta^2 x_j^{k+1}} = \sum_{j=1}^{n} \left(\left(x_j^k\right)^2 \sum_{i=1}^{l} w_{ij}^2 \sigma_i^2 \right) \tag{3.1.0.9}$$

And the expression accounting for covariances is:

$$D_c^2 = \overline{\delta^2 \sum_{j=1}^{n} x_j^{k+1}} \tag{3.1.0.10}$$

This expression can be refined using the following:

$$\sum_{j=1}^{n} x_j^{k+1} = \sum_{j=1}^{n} \left(x_j^k \sum_{i=1}^{l} w_{ij} z_i \right) = \sum_{i=1}^{l} \left(z_i \sum_{j=1}^{n} x_j^k w_{ij} \right) \tag{3.1.0.11}$$

And so (3.1.0.12) becomes:

$$D_c^2 = \overline{\delta^2 \sum_{i=1}^{l} \left(z_i \sum_{j=1}^{n} x_j^k w_{ij} \right)} = \sum_{i=1}^{l} \left(\sigma_i^2 \left(\sum_{j=1}^{n} w_{ij} x_j^k \right)^2 \right) \tag{3.1.0.12}$$

3.2 Equations to Solve

Using the variance formulae from the previous section, it is possible to derive the necessary equations that lead to minimal total variance.

The previously stated normalization condition has to be incorporated as a constraint:

$$\sum_{i=1}^{l} w_{ij} = 1 \tag{3.2.0.13}$$

Neglecting correlations, we have:

$$D_n^2 = \sum_{j=1}^{n} \left(\left(x_j^k\right)^2 \sum_{i=1}^{l} w_{ij}^2 \sigma_i^2 \right) \tag{3.2.0.14}$$

Applying the method of Lagrange multipliers to the above equations, we can define a function that can be minimized without constraints:

$$G_n = \sum_{j=1}^{n} \left(\left(x_j^k\right)^2 \sum_{i=1}^{l} w_{ij}^2 \sigma_i^2 \right) + \sum_{j=1}^{n} \lambda_j \left(1 - \sum_{i=1}^{l} w_{ij} \right) \tag{3.2.0.15}$$

And from this, minimizing with respect to every variable (λ_s, w_{qr}) we get the following system of equations:

$$\partial_{\lambda_s} G_n = 0 = 1 - \sum_{i=1}^{l} w_{is} \tag{3.2.0.16}$$

which of course is equivalent to the normalization constraint.

The partial derivatives with respect to weights yield:

$$\partial_{w_{qr}} G_n = 0 = 2 w_{qr} \left(x_r^k\right)^2 \sigma_q^2 - \lambda_r \tag{3.2.0.17}$$

In a completely analogous manner, the results accounting for correlations can be found. The function to be minimized:

$$G_c = \sum_{i=1}^{l} \left(\sigma_i^2 \left(\sum_{j=1}^{n} w_{ij} x_j^k \right)^2 \right) + \sum_{j=1}^{n} \lambda_j \left(1 - \sum_{i=1}^{l} w_{ij} \right) \tag{3.2.0.18}$$

Once again we recover the normalization condition:

$$\partial_{\lambda_s} G_c = 0 = 1 - \sum_{i=1}^{l} w_{is} \tag{3.2.0.19}$$

And the remaining equations are:

$$\partial_{w_{qr}} G_c = 0 = 2\sigma_q^2 x_r^k \left(\sum_{t=1}^{n} w_{qt} x_t^k \right) - \lambda_r \tag{3.2.0.20}$$

3.3 2x2 Systems

Calculations for the 2x2 system are tractable, but still illustrate some important properties of larger systems. Therefore a detailed solution will be included for such systems.

3.3.1 Neglecting Correlations

When correlations are neglected, we get the following system of equations from (3.2.0.15):

$$\begin{pmatrix} \lambda_1 \\ \lambda_1 \\ \lambda_2 \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} 2w_{11} \left(x_1^k\right)^2 \sigma_1^2 \\ 2w_{21} \left(x_1^k\right)^2 \sigma_2^2 \\ 2w_{12} \left(x_2^k\right)^2 \sigma_1^2 \\ 2w_{22} \left(x_2^k\right)^2 \sigma_2^2 \end{pmatrix} \tag{3.3.1.1}$$

and the constraints:

$$\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} w_{11} + w_{21} \\ w_{12} + w_{22} \end{pmatrix} \tag{3.3.1.2}$$

From (3.3.1.1), by adding σ_2^2 times the first equation to σ_1^2 times the second we get:

$$\lambda_1 \left(\sigma_1^2 + \sigma_2^2\right) = 2 \left(w_{11} + w_{21}\right) \left(x_1^k\right)^2 \sigma_1^2 \sigma_2^2 \tag{3.3.1.3}$$

applying the same the third and fourth equation yields:

$$\lambda_2 \left(\sigma_1^2 + \sigma_2^2\right) = 2 \left(w_{12} + w_{22}\right) \left(x_2^k\right)^2 \sigma_1^2 \sigma_2^2 \tag{3.3.1.4}$$

Substituting in from the constraints, the above two equations become:

$$\begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} 2 \left(x_1^k\right)^2 \frac{1}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} \\ 2 \left(x_2^k\right)^2 \frac{1}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} \end{pmatrix} \tag{3.3.1.5}$$

And finally substituting back into (3.3.1.1), we get:

$$\begin{pmatrix} w_{11} \\ w_{21} \\ w_{12} \\ w_{22} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sigma_1^2} \frac{1}{\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}} \\ \frac{1}{\sigma_2^2} \frac{1}{\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}} \\ \frac{1}{\sigma_1^2} \frac{1}{\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}} \\ \frac{1}{\sigma_2^2} \frac{1}{\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}} \end{pmatrix} \tag{3.3.1.6}$$

3.3.2 Including Correlations

(3.2.0.18) gives the following equations:

$$\begin{pmatrix} \lambda_1 \\ \lambda_1 \\ \lambda_2 \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} 2x_1^k\sigma_1^2\left(x_1^k w_{11} + x_2^k w_{12}\right) \\ 2x_1^k\sigma_2^2\left(x_1^k w_{21} + x_2^k w_{22}\right) \\ 2x_2^k\sigma_1^2\left(x_1^k w_{11} + x_2^k w_{12}\right) \\ 2x_2^k\sigma_2^2\left(x_1^k w_{21} + x_2^k w_{22}\right) \end{pmatrix} \tag{3.3.2.1}$$

and constraints:

$$\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} w_{11} + w_{21} \\ w_{12} + w_{22} \end{pmatrix} \tag{3.3.2.2}$$

From the first two equations in (3.3.2.1), we get:

$$\sigma_1^2\left(x_1^k w_{11} + x_2^k w_{12}\right) = \sigma_2^2\left(x_1^k w_{21} + x_2^k w_{22}\right) \tag{3.3.2.3}$$

Substituting this result into the third equations yields:

$$\lambda_2 = 2x_2^k\sigma_2^2\left(x_1^k w_{21} + x_2^k w_{22}\right) \tag{3.3.2.4}$$

which is the fourth equations itself. This means that the system of equations is redundant and thus does not have a unique solution. This degeneracy is further discussed in the next section. Still, we can go a bit further. Substituting the constraints into (3.3.2.3) we get:

$$\sigma_1^2\left(x_1^k w_{11} + x_2^k w_{12}\right) = \sigma_2^2\left(x_1^k\left(1 - w_{11}\right) + x_2^k\left(1 - w_{12}\right)\right) \tag{3.3.2.5}$$

which can be rearranged to yield:

$$x_1^k w_{11} + x_2^k w_{12} = \frac{1}{\sigma_1^2}\frac{x_1^k + x_2^k}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} \tag{3.3.2.6}$$

this is the equation of a straight line in w_{11} and w_{12}.

3.4 Larger Systems

The general solution for $n \times l$ systems is given here.

3.4.1 Neglecting Correlations

Based on the results from the 2x2 system, it is relatively simple to guess the general solution:

$$\lambda_r = 2\left(x_r^k\right)^2 \frac{1}{\sum\limits_{t=1}^{l} \frac{1}{\sigma_t^2}} \tag{3.4.1.1}$$

and

$$w_{qr} = \frac{\frac{1}{\sigma_q^2}}{\sum\limits_{t=1}^{l} \frac{1}{\sigma_t^2}} \tag{3.4.1.2}$$

These weights trivially satisfy the normalization constraint (3.2.0.13), and the following equations as well:

$$\partial_{w_{qr}} G_n = 0 = 2w_{qr}\left(x_r^k\right)^2 \sigma_q^2 - \lambda_r \tag{3.4.1.3}$$

Therefore the initial guesses must coincide with the solution. It is important that the numerical evaluation of the solution is also feasible. It can be performed without extensive CPU or memory cost if the common term $\sum\limits_{t=1}^{l} \frac{1}{\sigma_t^2}$ is calculated and stored in advance. This provides a way to estimate the difficulty of reconstruction for the particular system considered.

3.4.2 Including Correlations

From subsection 3.2, we have the normalization (3.2.0.13) and

$$\partial_{w_{qr}} G_c = 0 = 2\sigma_q^2 x_r^k \left(\sum_{t=1}^{n} w_{qt} x_t^k\right) - \lambda_r \tag{3.4.2.1}$$

the above can be re-arranged to read:

$$\frac{\lambda_r}{\sigma_q^2} = 2x_r^k \left(\sum_{t=1}^{n} w_{qt} x_t^k\right) \tag{3.4.2.2}$$

summing over all LORs gives:

$$\sum_{q=1}^{l} \frac{\lambda_r}{\sigma_q^2} = \sum_{q=1}^{l} \left(2x_r^k \left(\sum_{t=1}^{n} w_{qt} x_t^k \right) \right) = 2x_r^k \sum_{t=1}^{n} \left(x_t^k \sum_{q=1}^{l} w_{qt} \right) = 2x_r^k \sum_{t=1}^{n} x_t^k \qquad (3.4.2.3)$$

where the last equation follows from the normalization condition. The above can be re-cast into:

$$\lambda_r = 2x_r^k \frac{1}{\sum\limits_{q=1}^{l} \frac{1}{\sigma_q^2}} \sum_{t=1}^{n} x_t^k \qquad (3.4.2.4)$$

Substituting this into (3.4.2.1) we get:

$$2\sigma_q^2 x_r^k \left(\sum_{t=1}^{n} w_{qt} x_t^k \right) = 2x_r^k \frac{1}{\sum\limits_{q=1}^{l} \frac{1}{\sigma_q^2}} \sum_{t=1}^{n} x_t^k \qquad (3.4.2.5)$$

simplifying leads to:

$$\sum_{t=1}^{n} w_{qt} x_t^k = \frac{1}{\sigma_q^2} \frac{1}{\sum\limits_{s=1}^{l} \frac{1}{\sigma_s^2}} \sum_{t=1}^{n} x_t^k \qquad (3.4.2.6)$$

Together equations (3.4.2.2) and (3.4.2.6) are equivalent to (3.4.2.1), therefore this is all we can determine about the position of the minimum if correlations are included. This means that the minimum is a multi-dimensional surface, therefore the effect of individual system matrix elements on total reconstructed variance is relatively small. This is an important theoretical result concerning iterative reconstruction schemes.

Results derived neglecting and including correlations differ greatly. This discrepancy demonstrates that correlations between individual LOR counts have a significant impact on variances. Therefore, neglecting these correlations could lead to false results and improper optimizations.

3.5 2x2 Simulation Results

A simulation was used to check the theoretical results described in this chapter. It determined the variance of a weighted sum of two normally distributed random variables based on the weights used. For the depicted results, 50 divisions were made in both w_{11} and w_{12}. Each point on the graph was calculated as the sample variance of 100000 individual runs (random numbers), and the following values were used for $\sigma_1, \sigma_2, x_1^k, x_2^k$ in turn: $4, 5, 3, 2$. The results are shown on figures 3.1 to 3.4.

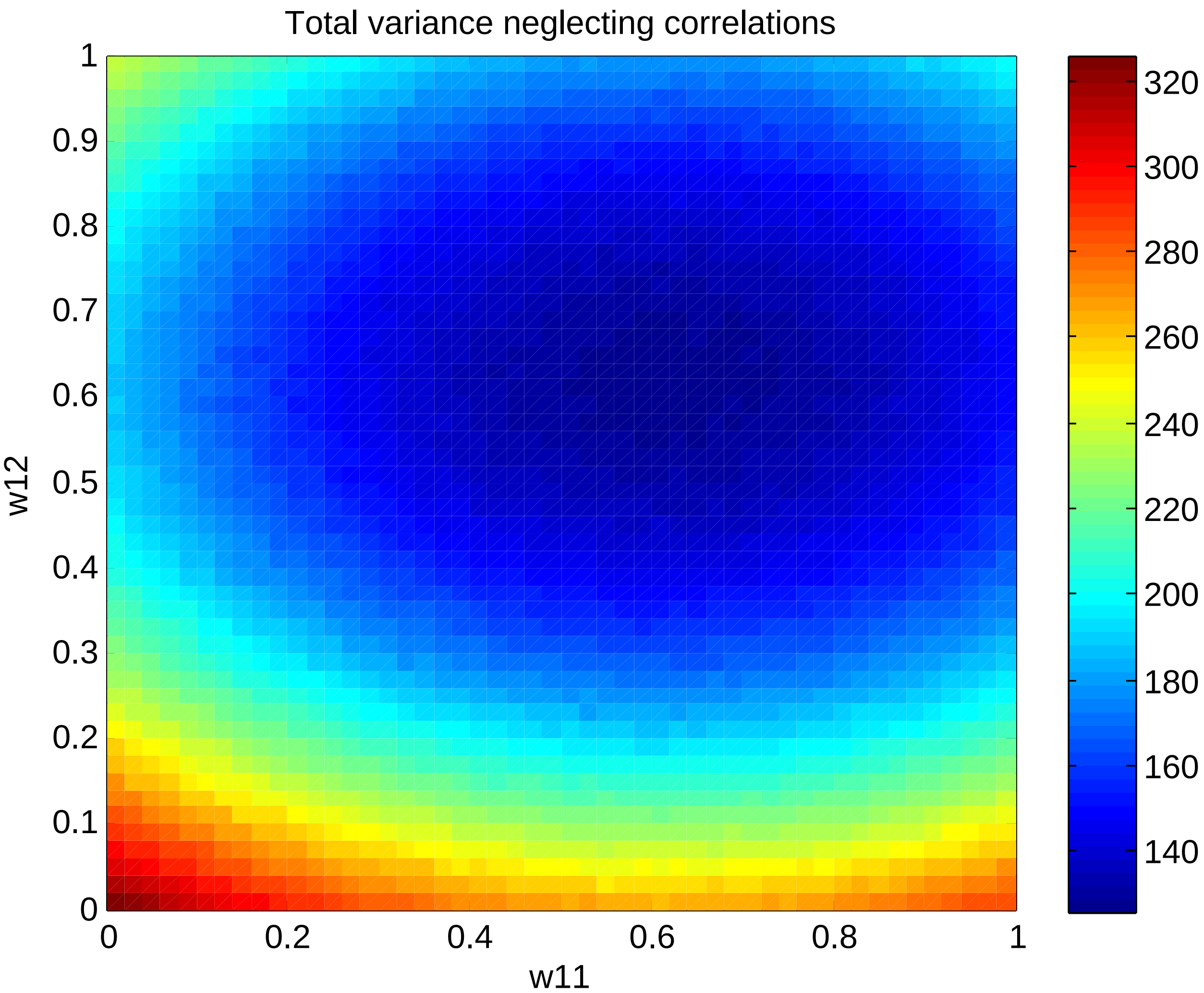

Figure 3.1: Total variance neglecting correlations as a function of reduced system matrix elements. As predicted by previous calculations, there appears to be a well-defined minimum.

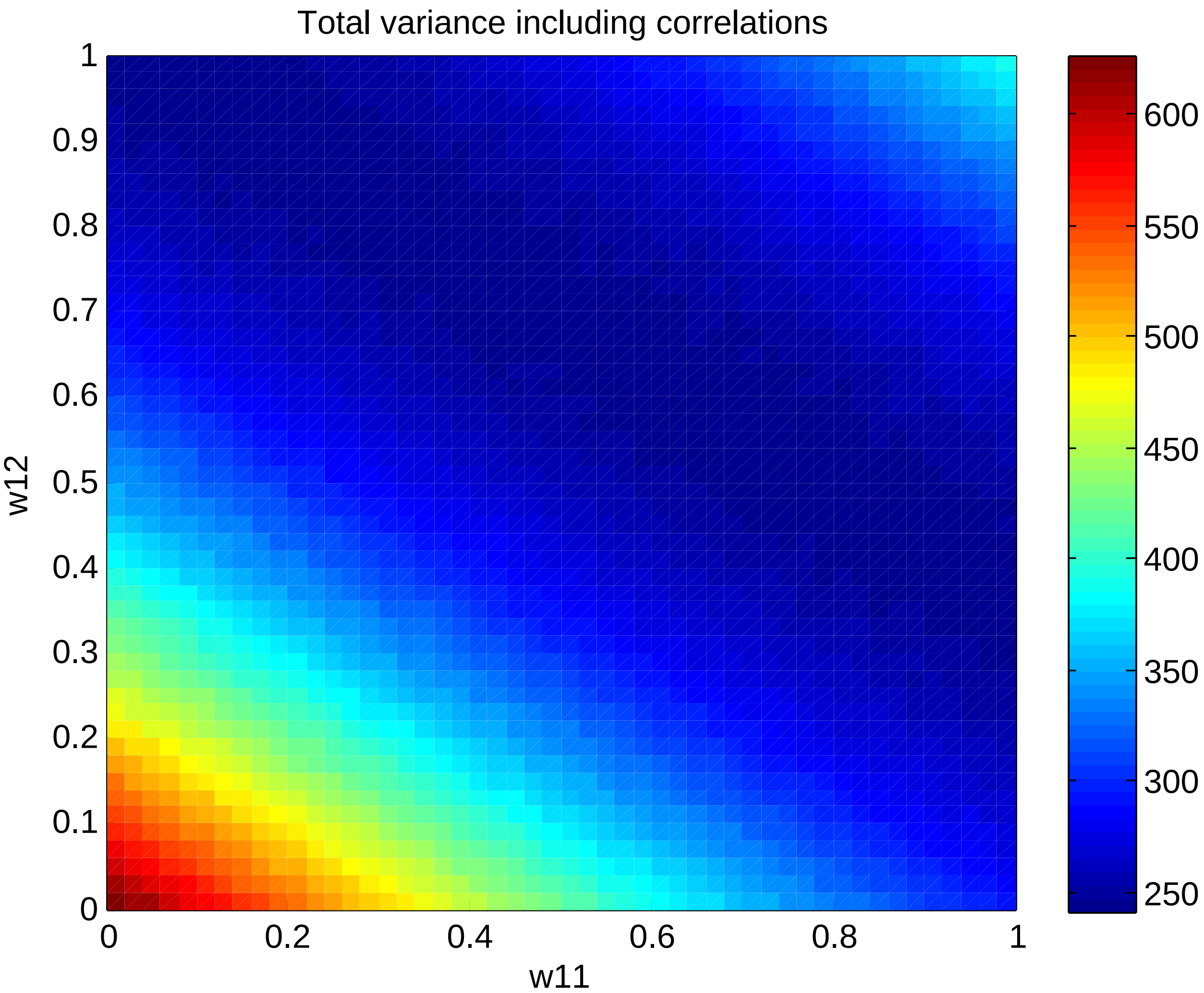

Figure 3.2: Total variance including correlations as a function of reduced system matrix elements. As predicted by theory, the true minimum is degenerate, indicating that a wide range of system matrices allow for L_2 optimal reconstruction. In agreement with (3.3.2.6), the minimum follows a straight line.

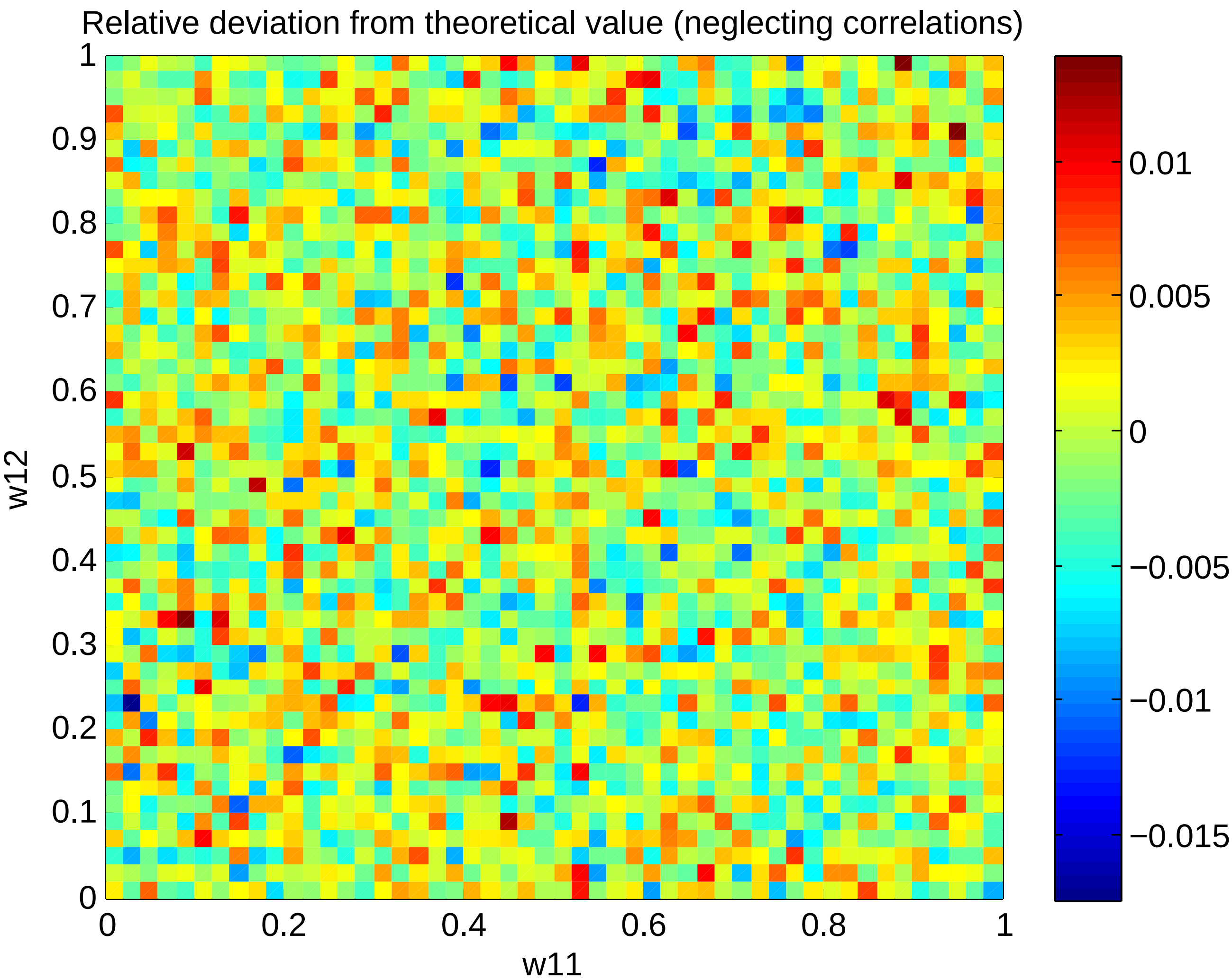

Figure 3.3: Relative difference between simulation and theoretical value; correlations are neglected. The plot shows that there are no noticeable systematic differences between simulated and calculated results.

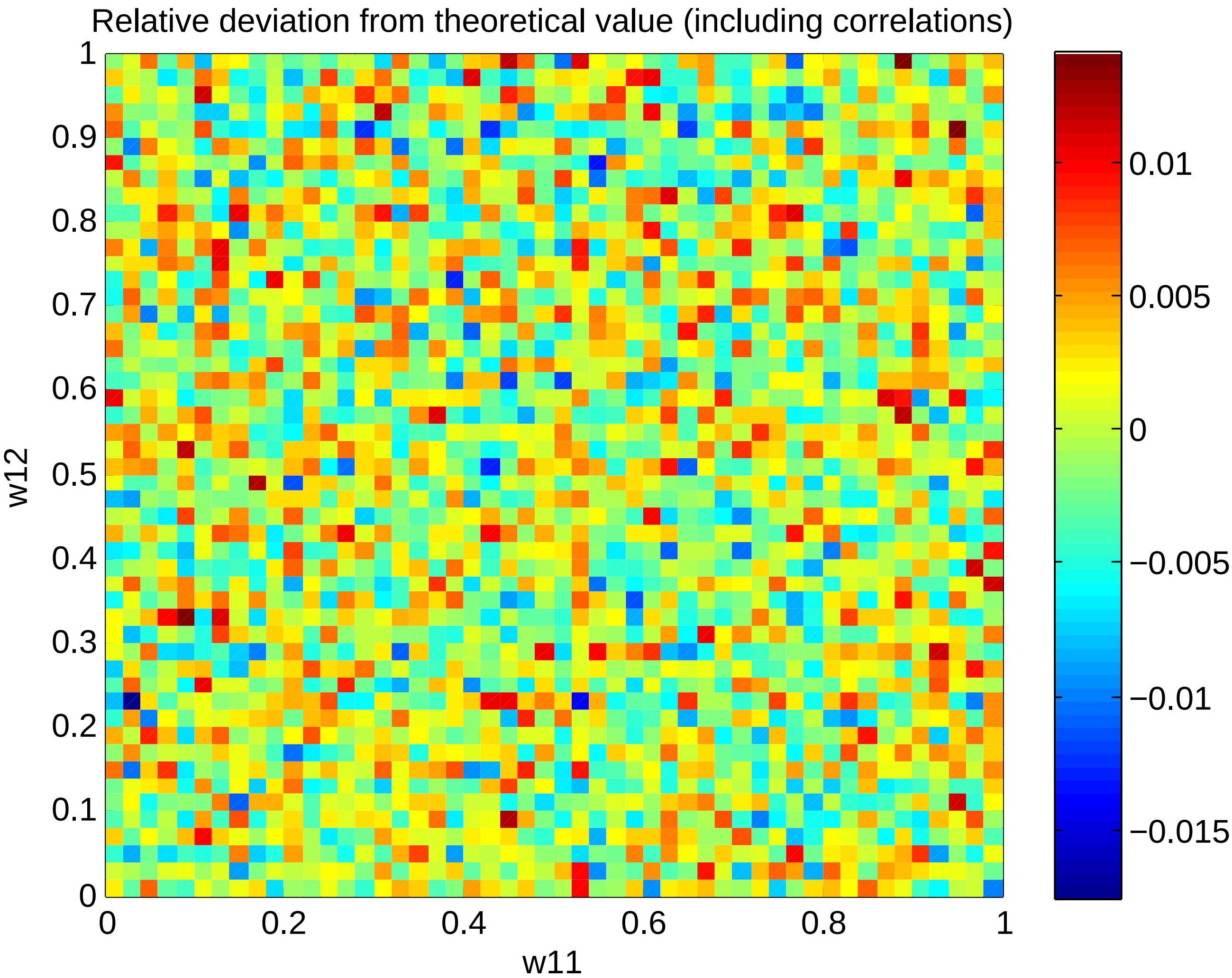

Figure 3.4: Relative difference between simulation and theoretical value; correlations are included. The plot shows that there are no noticeable systematic differences between simulated and calculated results.

Chapter 4

Optimizing Forward Projection for 2×2 Systems

While results discussed in the previous chapter are interesting, they do not help in building a better reconstruction algorithm. It was demonstrated in chapter 2 that the main source of mathematical difficulties is the forward-projection step.

When the system matrix can not be stored in memory, Monte Carlo simulations are used to approximate LOR counts. These counts are then used to correct voxel activity estimates. This is a crucial step in the reconstruction process, and is therefore a good candidate for optimization. In the following two chapters I will outline a computationally feasible method for ensuring minimal variance of simulated LOR counts.

Most of the following results will concern true variances, but Monte Carlo variances for the test system can be useful for simulation purposes. These results can be found in appendix A.1.

4.1 Variances

4.1.1 Single LOR Variance

S_1 is the number of photon pairs detected in L_1. It consists of two parts, photons arriving from voxel V_A and those from V_B. Denoting the number of detected particle pairs from V_A by i and the number from V_B by j, we can write:

$$S_1(i, j) = w_A i + w_B j \tag{4.1.1.1}$$

To determine the expected value, the probability of detecting i and j particle pairs from V_A and V_B respectively is also necessary. The behaviour of particles emitted from a specific voxel (eg. detected by D1 or leaving the system) are determined by a multinomial distribution dependent on the system matrix $\boldsymbol{A}$. Since particle emissions in V_A and V_B are independent, the overall probability of the pair (i, j) is simply the product of two binomial distributions.

$$\begin{aligned} \mathrm{P}\left(\#V_A \to L_1 = i, \#V_B \to L_1 = j\right) &= \mathrm{P}\left(\#V_A \to L_1 = i\right) \cdot \mathrm{P}\left(\#V_B \to L_1 = j\right) = \\ &= \left(\binom{N_A}{i} A_{11}^i (1 - A_{11})^{N_A - i}\right) \cdot \left(\binom{N_B}{j} A_{12}^j (1 - A_{12})^{N_B - j}\right) \end{aligned} \tag{4.1.1.2}$$

Variance can be calculated according to its definition:

$$\overline{\delta^2 S_1} = \overline{S_1^2} - \overline{S_1}^2 \tag{4.1.1.3}$$

$\overline{S_1}$ is known, however to determine $\overline{S_1^2}$, the following sum has to be calculated:

$$\overline{S_1^2} = \sum_{i=0}^{N_A} \sum_{j=0}^{N_B} (w_A i + w_B j)^2 \left(\binom{N_A}{i} A_{11}^i (1 - A_{11})^{N_A - i}\right) \cdot \left(\binom{N_B}{j} A_{12}^j (1 - A_{12})^{N_B - j}\right) \tag{4.1.1.4}$$

While it is possible to evaluate the above sum by purely algebraic means, a simple statistical argument can be used instead to quickly find the variance.

S_1 is the weighted sum of random variables drawn from two independent binomial distributions, thus it is a random variable from the distribution $\mathcal{S}_1$ given by:

$$\mathcal{S}_1 = w_A \mathcal{B}_{1A} + w_B \mathcal{B}_{1B} \tag{4.1.1.5}$$

where $\mathcal{B}_{1A} \sim \mathcal{B}(N_A, A_{11})$, $\mathcal{B}_{1B} \sim \mathcal{B}(N_B, A_{12})$ and $\mathcal{B}(N, p)$ denotes a binomial distribution with N trials and probability p. From here, evaluating EV and variance of $\mathcal{S}_1$ becomes trivial:

$$\mathbb{E}[\mathcal{S}_1] = w_A \mathbb{E}[\mathcal{B}(N_A, A_{11})] + w_B \mathbb{E}[\mathcal{B}(N_B, A_{12})] = w_A N_A A_{11} + w_B N_B A_{12} \tag{4.1.1.6}$$

and

$$\mathbb{D}^2[\mathcal{S}_1] = w_A^2 \mathbb{D}^2[\mathcal{B}(N_A, A_{11})] + w_B^2 \mathbb{D}^2[B(N_B, A_{12})] = w_A^2 N_A A_{11}(1 - A_{11}) + w_B^2 N_B A_{12}(1 - A_{12}) \tag{4.1.1.7}$$

which is the variance of S_1.

By the same argument and calculations, the expected variance of S_2 is:

$$\overline{\delta^2 S_2} = \mathbb{D}^2[\mathcal{S}_2] = w_A^2 N_A A_{21}(1 - A_{21}) + w_B^2 N_B A_{22}(1 - A_{22}) \tag{4.1.1.8}$$

4.1.2 Covariance

Calculating the covariance between S_1 and S_2 is similar to calculating individual variances. Definitions of important quantities: i_1 is the number of photon pairs detected by L_1 from V_A, j_1: particle pairs from V_B detected by L_1. And similarly i_2: particle pairs from V_A to L_2 and j_2: photon pairs from V_B to L_2

The definition of the covariance:

$$\overline{\delta^2 S_1 S_2} = \overline{S_1 S_2} - \overline{S_1} \cdot \overline{S_2} \tag{4.1.2.1}$$

Since particles from V_A and V_B are independent, we can write the probability mass function as:

$$\begin{aligned} &\mathrm{P}\left(\#V_A \to L_1 = i_1, \#V_A \to L_2 = i_2, \#V_B \to L_1 = j_1, \#V_B \to L_1 = j_2\right) = \\ &\mathrm{P}\left(\#V_A \to L_1 = i_1, \#V_A \to L_2 = i_2\right) \cdot \mathrm{P}\left(\#V_B \to L_1 = j_1, \#V_B \to L_1 = j_2\right) \end{aligned} \tag{4.1.2.2}$$

Knowing that photon pairs emitted from a voxel follow a multinomial distribution, we can write:

$$\begin{aligned} &\mathrm{P}\left(\#V_A \to L_1 = i_1, \#V_A \to L_2 = i_2\right) = \\ &\frac{N_A!}{i_1! i_2! (N_A - i_1 - i_2)!} A_{11}^{i_1} A_{21}^{i_2} \left(1 - (A_{11} + A_{21})\right)^{N_A - i_1 - i_2} = \mathrm{M}\left(A_{11}, A_{21}, N_A, i_1, i_2\right) \end{aligned} \tag{4.1.2.3}$$

and

$$\begin{aligned} &\mathrm{P}\left(\#V_B \to L_1 = j_1, \#V_B \to L_2 = j_2\right) = \\ &\frac{N_B!}{j_1! j_2! (N_B - j_1 - j_2)!} A_{12}^{j_1} A_{22}^{j_2} \left(1 - (A_{12} + A_{22})\right)^{N_B - j_1 - j_2} = \mathrm{M}\left(A_{12}, A_{22}, N_B, j_1, j_2\right) \end{aligned} \tag{4.1.2.4}$$

Here $\mathrm{M}(x_1, x_2, N, p_1, p_2)$ denotes the probability mass function of a multinomial distribution with three outcomes, arguments related to the third bin have been dropped without loss of generality. Since $\overline{S_1}$ and $\overline{S_2}$ are known, only $\overline{S_1 S_2}$ has to be determined. From the above definitions:

$$S_1S_2 = (w_Ai_1 + w_Bj_1)(w_Ai_2 + w_Bj_2) \tag{4.1.2.5}$$

and so:

$$\overline{S_1S_2} = \sum_{i_1=0}^{N_A}\sum_{i_2=0}^{N_A-i_1}\sum_{j_1=0}^{N_B}\sum_{j_2=0}^{N_B-j_1}(w_Ai_1+w_Bj_1)(w_Ai_2+w_Bj_2)\mathrm{M}\left(A_{11},A_{21},N_A,i_1,i_2\right)\mathrm{M}\left(A_{12},A_{22},N_B,j_1,j_2\right) \tag{4.1.2.6}$$

Once again it is possible to evaluate the sum relying on algebraic means, but there is a simpler way using certain results from probability theory. As in the previous section, the distributions of S_1 and S_2 can be written as:

$$\mathcal{S}_1 = w_A\mathcal{B}_{1A} + w_B\mathcal{B}_{1B} \tag{4.1.2.7}$$

and

$$\mathcal{S}_2 = w_A\mathcal{B}_{2A} + w_B\mathcal{B}_{2B} \tag{4.1.2.8}$$

where $\mathcal{B}_{1A} \sim \mathcal{B}(N_A, A_{11})$, $\mathcal{B}_{1B} \sim \mathcal{B}(N_B, A_{12})$, $\mathcal{B}_{2A} \sim \mathcal{B}(N_A, A_{21})$, $\mathcal{B}_{2B} \sim \mathcal{B}(N_B, A_{22})$ and $\mathcal{B}(N,p)$ denotes a binomial distribution with N trials and probability p.

Using this formulation, the covariance in question can be written:

$$\overline{\delta^2S_1S_2} = w_A^2\mathbb{D}^2[\mathcal{B}_{1A},\mathcal{B}_{2A}] + w_B^2\mathbb{D}^2[\mathcal{B}_{1B},\mathcal{B}_{2B}] + w_Aw_B\left(\mathbb{D}^2[\mathcal{B}_{1A},\mathcal{B}_{2B}] + \mathbb{D}^2[\mathcal{B}_{1B},\mathcal{B}_{2A}]\right) \tag{4.1.2.9}$$

Distributions that belong to different voxels are independent, but those that carry the same voxel index are correlated. Using the known covariance matrix of the multinomial distribution (off diagonal elements are of the form $-Np_ip_j$, as in [10]) the covariance can be evaluated:

$$\overline{\delta^2S_1S_2} = w_A^2\mathbb{D}^2[\mathcal{B}_{1A},\mathcal{B}_{2A}] + w_B^2\mathbb{D}^2[\mathcal{B}_{1B},\mathcal{B}_{2B}] = -w_A^2N_AA_{11}A_{21} - w_B^2N_BA_{12}A_{22} \tag{4.1.2.10}$$

4.2 Total Signal Variance

Our signal is $S_1 + S_2$, therefore the total signal variance is given by:

$$\overline{\delta^2(S_1+S_2)} = \overline{\delta^2S_1} + \overline{\delta^2S_2} + 2\overline{\delta^2S_1S_2} \tag{4.2.0.11}$$

Substituting in from equations (4.1.1.7), (4.1.1.8) and (4.1.2.10) we get:

$$\overline{\delta^2(S_1+S_2)} = w_A^2 N_A A_{11}(1-A_{11}) + w_B^2 N_B A_{12}(1-A_{12}) + \\ w_A^2 N_A A_{21}(1-A_{21}) + w_B^2 N_B A_{22}(1-A_{22}) - 2\left(w_A^2 N_A A_{11} A_{21} + w_B^2 N_B A_{12} A_{22}\right) \tag{4.2.0.12}$$

After some re-arrangement, the above can be cast in the form:

$$\overline{\delta^2(S_1+S_2)} = w_A^2 N_A \left(1-(A_{11}+A_{21})\right)(A_{11}+A_{21}) + w_B^2 N_B \left(1-(A_{12}+A_{22})\right)(A_{12}+A_{22}) \tag{4.2.0.13}$$

This can be re-written in a more convenient form:

$$\overline{\delta^2(S_1+S_2)} = w_A^2 N_A (1-P_A) P_A + w_B^2 N_B (1-P_B) P_B \tag{4.2.0.14}$$

where $P_A = A_{11} + A_{21}$ is the probability that a photon pair emitted from voxel V_A gets picked up by one of the LORs and likewise $P_B = A_{12} + A_{22}$ is the probability that a particle pair from V_B gets picked up.

4.3 Alternative Derivation

While the above derivation for total measured variance is sound and natural in a sense, it is complicated an unfeasible to apply for larger systems. Therefore it is important to have an alternative solution that is easier to generalize. To this end, let us think of the total measured signal in a different way. While it is apparent that the total measured signal is the sum of all individual LOR counts, it can also be calculated by summing contributions from individual voxels. Once again denoting the number of virtual particle pairs going from V_A to L_1 as i_1, from V_B to L_1: j_1, V_A to L_2: i_2, V_B to L_2: j_2

$$S_1+S_2 = (w_A i_1 + w_B j_1) + (w_A i_2 + w_B j_2) = w_A(i_1+i_2) + w_B(j_1+j_2) = S_A + S_B \tag{4.3.0.15}$$

where S_A and S_B are the total LOR counts caused by the individual voxels (ie. S_A is the total signal that would be observed if V_B was completely removed from the system).

S_A and S_B are much simpler to handle than S_1 and S_2 owing to their simpler distributions. $\mathcal{S}_A$, the distribution from which S_A is drawn, is a weighted sum of two independent binomial distributions, as in (4.1.2.7) and (4.1.2.8) :

$$\mathcal{S}_A = w_A\mathcal{B}_{1A}+w_A\mathcal{B}_{2A} = w_A\left(\mathcal{B}(N_A, A_{11}) + \mathcal{B}(N_A, A_{21})\right) = w_A\mathcal{B}(N_A, A_{11}+A_{21}) \quad (4.3.0.16)$$

The above result is also easy to derive from the system itself. A virtual photon pair from V_A is either detected by one of the LORs and results in a w_A strength measured signal, or escapes the system. Therefore in terms of total measured signal, the only relevant information is whether the particles get detected or not. Thus we can write:

$$\mathcal{S}_A = w_A\mathcal{B}(N_A, P_A) \quad (4.3.0.17)$$

This is exactly the same as the above result. Similarly we get:

$$\mathcal{S}_B = w_B\mathcal{B}(N_B, P_B) \quad (4.3.0.18)$$

The variance of the binomial distribution $\mathbb{D}^2[\mathcal{B}(N, p)] = Np(1 - p)$ as in [10], is well known and using this, we can write:

$$\mathbb{D}^2[\mathcal{S}_A] = w_A^2\mathbb{D}^2[\mathcal{B}(N_A, P_A)] = w_A^2 N_A P_A(1 - P_A) \quad (4.3.0.19)$$

and

$$\mathbb{D}^2[\mathcal{S}_B] = w_B^2\mathbb{D}^2[\mathcal{B}(N_B, P_B)] = w_B^2 N_B P_B(1 - P_B) \quad (4.3.0.20)$$

Since decays and subsequent particle paths in different voxels are independent of one another, S_A and S_B are also independent. Since $S_1 + S_2 = S_A + S_B$, we can write the total variance as:

$$\overline{\delta^2(S_1 + S_2)} = \overline{\delta^2(S_A + S_B)} = \mathbb{D}^2[\mathcal{S}_A] + \mathbb{D}^2[\mathcal{S}_B] = w_A^2 N_A P_A(1 - P_A) + w_B^2 N_B P_B(1 - P_B) \quad (4.3.0.21)$$

which is exactly what we had before.

4.4 Minimum Total Variance

Previously we have shown that total variance of the LOR counts is:

$$\overline{\delta^2(S_1 + S_2)} = w_A^2 N_A\left(1 - P_A\right)P_A + w_B^2 N_B\left(1 - P_B\right)P_B \quad (4.4.0.22)$$

N_A and w_A are not independent quantities, as demonstrated by (1.3.0.4). Minimizing total variance is easier if the equation is written in terms of independent quantities, so substituting in from (1.3.0.4), we get:

$$\overline{\delta^2(S_1+S_2)} = \frac{c_A^2}{N_A}\left(1-P_A\right)P_A + \frac{c_B^2}{N_B}\left(1-P_B\right)P_B \tag{4.4.0.23}$$

While N_A and N_B can be varied independently, to get meaningful minimization results $N = N_A + N_B$ should be kept constant. The method of Lagrange multipliers can be used to incorporate all of the necessary constraints in the minimization problem:

$$\vec{0} = \nabla|_{N_A=N'_A, N_B=N'_B}\left(\frac{c_A^2}{N_A}\left(1-P_A\right)P_A + \frac{c_B^2}{N_B}\left(1-P_B\right)P_B - \lambda\left(N-\left(N_A+N_B\right)\right)\right) \tag{4.4.0.24}$$

Variables of the gradient are N_A, N_B, λ, where λ is the Lagrange multiplier itself and N'_A, N'_B is the place of the minimum. Evaluating the derivatives yields:

$$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \lambda - \frac{c_A^2}{N_A'^2}P_A(1-P_A) \\ \lambda - \frac{c_B^2}{N_B'^2}P_B(1-P_B) \\ N - (N'_A + N'_B) \end{pmatrix} \tag{4.4.0.25}$$

The above equations can be solved to find N'_A, which minimizes total variance:

$$N'_A = N\frac{1}{1+\frac{c_B}{c_A}\sqrt{\frac{P_B(1-P_B)}{P_A(1-P_A)}}} \tag{4.4.0.26}$$

However, it might be more convenient, to simply rearrange the equations into a symmetrical form:

$$\frac{N'_A}{c_A\sqrt{P_A(1-P_A)}} = \frac{N'_B}{c_B\sqrt{P_B(1-P_B)}} \tag{4.4.0.27}$$

and keep the constraint $N = N'_A + N'_B$ in mind. This solution minimizes variance by defining a simple and symmetrical weighting for the number of virtual particles originating from individual voxels.

Chapter 5

Optimizing Forward Projection in a General System

5.1 Total Variance

Real imaging systems have more than two LORs and voxels. Let us consider a system of n voxels: $V_1, V_2, \ldots, V_n$ and l LORs: $L_1, L_2, \ldots, L_l$, whose measured signals are $S_1, S_2, \ldots, S_l$. Using notation similar to that applied in the 2×2 system, the total number of decays in the individual voxels are: $c_1, c_2, \ldots, c_n$. The number of virtual photon pairs started from them are: $N_1, N_2, \ldots, N_n$ and their weights are $w_1, w_2, \ldots, w_n$. Analogous to (1.3.0.4) these are related by:

$$c_i = N_i w_i \forall i \in [1, \ldots, n] \tag{5.1.0.1}$$

Elements of the system matrix $\boldsymbol{A}$ are defined as follows: A_{ij} is the probability that a particle from voxel V_j gets picked up by LOR L_i.

The total measured LOR count can once again be written as:

$$S_T = \sum_{i=1}^{l} S_i \tag{5.1.0.2}$$

And defining the total signal contributions from the individual voxels as S_{V_i} (analogous to S_A and S_B in (4.3.0.15)), the above can be expressed as:

$$S_T = \sum_{i=1}^{l} S_i = \sum_{i=1}^{n} S_{V_i} \tag{5.1.0.3}$$

we know that S_{V_i} are independent and also that they are binomially distributed as in (4.3.0.17):

$$S_{V_i} \sim \mathcal{S}_{V_i} = w_i \mathcal{B}(N_i, P_i) \tag{5.1.0.4}$$

where P_i is the probability that a virtual particle starting from V_i get detected by one of the LORs. This means that the variances of voxel contributions are given by:

$$\mathbb{D}^2[S_{V_i}] = w_i^2 N_i P_i (1 - P_i) \tag{5.1.0.5}$$

From the definition of the system matrix P_i values can be calculated as:

$$P_i = \sum_{j=1}^{l} A_{ji} \tag{5.1.0.6}$$

using this and independence, we can already write down the total variance:

$$\overline{\delta^2 S_T} = \sum_{i=1}^{n} \mathbb{D}^2[\mathcal{S}_{V_i}] = \sum_{i=1}^{n} w_i^2 N_i P_i (1 - P_i) \tag{5.1.0.7}$$

5.2 Minimum Variance

We know that total variance of LOR counts in a general system is given by:

$$\overline{\delta^2 S_T} = \sum_{i=1}^{n} \mathbb{D}^2[\mathcal{S}_{V_i}] = \sum_{i=1}^{n} w_i^2 N_i P_i (1 - P_i) \tag{5.2.0.8}$$

If we substitute in $w_i = \frac{c_i}{N_i}$, we get:

$$\overline{\delta^2 S_T} = \sum_{i=1}^{n} \frac{c_i^2}{N_i} P_i (1 - P_i) \tag{5.2.0.9}$$

this is the expression we aim to minimize. As before, the total particle count $N = \sum_{i=1}^{n} N_i$ should be kept constant. Using Lagrange multipliers, the minimization problem can be formulated as:

$$\vec{0} = \nabla|_{N_i = N_i'} \left(-\lambda \left(N - \sum_{i=1}^{n} N_i \right) + \sum_{i=1}^{n} \frac{c_i^2}{N_i} (1 - P_i) P_i \right) \tag{5.2.0.10}$$

Evaluating the expression at the individual indices gives:

$$0 = \lambda - \frac{c_i^2}{N_i'^2}\left(1 - P_i\right) P_i \tag{5.2.0.11}$$

substituting in from the equation for index i into the one for index j, we get:

$$\tfrac{c_i^2}{N_i'^2}\left(1 - P_i\right) P_i = \tfrac{c_j^2}{N_j'^2}\left(1 - P_j\right) P_j \quad \forall i, j \in [1, \ldots, n] \tag{5.2.0.12}$$

rearranging the above equation leads to:

$$\tfrac{N_i'}{c_i\sqrt{(1-P_i)P_i}} = \tfrac{N_j'}{c_j\sqrt{(1-P_j)P_j}} \quad \forall i, j \in [1, \ldots, n] \tag{5.2.0.13}$$

which means that the distribution of virtual particle pairs which minimizes total variance in the forward projection is given by:

$$N_i' = N \frac{c_i\sqrt{P_i(1-P_i)}}{\sum\limits_{j=1}^{n} c_j\sqrt{P_j(1-P_j)}} \tag{5.2.0.14}$$

Chapter 6

Forward Projection Simulation

A simulation was written to test the theoretical results. It uses a setup identical to the one introduced in section 1.2. It consists of two voxels and two LORs. The voxels emit a given number of virtual particles that have weights defined in (1.3.0.4). Subsequent fates of said particles are determined by random samples drawn from a multinomial distribution characterized by the system matrix.

Simulation results were compared to calculated variance values. Both cases, neglected and included LOR correlations, were tested. Simulation results and comparisons to theoretical values are shown on figures 6.1 and 6.2.

6.1 Simulation Results

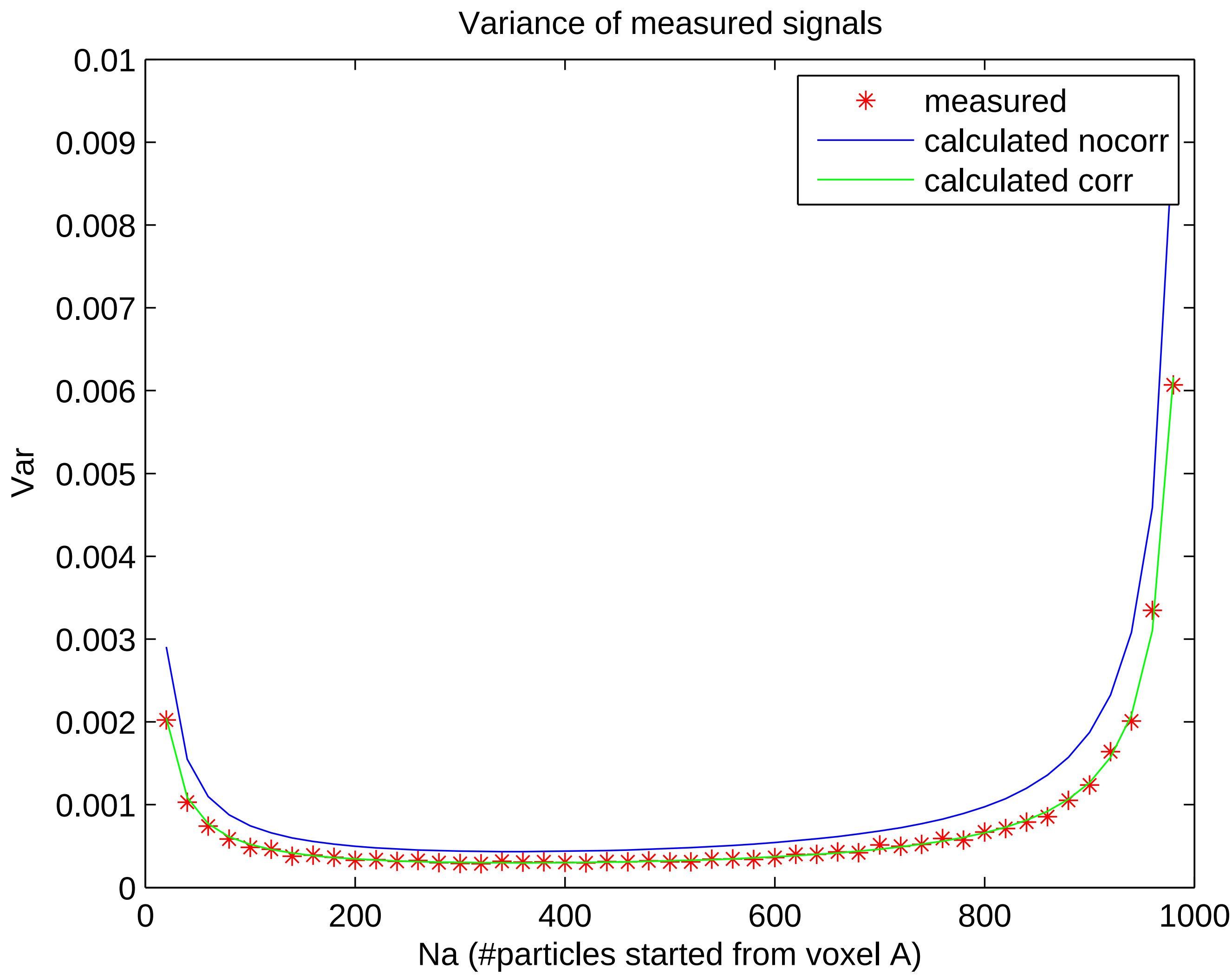

Figure 6.1: Linear plot of variance results. It is clear that simulated and calculated results accounting for correlations are in agreement. Neglecting correlations causes a slight deviation from simulated values.

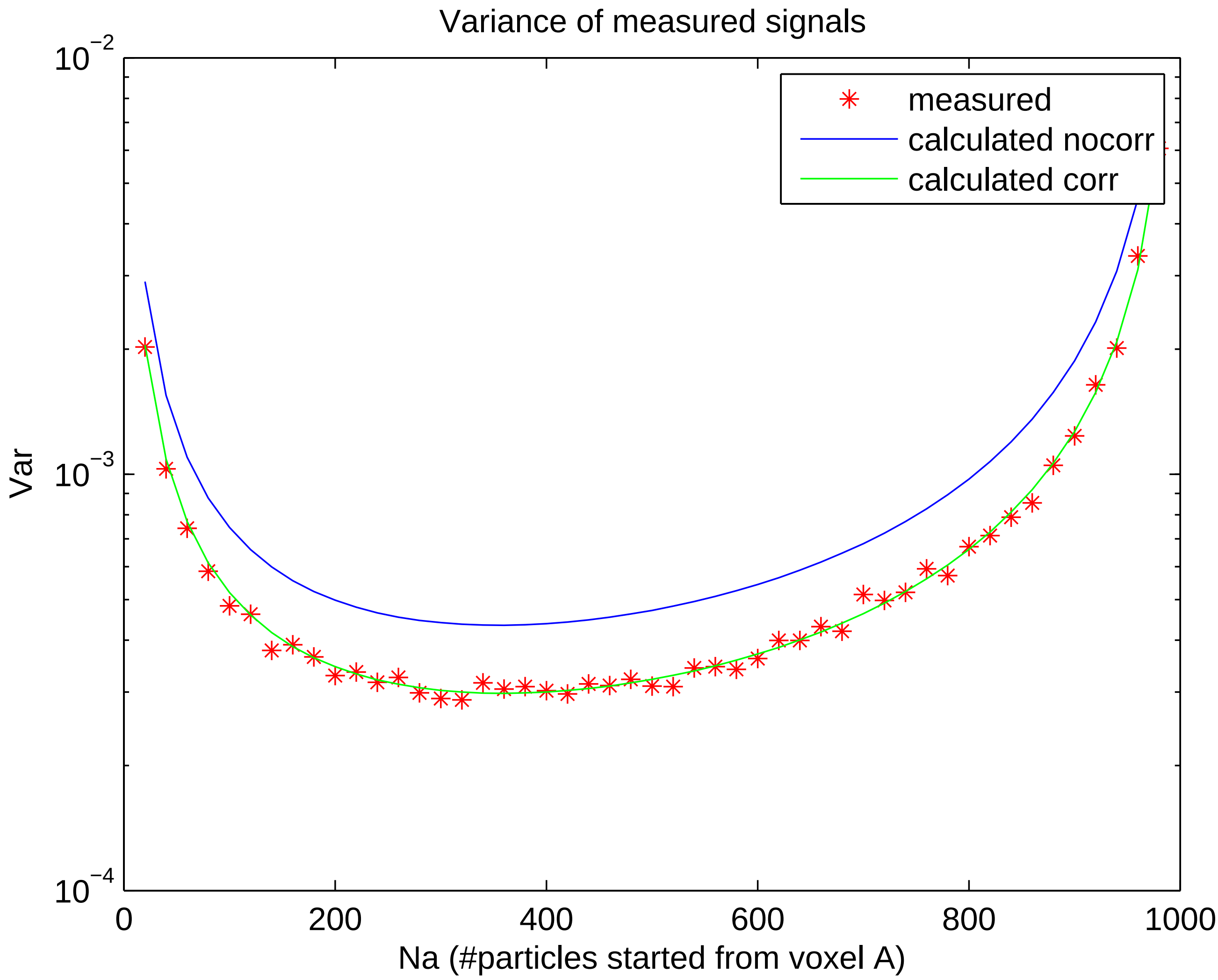

Figure 6.2: Log-linear plot of variance results. It is clear that simulated and calculated results accounting for correlations are in agreement. Neglecting correlations causes a slight deviation from simulated values.

6.2 Effect on Measured Signal

While the total variance of the forward projection step is an important measure, it is not what actually affects the quality of reconstruction. The quantity that has to carry low variance is $\frac{y_i^m}{y_i^s}$, the ratio of measured and simulated LOR counts. Several simulations were run to determine the effect of different samplings on this quantity. The measured values were replaced with true EVs (this doesn't affect the behaviour of different samplings). Results are shown in figure 6.3 and 6.4. The conclusion is that neither sampling is obviously closer to the true minimum than the others. Depending on system specifics, relative performance of the three tested samplings (uniform, activity weighted and theoretically optimal) vary greatly. To properly compare their performance, a whole iterative reconstruction is needed, such a comparison can be found in chapter 8. The fact that optimal total forward projection variance does not immediately translate to better reconstruction is in agreement with [1]. According to their findings, relative variance of individual LOR counts is more relevant to reconstruction quality.

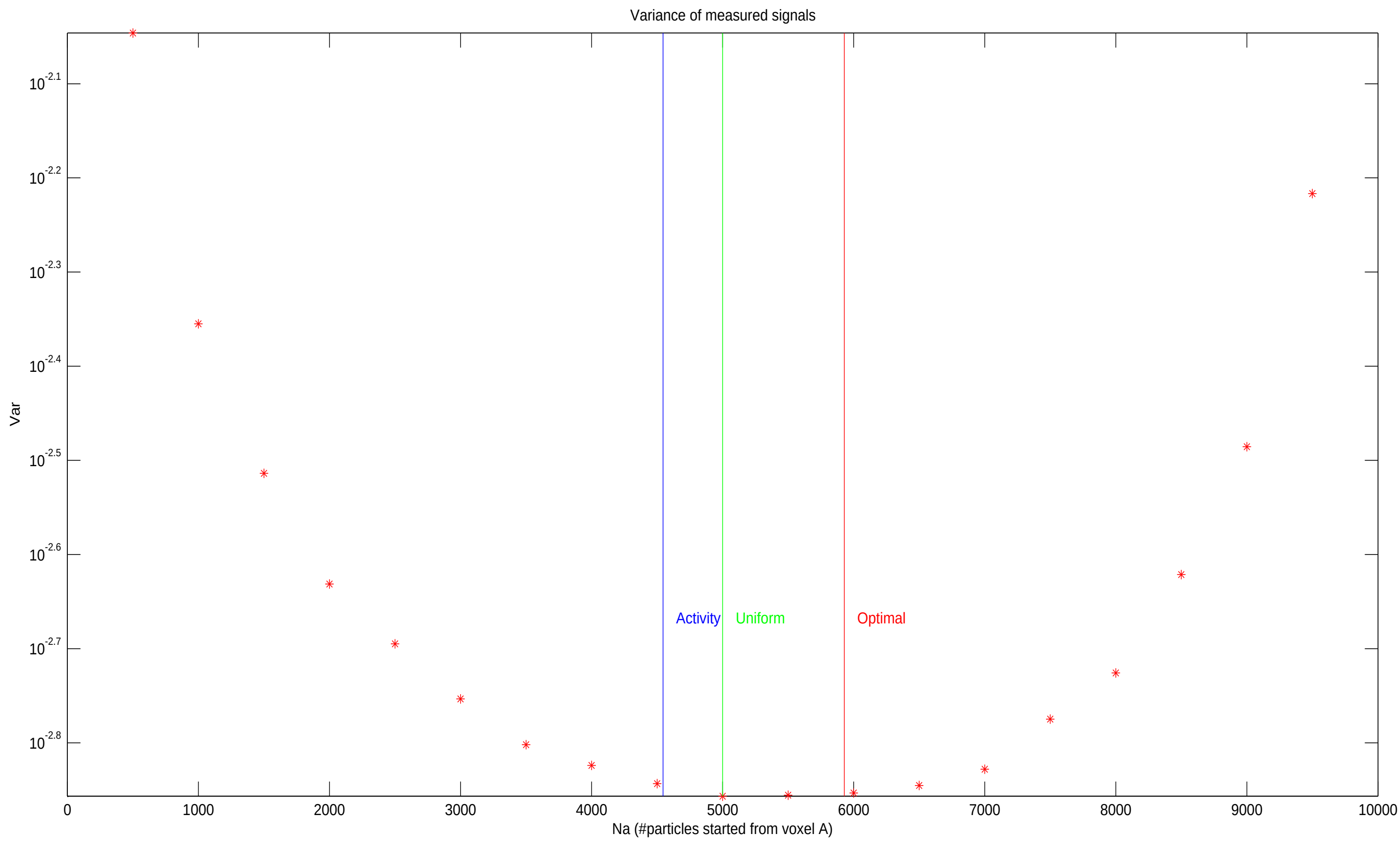

Figure 6.3: Total variance of the ratio of measured and simulated LOR counts. Red markers show simulation results, while vertical lines show what portion of all virtual particles the different samplings appoint to voxel A. System matrix: $[0.1, 0.04; 0.4, 0.05]$, voxel activities: $[1, 12]$.

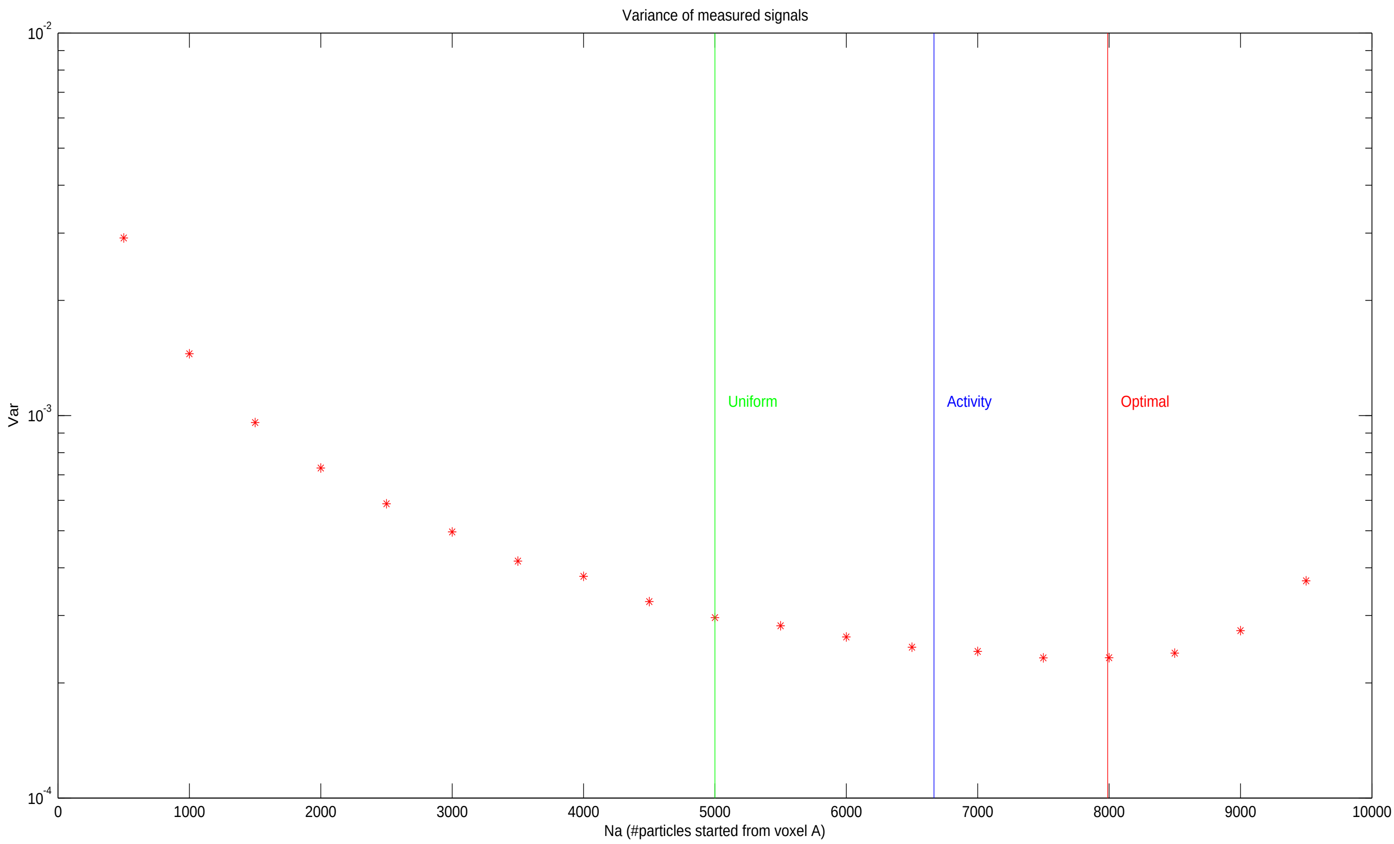

Figure 6.4: Total variance of the ratio of measured and simulated LOR counts. Red markers show simulation results, while vertical lines show what portion of all virtual particles the different samplings appoint to voxel A. System matrix: $[0.1, 0.45; 0.15, 0.5]$, voxel activities: $[1, 12]$.

Chapter 7

Optimizing Backward Projection

After discussing the forward projection step, it is worthwhile to analyze backward projection as well.

7.1 Introduction

To choose an approach for optimization, it is important to understand exactly how the system works. The back projection step uses the formula (3.1.0.2):

$$x_j^{k+1} = x_j^k \sum_{i=1}^{l} \frac{A_{ij}}{\sum_{s=1}^{l} A_{sj}} \frac{y_i^m}{\sum_{t=1}^{n} A_{it} x_t^k} = x_j^k \sum_{i=1}^{l} w_{ij} z_i \tag{7.1.0.1}$$

where w_{ij} are the reduced system matrix elements and z_i are the ratios of measured and simulated LOR counts. While the system matrix elements are not known (the matrix is too large to store in memory), they can be simulated during reconstruction to evaluate the above expression. The way this is done is that a particle transport simulation is run which chooses LOR i with probability w_{ij}. The indicator variable ζ^j is the index of the particular LOR that picked up the photon pair. To properly approximate the above sum, multiple particle transport simulations are needed. If K_j is the sample size, then

$$\zeta_m^j, 1 \leq m \leq K_j \tag{7.1.0.2}$$

are the indicator variables, which are multinomially distributed with probabilities w_{ij} and sample size K_j. This sampling is used in [7] with activity weighted K_j values.

Now, the Monte Carlo estimate for the reconstructed voxel count can be written as:

$$\hat{x_j}^{k+1} = \hat{x_j}^k \frac{1}{K_j} \sum_{m=1}^{K_j} z_{\zeta_m^j} \tag{7.1.0.3}$$

To proceed further, information about the distribution of z_i is needed. Like before, we can assume that it is a normal random variable with mean μ_i and variance σ_i^2, or $z_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$. If we denote the probability density function by $f_i(x)$, then the PDF of $z_{\zeta_m^j}$ is given by:

$$f_{\zeta_m^j}(x) = \sum_{i=1}^{l} w_{ij} f_i(x) \tag{7.1.0.4}$$

(the individual PDF-s have to be weighted by the probability that they are chosen in any one trial). The expected value can then be expressed as:

$$M_j = \mathbb{E}\left[f_{\zeta_m^j}(x)\right] = \sum_{i=1}^{l} w_{ij} \mathbb{E}\left[f_i(x)\right] = \sum_{i=1}^{l} w_{ij} \mu_i \tag{7.1.0.5}$$

The above shows that the EV of the above MC procedure is exactly the deterministic result (3.1.0.2), therefore it is an unbiased estimator for the iterative reconstruction step.

Determining the variance of $z_{\zeta_m^j}$ is also possible.

$$d_j^2 = \mathbb{D}^2\left[f_{\zeta_m^j}(x)\right] = \int_{-\infty}^{\infty} (x - M_j)^2 f_{\zeta_m^j}(x) dx = \sum_{i=1}^{l} w_{ij} \int_{-\infty}^{\infty} (x - M_j)^2 f_i(x) dx \tag{7.1.0.6}$$

The second non-central moment of a distribution can be calculated analytically (see appendix A.3, and (A.3.0.22) in particular). So the problematic term on the left becomes:

$$\int_{-\infty}^{\infty} (x - M_j)^2 f_i(x) dx = \sigma_i^2 + (\mu_i - M_j)^2 \tag{7.1.0.7}$$

and so the variance reads:

$$d_j^2 = \sum_{i=1}^{l} w_{ij} \left(\sigma_i^2 + (\mu_i - M_j)^2\right) \tag{7.1.0.8}$$

Since the reconstructed activity is calculated from the average of K_j independent identically distributed (IID) random variables, the variance is reduced by a factor of K_j, so the variance of the reconstructed activity can be written as:

$$D_j^2 = \left(x_j^k\right)^2 \frac{1}{K_j} \sum_{i=1}^{l} w_{ij} \left(\sigma_i^2 + (\mu_i - M_j)^2\right) \tag{7.1.0.9}$$

7.2 Optimal Biased Probabilities

A possible method for reducing variance, while keeping the EV constant is to introduce a bias in the selection probabilities. Instead of using the natural w_{ij} values, we can opt to use a selection method that has probabilities P_{ij}. These still have to satisfy the normalization:

$$\sum_{i=1}^{l} P_{ij} = 1 \tag{7.2.0.10}$$

otherwise they would not define proper multinomial distributions. Furthermore, the change in probabilities has to be compensated for by introducing weight factors that multiply the z_i values. Denoting these weights by q_{ij} and the new indicators by ξ_m^j, the new EV is is given by:

$$M_j' = \mathbb{E}\left[f_{\xi_m^j}(x)\right] = \sum_{i=1}^{l} P_{ij} q_{ij} \mathbb{E}\left[f_i(x)\right] = \sum_{i=1}^{l} P_{ij} q_{ij} \mu_i \tag{7.2.0.11}$$

to get the same EV as (7.1.0.5), we must have $P_{ij} q_{ij} = w_{ij}$, so the weights are constrained to be:

$$q_{ij} = \frac{w_{ij}}{P_{ij}} \tag{7.2.0.12}$$

This means that the new scheme for the back projection step is:

$$\hat{x_j}^{k+1} = \hat{x_j}^{k} \frac{1}{K_j} \sum_{m=1}^{K_j} u^j_{\xi_m^j} \tag{7.2.0.13}$$

where $u_i^j = \frac{w_{ij}}{P_{ij}} z_i$. This means that the normally distributed variables have different means and variances:

$$u_i^j \sim \mathcal{N}\left(\frac{w_{ij}}{P_{ij}} \mu_i, \left(\frac{w_{ij}}{P_{ij}}\right)^2 \sigma_i^2\right) \tag{7.2.0.14}$$

The new variance of a single MC run can be calculated analogously to the previous calculation, by noting that the new probabilities are given by P_{ij}, the new variances are $\left(\frac{w_{ij}}{P_{ij}}\right)^2 \sigma_i^2$ and the new means are $\frac{w_{ij}}{P_{ij}} \mu_i$, while the overall mean M_j is unchanged. This yields:

$$d_j'^2 = \sum_{i=1}^{l} P_{ij} \left(\left(\frac{w_{ij}}{P_{ij}}\right)^2 \sigma_i^2 + \left(\frac{w_{ij}}{P_{ij}} \mu_i - M_j\right)^2\right) \tag{7.2.0.15}$$

And so the total reconstructed variance becomes (after averaging K_j runs):

$$D_j'^2 = \left(x_j^k\right)^2 \frac{1}{K_j} \sum_{i=1}^{l} P_{ij} \left(\left(\frac{w_{ij}}{P_{ij}}\right)^2 \sigma_i^2 + \left(\frac{w_{ij}}{P_{ij}}\mu_i - M_j\right)^2 \right) \tag{7.2.0.16}$$

Now we can determine the optimal P_{ij} values by incorporating the normalization constraint into a minimization problem using Lagrange multipliers:

$$G_j = \left(x_j^k\right)^2 \frac{1}{K_j} \sum_{i=1}^{l} P_{ij} \left(\left(\frac{w_{ij}}{P_{ij}}\right)^2 \sigma_i^2 + \left(\frac{w_{ij}}{P_{ij}}\mu_i - M_j\right)^2 \right) - \lambda_j \left(1 - \sum_{i=1}^{l} P_{ij}\right) \tag{7.2.0.17}$$

The partial derivative with respect to λ_j returns the normalization condition, while derivatives with respect to the probabilities yield:

$$\partial_{P_{ij}} G_j = 0 = \left(x_j^k\right)^2 \frac{1}{K_j} \left(-\frac{w_{ij}^2 \sigma_i^2}{P_{ij}^2} - \frac{w_{ij}^2 \mu_i^2}{P_{ij}^2} + M_j^2 \right) + \lambda_j \tag{7.2.0.18}$$

so, for the optimal probabilities P_{ij}':

$$\frac{K_j}{\left(x_j^k\right)^2} \lambda_j + M_j = \frac{w_{ij}^2 \left(\sigma_i^2 + \mu_i^2\right)}{P_{ij}'^2} \tag{7.2.0.19}$$

The left had side is a constant. Denoting it by c_j^2, we get:

$$P_{ij}' = \frac{1}{c_j} w_{ij} \sqrt{\sigma_i^2 + \mu_i^2} \tag{7.2.0.20}$$

or, using the normalization condition, we can write:

$$P_{ij}' = \frac{w_{ij}\sqrt{\sigma_i^2 + \mu_i^2}}{\sum_{r=1}^{l} w_{rj}\sqrt{\sigma_r^2 + \mu_r^2}} \tag{7.2.0.21}$$

using the notation $\gamma_j = \sum_{r=1}^{l} w_{rj}\sqrt{\sigma_r^2 + \mu_r^2}$ and substituting back into (7.2.0.16), we can calculate the minimum variance using K_j MC runs. After simplifying and re-arranging the equation, we arrive at:

$$D_j''^2 = \left(x_j^k\right)^2 \frac{1}{K_j} \left(\gamma_j^2 - M_j^2\right) = \left(x_j^k\right)^2 \frac{1}{K_j} \left(\left(\sum_{r=1}^{l} w_{rj}\sqrt{\sigma_r^2 + \mu_r^2} \right)^2 - \left(\sum_{s=1}^{l} w_{sj}\mu_s \right)^2 \right) \tag{7.2.0.22}$$

Interestingly the variance using natural probabilities w_{ij} is given by a similar expression:

$$D_j^2 = \left(x_j^k\right)^2 \frac{1}{K_j} \left(\left(\sum_{i=1}^{l} w_{ij} \left(\sigma_i^2 + \mu_i^2\right) \right) - \left(\sum_{s=1}^{l} w_{sj} \mu_s \right)^2 \right) \tag{7.2.0.23}$$

7.3 Optimal Distribution of Monte Carlo Samples

Our primary aim is to optimize variance of the whole simulation, not just the iteration steps for the individual voxels. This means finding the ideal distribution of K MC runs between the individual voxel reconstructions that use K_j respective Monte Carlo samples. Therefore the constraint is:

$$\sum_{j=1}^{n} K_j = K \tag{7.3.0.24}$$

And the total variance assuming independent reconstructed voxel activities:

$$D_t^2 = \sum_{r=1}^{n} D_j''^2 = \sum_{j=1}^{n} \left(x_j^k\right)^2 \frac{1}{K_j} \left(\gamma_j^2 - M_j^2\right) \tag{7.3.0.25}$$

Combining the two into an unconstrained minimization problem utilizing Lagrange multipliers:

$$G_t = \sum_{j=1}^{n} \left(x_j^k\right)^2 \frac{1}{K_j} \left(\gamma_j^2 - M_j^2\right) - \lambda \left(K - \sum_{j=1}^{n} K_j \right) \tag{7.3.0.26}$$

Partial derivative with respect to λ returns the constraint, while the partial derivatives WRT K_j give:

$$\partial_{K_j} G_t = 0 = \lambda - \frac{\left(x_j^k\right)^2 \left(\gamma_j^2 - M_j^2\right)}{K_j'^2} \tag{7.3.0.27}$$

so for optimal K_j' sample sizes:

$$K_j' = \frac{x_j^k \sqrt{\gamma_j^2 - M_j^2}}{\sqrt{\lambda}} \tag{7.3.0.28}$$

finally, using the constraint, we get:

$$K_j' = \frac{x_j^k \sqrt{\gamma_j^2 - M_j^2}}{\sum_{r=1}^{n} x_r^k \sqrt{\gamma_r^2 - M_r^2}} \tag{7.3.0.29}$$

When individual voxel statistics do not differ significantly then the expression $\gamma_r^2 - M_r^2$ can be assumed to be independent of index r. In this case the above expression simplifies to:

$$K'_j = \frac{x_j^k}{\sum\limits_{r=1}^{n} x_r^k} \tag{7.3.0.30}$$

which is the widely used activity scaled sampling. However, when voxels exhibit varied behaviour, the above simplification could lead to sub-optimal sampling, and thus increased reconstructed variance.

Chapter 8

Reconstruction Results and Conclusion

The three different back projection samplings were compared by reconstructing an image using the same uniform forward projection sampling for all three. Results after averaging 10000 different simulations are shown in figure 8.1. The sampling that minimizes back projection variance does in fact result in slightly lower L_2 error in reconstruction.

The three different forward projection samplings were also compared. A reconstruction was carried out with each forward projection sampling, using the same uniform back projection sampling for all of them. Results after averaging 10000 different simulations are shown in figure 8.2. It is clear that minimizing total forward projection variance does not necessarily result in better reconstruction. Activity weighted forward projection sampling results in lower L_2 error than "optimal" sampling.

These conclusions are in agreement with [1]. Their findings indicate that minimizing total variance in the back projection step is a reasonable optimization, but relative LOR variances should be minimized during forward projection. Based on these results, further study into a sampling minimizing relative LOR count variances in the forward projection step is warranted.

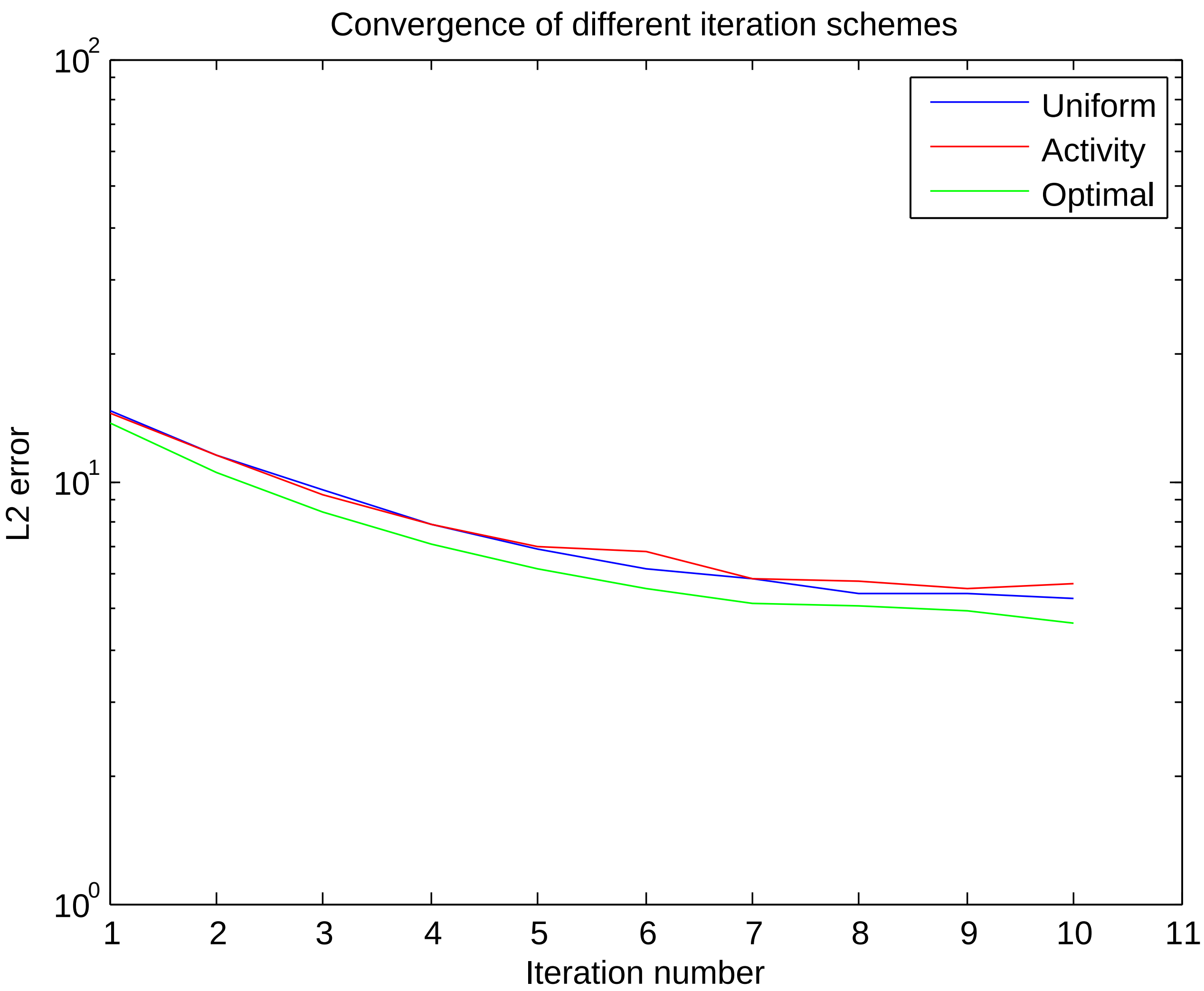

Figure 8.1: The log-linear plot shows total squared error of reconstructed activity values as a function of iteration number. It can be seen that choosing between uniform and activity weighted sampling in the back projection step has little to no effect, while the optimal sampling performs slightly better than the rest. System matrix: $[0.3, 0.1; 0.4, 0.8]$, activity concentrations: $[1, 9]$

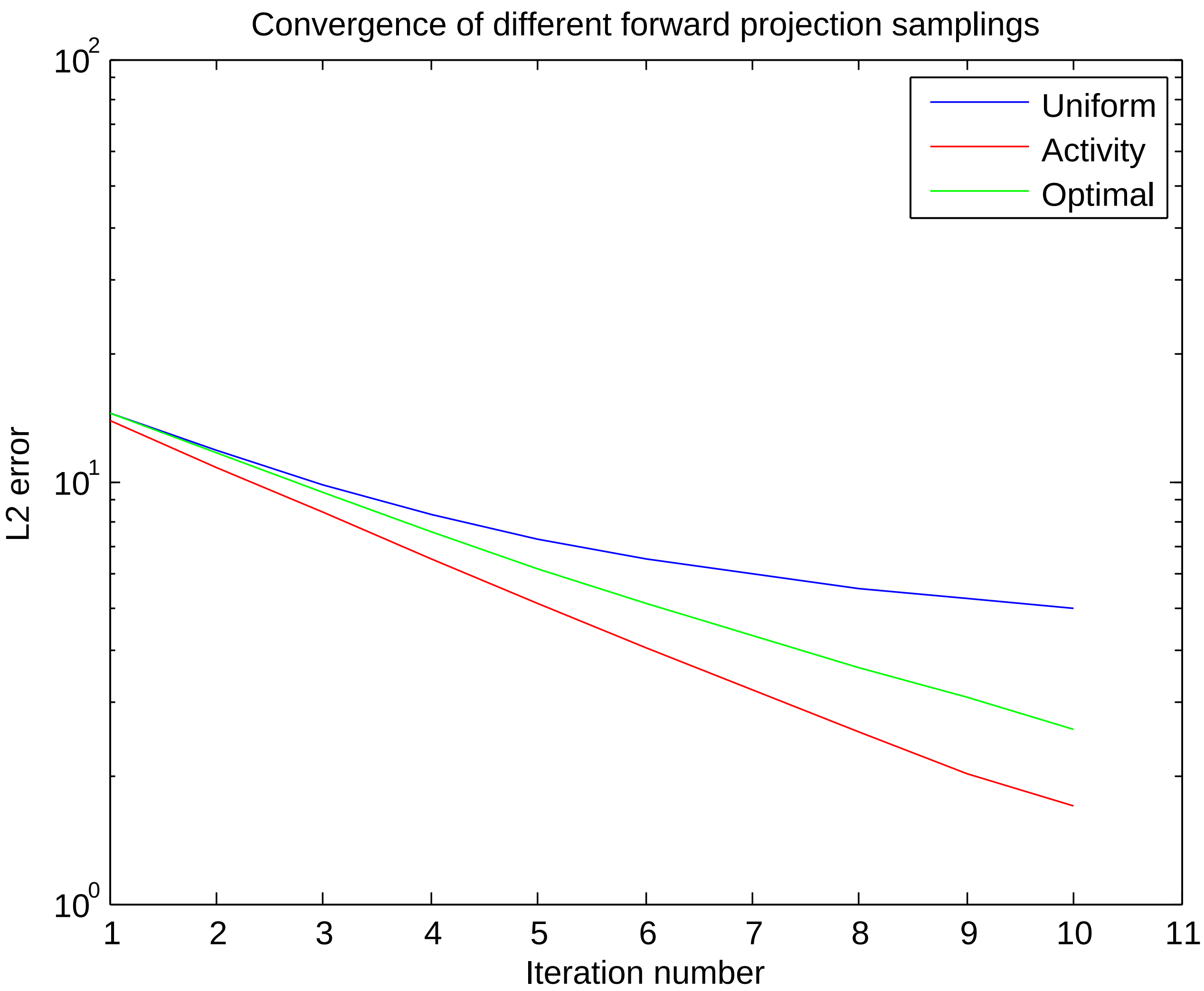

Figure 8.2: The log-linear plot shows total squared error of reconstructed activity values as a function of iteration number. It can be seen that uniform is significantly worse than either of the others. Furthermore activity weighted sampling performs better than the theoretically derived optimal sampling. System matrix: $[0.3, 0.1; 0.4, 0.8]$, activity concentrations: $[1, 9]$

Appendices

Appendix A

Calculations

A.1 Monte Carlo Variances

After defining the model, it is possible to calculate Monte Carlo variances for it. To do this, let us consider once again S_1, the signal measured by L_1. Each individual virtual particle can be assigned a detected weight q_i, which describes how much of that particle is picked up by L_1. In our simple case, this can only be 0 or the total weight of the particle.

$$S_1 = \sum_{i=1}^{N} q_i \tag{A.1.0.1}$$

Now, by the definition of variance we can write:

$$\delta^2 S_1 = \sum_{i=1}^{N} (q_i - \overline{q})^2 \tag{A.1.0.2}$$

where $\overline{q}$ is the average weight of detected particles, which is by definition

$$\overline{q} = \frac{1}{N} \sum_{i=1}^{N} q_i = \frac{1}{N} S_1 \tag{A.1.0.3}$$

substituting this into (A.1.0.2) yields:

$$\delta^2 S_1 = \sum_{i=1}^{N} \left(q_i - \frac{S_1}{N} \right)^2 = \sum_{i=1}^{N} q_i^2 - 2\frac{S_1}{N} \sum_{i=1}^{N} q_i + \sum_{i=1}^{N} \left(\frac{S_1}{N} \right)^2 \tag{A.1.0.4}$$

once again making the substitution (A.1.0.1):

$$\delta^2 S_1 = \sum_{i=1}^{N} q_i^2 - \frac{S_1^2}{N} = S_1^2 \left(\frac{\sum_{i=1}^{N} q_i^2}{\left(\sum_{i=1}^{N} q_i\right)^2} - \frac{1}{N} \right) \tag{A.1.0.5}$$

which is the traditional form of Monte-Carlo variance

Signals detected by different LORs are not independent, since the number of particles going from a specific voxel to different LORs are correlated. This means that there are non-zero covariances between different LOR counts. S_2 can also be defined in a manner analogous to (A.1.0.1):

$$S_2 = \sum_{i=1}^{N} p_i = N\overline{p} \tag{A.1.0.6}$$

where p_i are the detection weights assigned to L_2. Now, from the definition of covariance we can write:

$$\delta^2 S_1 S_2 = \sum_{i=1}^{N} (q_i - \overline{q})(p_i - \overline{p}) \tag{A.1.0.7}$$

Now substituting in from (A.1.0.1) and (A.1.0.6), we get:

$$\delta^2 S_1 S_2 = \sum_{i=1}^{N} q_i p_i - \frac{S_1 S_2}{N} \tag{A.1.0.8}$$

in this model no particle can be detected by more than one LOR, p_i and q_i can not be non-zero simultaneously, so the first term is zero. This means that:

$$\delta^2 S_1 S_2 = -\frac{S_1 S_2}{N} \tag{A.1.0.9}$$

A.2 Hypergeometric Sums

(2.2.0.4) can be re-arranged to give:

$$y_m^i \sum_{k=1}^{n} \frac{1}{k} \binom{n}{k} p^k (1-p)^{n-k} = y_i^m np (1-p)^{n-1} \sum_{k=1}^{n} \frac{1}{k} \frac{(n-1)!}{(n-k)!k!} \left(\frac{p}{1-p}\right)^{k-1} \tag{A.2.0.10}$$

To show that the sum on the right is equal to ${}_pF_q\left(\{1,1,1-n\}\,;\{2,2\}\,;\frac{p}{p-1}\right)$, we can start from the definition of the generalized hypergeometric function as in [8]:

$${}_pF_q\left(\{a_1,\ldots,a_q\}\,;\{b_1,\ldots,b_r\}\,;z\right) = \sum_{l=0}^{\infty} \frac{\prod_{i=1}^{q} (a_i)_l}{\prod_{j=1}^{r} (b_j)_l} \frac{z^l}{l!} \tag{A.2.0.11}$$

Where $(a)_l$ is the Pochhammer symbol (rising factorial notation) defined by:

$$(a)_l = \begin{cases} 1 & \text{if } l = 0 \\ \prod_{i=0}^{l-1} a+i & \text{if } l \geq 1 \end{cases} \tag{A.2.0.12}$$

Note that from the above, $(1)_l = l!$ and $(2)_l = (l+1)!$

From the above definitions, we can expand the hypergeometric term:

$$\begin{gathered} {}_pF_q\left(\{1,1,1-n\}\,;\{2,2\}\,;\frac{p}{p-1}\right) = \sum_{k=0}^{\infty} \frac{(1)_k (1)_k (1-n)_k}{(2)_k (2)_k} \frac{1}{k!} \left(\frac{p}{p-1}\right)^k = \\ \sum_{k=0}^{\infty} \frac{k!k!(1-n)_k}{(k+1)!(k+1)!} \frac{1}{k!} \left(\frac{p}{p-1}\right)^k = \sum_{k=0}^{\infty} \frac{(1-n)_k}{(k+1)(k+1)!} \left(\frac{p}{p-1}\right)^k \end{gathered} \tag{A.2.0.13}$$

The above expression has several important properties. For $k \geq n \geq 1$, $(1-n)_k = \prod_{j=1}^{k} (j-n) = 0$, therefore the sum only has to run up to $n-1$ instead of ∞. Furthermore, for all of these value, the terms $(j-n)$ are negative as well as the $p-1$ terms; a $(-1)^k$ multiplier for both expression will make them positive. Making these re-arrangements and introducing the index $i = k+1$ we get:

$$\sum_{k=0}^{\infty} \frac{\prod_{j=1}^{k} (j-n)}{(k+1)(k+1)!} \left(\frac{p}{p-1}\right)^k = \sum_{i=1}^{n} \frac{(-1)^{i-1} (1-n)_{i-1}}{i \cdot i!} (-1)^{i-1} \left(\frac{p}{p-1}\right)^{i-1} \tag{A.2.0.14}$$

In the above equation $(-1)^{i-1}(1-n)_{i-1} = \frac{(n-1)!}{(n-i)!}$ (this is only true for indices $1 \leq i \leq n$, which are exactly the indices that appear in the sum). Finally by noting that $(-1)^{i-1}\left(\frac{p}{p-1}\right)^{i-1} = \left(\frac{p}{1-p}\right)^{i-1}$, we can write:

$$\sum_{i=1}^{n} \frac{(-1)^{i-1}(1-n)_{i-1}}{i \cdot i!}(-1)^{i-1}\left(\frac{p}{p-1}\right)^{i-1} = \sum_{i=1}^{n} \frac{1}{i}\frac{(n-1)!}{(n-i)!i!}\left(\frac{p}{1-p}\right)^{i-1} = {}_pF_q\left(\{1,1,1-n\};\{2,2\};\frac{p}{p-1}\right) \tag{A.2.0.15}$$

Comparing this to (A.2.0.10) proves the initial proposition that

$$y_m^i \sum_{k=1}^{n} \frac{1}{k}\binom{n}{k} p^k (1-p)^{n-k} = y_i^m np(1-p)^{n-1} {}_pF_q\left(\{1,1,1-n\};\{2,2\};\frac{p}{p-1}\right) \tag{A.2.0.16}$$

From the above derivation it is clear that extending the arguments to ${}_pF_q\left(\{1,1,1,1-n\};\{2,2,2\};\frac{p}{p-1}\right)$ simply multiplies every term in the sum by $\frac{1}{k}$, and so (2.2.0.5) gives:

$$\left(y_m^i\right)^2 \sum_{k=1}^{n} \left(\frac{1}{k}\right)^2 \binom{n}{k} p^k (1-p)^{n-k} = \left(y_i^m\right)^2 np(1-p)^{n-1} \sum_{k=1}^{n} \left(\frac{1}{k}\right)^2 \frac{(n-1)!}{(n-k)!k!}\left(\frac{p}{1-p}\right)^{k-1} = \left(y_i^m\right)^2 np(1-p)^{n-1} {}_pF_q\left(\{1,1,1,1-n\};\{2,2,2\};\frac{p}{p-1}\right) \tag{A.2.0.17}$$

A.3 Probability Density Functions

Given an arbitrary probability density function $f(x)$, by definition we have the normalization property:

$$\int\limits_{-\infty}^{\infty} f(x)dx = 1 \tag{A.3.0.18}$$

The definition of the mean:

$$\int\limits_{-\infty}^{\infty} xf(x)dx = \mu \tag{A.3.0.19}$$

and variance:

$$\int\limits_{-\infty}^{\infty} (x-\mu)^2 f(x)dx = \sigma^2 \tag{A.3.0.20}$$

multiplying (A.3.0.21) by μ and subtracting from (A.3.0.19), we get:

$$\int\limits_{-\infty}^{\infty} (x-\mu) f(x)dx = 0 \tag{A.3.0.21}$$

The above equations can be used to derive the value of the non-central second moment around an arbitrary value a:

$$\begin{gathered}\int\limits_{-\infty}^{\infty} (x-a)^2 f(x)dx = \int\limits_{-\infty}^{\infty} ((x-\mu)+(\mu-a))^2 f(x)dx = \\ \sigma^2 + (\mu-a)^2 + 2(\mu-a)\int\limits_{-\infty}^{\infty} (x-\mu) f(x)dx = \sigma^2 + (\mu-a)^2\end{gathered} \tag{A.3.0.22}$$

Bibliography

[1] L. Szirmay-Kalos and Milan Magdics and Balazs Toth and Tamas Buki, Instant 3D Reconstruction for PET on the GPU.

[2] A. Ore and J. L. Powell, Three-Photon Annihilation of and Electron-Positron Pair. Phys. Rev., 1949, DOI 10.1103/PhysRev.75.1696.

[3] I. Lux and Laszlo Koblinger, Monte Carlo particle transport methods: neutron and photon calculations. CRC Press, 1991, ISBN 9780849360749.

[4] Shepp, L.A. and Vardi, Y., Maximum Likelihood Reconstruction for Emission Tomography. IEEE, 1982, ISSN 0278-0062.

[5] W. RICHARDSON, Bayesian-Based Iterative Method of Image Restoration. Optical Society of America, 1972, JOSA.62.000055.

[6] Van Heertum, R.L. and Tikofsky, R.S. and Ichise, M., Functional Cerebral SPECT and PET Imaging. Wolters Kluwer Health, 2013, ISBN 9781451153392.

[7] D. Legrady and A. Cserkaszky and A. Wirth and B. Domonkos, Pet image reconstruction with on-the-Fly Monte Carlo using GPU. ANS, 2010, ISBN 9780894480799.

[8] Abramowitz, M. and Stegun, I.A., Handbook of Mathematical Functions. Dover Publications, 1972, ISBN 9780486612720.

[9] Audenaert, Koenraad, Inverse moments of univariate discrete distributions via the Poisson expansion. arXiv, 2008, eprint 0809.4155.

[10] Evans, M. and Hastings, N.A.J. and Peacock, J.B., Statistical Distributions. Wiley, 2000, ISBN 9780471371243.

Acknowledgements

I would like to thank my supervisor David Legrady for his patience and his help with both mathematical derivations and MATLAB simulations. He was an undepletable source of academic knowledge and technical know-how. Our weekly discussions spawned numerous ideas essential to this work and were a constant source of inspiration for me.

I would also like to thank Anita for her help with matters both related to and independent of my thesis.