CHAPTER 4

# A Markovian Data Reduction Framework

## 4.1 Introduction

The radiation damage correction models implemented thus far have all focussed on correcting reflection intensities. Scaling methods generally employ an average resolution dependent correction to all reflection intensities (**???**), whereas specific correction methods employ regression analysis on individual reflections (**??**). However correcting the reflection intensities presents many problems because each independent reflection exhibits its own behaviour, which is not necessarily linear, nor monotonic (**?**). Therefore, rather than addressing the problem of damage at the level of the reflection intensities, an alternative approach is to track the changes at the level of the sample undergoing the radiation damage.

This chapter presents a (parametric) time series model used to describe crystal changes during the X-ray crystallography experiment as a Markovian process. Within this framework existing algorithms - the Unscented Kalman filter and the Unscented Rauch-Tung-Striebel smoother - are used to determine the “optimal” values of the underlying crystal state, defined as the set of structure factor amplitudes, at each point in time during the MX experiment.

## 4.2 Why Julia?

The code for the algorithm presented in this chapter was all written in a recently released programming language called Julia. Given that this language is still in its beta version (it has yet to reach version 1 release) and is relatively unknown in the crystallography community, this choice may seem very unorthodox, so it is worth discussing why this language was chosen.

The total time of the implementation and run time of any piece of code can be crudely given as

$$\text{Total time} = \text{Computation time} + \text{Development time}$$

A compromise between the development time and the computation time has to be made, with the additional constraint that the developer's time is more important than the time of a computer. Given that the development of a new algorithm requires a lot of data and parameter exploration, writing prototype code in a low-level language such as C/C++ or Fortran would not necessarily be the optimum choice, particularly when the developer is unfamiliar with these languages. A popular alternative is to use a high-level dynamic language such as Python, R or Matlab, which are usually designed, or have packages to support technical computing. The productivity that can be achieved with these languages is counteracted by the relatively slow computation time when compared to low-level languages.

Julia is a dynamic language designed for technical computing (**??**) that is also very fast. Generally the computational performance of algorithms written in Julia executes within a factor of 2 of the speed of C (Figure 4.1). Therefore despite having to learn the Julia language, it seemed to be the best trade-off for coding the algorithm described here.

As a dynamic language that employs type inference, Julia code can be written in such a way that it compiles to non-optimal machine code.. This means that Julia has to be written in a particular manner to achieve the performance claimed by the language authors. A further argument for using Python or C/C++ over Julia is the large library of existing crystallographic packages that have been written in those languages. However Julia has built-in support for calling methods from C/Fortran libraries with a single line of code. Additionally, the `PyCall` package was written in Julia to allow Python libraries to be accessed directly from Julia with

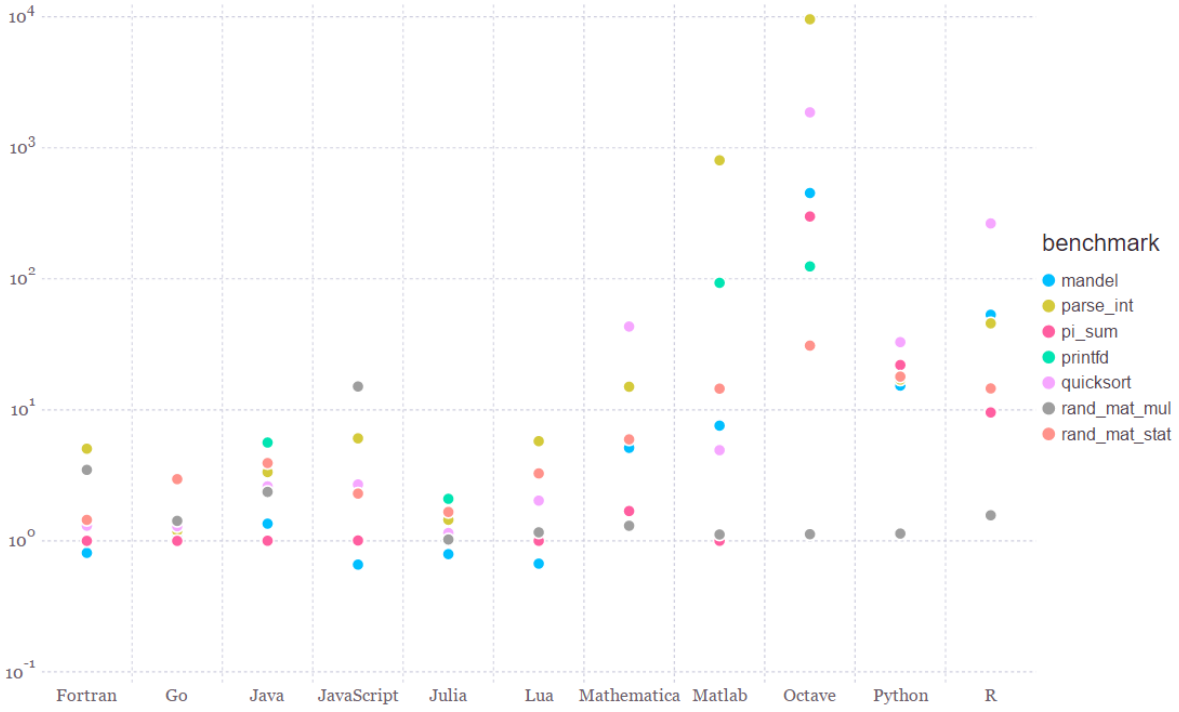



Figure 4.1: Benchmark times taken to run a given algorithm for various languages relative to C (smaller is better, C performance = 1.0). The Python implementations of rand_mat_stat and rand_mat_mul use NumPy (v1.9.2) functions, the rest are pure Python implementations. The table of data for this plot can be found on the main Julia programming language webpage, `http://julialang.org/`, along with a link to the plot.

a single line of code. Thus the crystallographic libraries were still easily accessible.

## 4.3 A hidden Markov model of the data collection experiment

The data collection experiment can be viewed as a time series: a sequence of diffraction data generated by a time-dependent process. Time series analysis seeks to understand the underlying processes that produced the data and allow forecasting or monitoring of the process. A more accurate analogy for the data collection experiment would be to describe it as a dose series, because the changes in the crystal state (and hence the observed data) are generally attributed to a dose dependent process.

Figure 4.2 shows a schematic of the dose series model of the experiment. At time t = i - 1 the crystal is in its initial state where the atoms have relatively well defined positions. After an initial X-ray exposure the crystal state has changed at time t = i. The atomic positions have changed and they have an effective smearing of their position due to their increased atomic B-factors. It is this state of the crystal that gives rise to the diffraction pattern at time t = i. The process repeats itself at time t = i + 1 and beyond. Thus each diffraction image can be regarded as being generated from a different but related crystal. In the model, the only components that are observed are the diffraction patterns, despite the fact that the crystal states are the desired quantities. The crystal states are effectively 'hidden'. Furthermore, the changes in crystal state as a result of X-ray exposure are assumed a Markovian process i.e. the state of the crystal at time i is completely determined by the state of the crystal at time i - 1. The model described here is known as a hidden Markov model.

The problem can now be stated as: **what is the most likely sequence of crystal states that generated the sequence of observed diffraction patterns?**

Before addressing the question, it is necessary to understand what is meant by the crystal state. The state of the crystal is defined by its constituent atoms and their positions. Equivalently the state of the crystal can be described by the set of structure factors in reciprocal space: amplitudes and their corresponding phases. However, at the data reduction stage of the crystallographic structure solution pipeline, the phases are completely unknown. Therefore the state of the crystal is represented solely by the set of structure factor amplitudes.

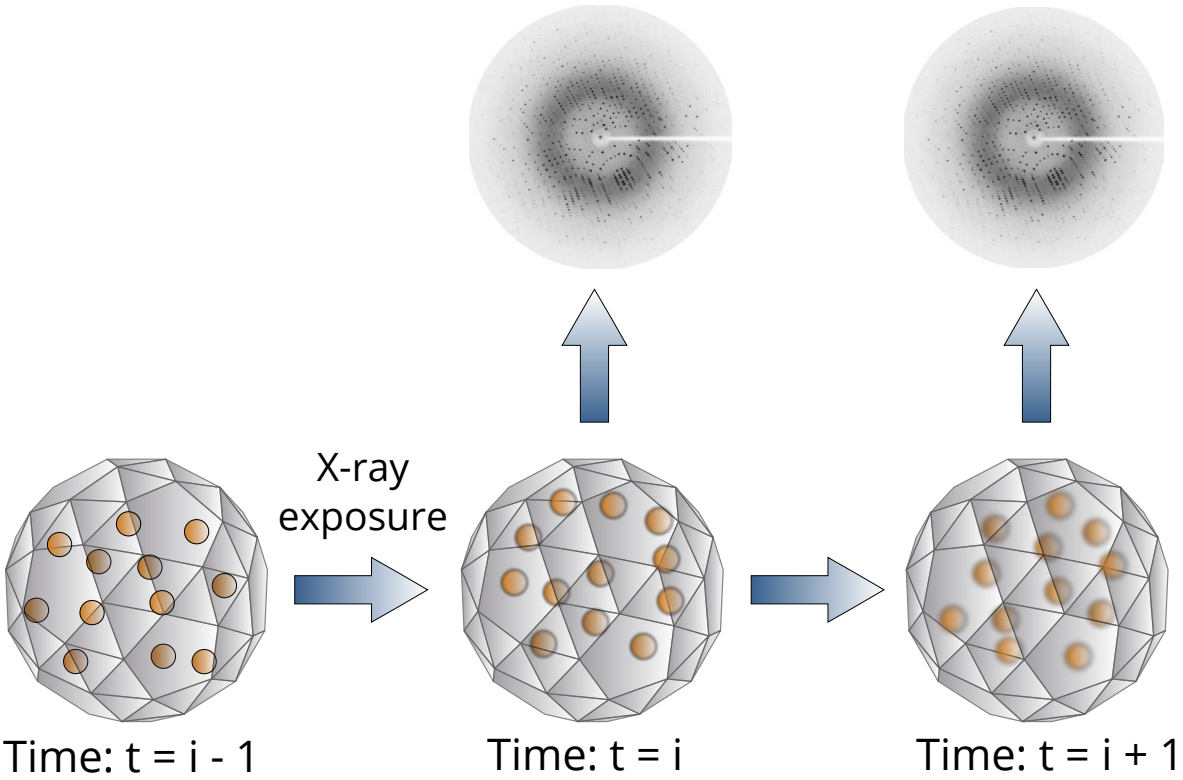



Figure 4.2: Hidden Markov model representation of the diffraction experiment. At time t = i - 1 the crystal is in an undamaged state and the constituent atoms have fairly well defined positions. After an X-ray exposure the crystal changes state at time t = i. Some of the constituent atoms change positions and their atomic B factors increase, which is represented by a slight blurring of the atoms. However the state of the crystal is not observed by the experimenter, instead the experimenter observes the diffraction image that is generated as a result of the exposure. This process repeats itself, typically until enough images have been collected to solve the structure.

### 4.3.1 Mathematical Notation

This chapter contains many formulas and symbols so a glossary of terms that define the notations for this chapter is provided below.

| Symbol | Definition |
|---|---|
| $\boldsymbol{F}_i$ | Structure factor of a reflection at discrete time point $i$. |
| $\lvert\boldsymbol{F}_i\rvert = F_i$ | Structure factor amplitude of a reflection at discrete time point $i$ |
| $F_c$ | Structure factor amplitude to be calculated using Bayesian inference. |
| $F_0$ | Initial structure factor amplitude calculated from a cycle of the forward-backward algorithm (FBA). |
| $\{\lvert\boldsymbol{F}\rvert\}_i$ | Set of structure factor amplitudes at discrete time point $i$ |
| $\{O\}_i$ | Set of intensity observations on an image collected at discete time point $i$ |
| $P_a(\boldsymbol{F}_{i+1};\boldsymbol{F}_i)$ | Probability of a structure factor, $\boldsymbol{F}_{i+1}$, of an acentric reflection at time point $i+1$ conditional on the structure factor value, $\boldsymbol{F}_i$, at time $i$. |
| $P_c(\boldsymbol{F}_{i+1};\boldsymbol{F}_i)$ | Probability of a structure factor, $\boldsymbol{F}_{i+1}$, of a centric reflection at time point $i+1$ conditional on the structure factor value, $\boldsymbol{F}_i$, at time $i$. |
| $I_i$ | Intensity of a reflection observed on the image generated at time point $i$. |
| $K$ | Scale factor which is assumed constant throughout the experiment. |
| $B_i$ | Scaling $B$ factor calculated from the image generated at time point $i$. |
| $\Delta\boldsymbol{r}$ | Average coordinate error. |
| $\boldsymbol{s}$ | Reciprocal space vector. |
| $\Delta B$ | Change in $B$ between consecutive time points $i$ and $i+1$. |
| $\Sigma_N$ | Expected intensity of a reflection. |
| $\sigma^2$ | = $(1-\lvert\boldsymbol{D}\rvert^2)\Sigma_N$ is the variance. |
| $\varepsilon$ | Multiplicity of a reflection. |
| $\boldsymbol{D}$ | = $\exp\left(-2\Delta B\frac{\sin^2(\theta)}{\lambda^2}\right)\cos\left(2\pi\boldsymbol{s}\cdot\Delta\boldsymbol{r}\right)$. (Although in the general formulation (see Read (1990)) $\boldsymbol{D}$ is a complex number is hence given a bold symbol). |
| $\boldsymbol{D}\boldsymbol{F}_i$ | The (complex) product of the structure factor, $\boldsymbol{F}_i$, with the multiplier, $\boldsymbol{D}$. |

### 4.3.2 Bayesian optimal filtering

Methods used to estimate the (hidden) state, $\boldsymbol{x}_j$, of a time-varying system observed indirectly via noisy measurements, $\boldsymbol{y}_j$, at a given point in time, $j$, are known as *optimal filtering* methods. The filtering distribution can be defined mathematically as

$$P(\boldsymbol{x}_j;\boldsymbol{y}_1,\ldots,\boldsymbol{y}_j), \tag{4.3.1}$$

i.e. the probability distribution of the state given all of the previous observations up to time point $j$. Filtering is sometimes regarded as a *forwards pass* through the data. There are many ways to define what is meant by *optimal* (**?**): here the optimality criterion is the minimum mean-squared error (MMSE) estimate of the state of the system. Bayesian filtering refers to the formulation of an optimal filter within a Bayesian framework. A Bayesian optimal filter can therefore be used to solve the problem of determining the crystal states given a set of (noisy) diffraction images. Due to the non-linear relationship ($I \propto |\boldsymbol{F}|^2$) between the reflection intensity, $I$, and the structure factor amplitudes, $|\boldsymbol{F}|$, it is necessary to use a non-linear Bayesian optimal filter for the crystallographic diffraction experiment problem. The filter chosen is the Unscented Kalman filter (UKF) because it propagates the probability density function in an effective manner (using the unscented transform) to achieve up to third order accuracy in the posterior mean and covariance estimates (**?**). The UKF algorithm is outlined in Wan and van der Merwe (2002).

### 4.3.3 Bayesian smoothing

Bayesian Smoothing can be considered to be a class of methods within the field of Bayesian filtering. Whereas filters generally compute estimates of the system state based on the observation history, smoothers use all of the available information and thus they can estimate states that happened before the current time (**?**). Mathematically the smoothing distribution is

$$P(\boldsymbol{x}_j; \boldsymbol{y}_1, \ldots, \boldsymbol{y}_j, \ldots, \boldsymbol{y}_\tau) \tag{4.3.2}$$

where $j < \tau$. Smoothers are regarded as *backwards passes* because they can be used to estimate states prior to the current time.

The goal of the data reduction stage in the macromolecular structure solution pipeline is to reduce the intensity values to accurate estimates of the structure factor amplitude for each reflection. If this problem is phrased in a probabilistic manner, then the distribution of interest is

$$P(\{|\boldsymbol{F}|\}_i\,;\, \{O\}_1, \ldots, \{O\}_i, \ldots, \{O\}_\tau), \tag{4.3.3}$$

where $\{|\boldsymbol{F}|\}_i$ is the set of structure factor amplitudes and $\{O\}_i$ is the set of intensity measurements, $I_{hkl}$ at time point $i$. Equation 4.3.3 is the probability distribution that describes

the values of the set of structure factor amplitudes given all of the observed data on the diffraction images. Comparison of equations 4.3.3 and 4.3.2 show that they are identical and hence using a Bayesian smoother (after application of the UKF) should provide the desired solution to the data reduction problem. The application of both the forwards and backwards passes is known as the *forward-backward algorithm*. The important difference of this formulation compared to the current data reduction methods is that the set of structure factor amplitudes is found for every time point, $i$, as opposed to just producing a single set of amplitudes from the data.

The Bayesian smoother chosen for this problem was the unscented Rauch-Tung-Striebel smoother (URTSS) (**?**).

### 4.3.4 Process function and covariance

In order to apply the UKF it is necessary to define the function that relates the crystal state at time point i to the crystal state at time point i + 1. This function is known as the process function, and is typically taken as the expected value of a conditional probability distribution (transition probability) of the probability of crystal state at i + 1 given the crystal state at i. The process covariance is also an important quantity because it effectively describes the level of uncertainty of the process.

The crystal state changes as a result of the X-ray exposure (the duration of which will differ for a single diffraction image depending on the goal of the experiment) and the magnitude of these changes will vary depending on the radiation sensitivity of the irradiated crystal. It is assumed that X-ray irradiation on the timescale of image collection is short enough such that the change in crystal state is fairly small. Thus the crystal state at one image is closely related to the crystal state for the subsequent image. This assumption along with the assumptions that changes in structure factors are independent* and identically distributed, gives the Luzzati distributions (**???**)

$$P_a(\boldsymbol{F}_{i+1};\boldsymbol{F}_i) = \frac{1}{\pi\varepsilon\sigma^2}\exp\left(-\frac{|\boldsymbol{F}_{i+1}-\boldsymbol{D}\boldsymbol{F}_i|^2}{\varepsilon\sigma^2}\right), \tag{4.3.4}$$

*In reality structure factors are not independent, however this assumption simplifies the equations significantly and still provides useful information (**?**).

for acentric reflections and

$$P_c(\boldsymbol{F}_{i+1};\boldsymbol{F}_i) = \frac{1}{[2\pi\varepsilon\sigma^2]^{1/2}}\exp\left(-\frac{|\boldsymbol{F}_{i+1}-\boldsymbol{D}\boldsymbol{F}_i|^2}{2\varepsilon\sigma^2}\right), \tag{4.3.5}$$

for centric reflections where $P(\boldsymbol{F}_{i+1};\boldsymbol{F}_i)$ denotes the probability of structure factor, $\boldsymbol{F}_{i+1}$, at time i + 1 given the structure factor, $\boldsymbol{F}_i$, at time i, $\varepsilon$ is the multiplicity of the reflection, $\sigma^2 = (1-|\boldsymbol{D}|^2)\,\Sigma_N$, where $\Sigma_N$ is the expected intensity of the reflection. In this work the expected intensity value was calculated as the sum of the squared atomic scattering factors, and $\boldsymbol{D}$ is a (complex) multiplier, which quantifies the effects of crystal perturbations on the structure factor and is explicitly defined in section 4.3.6.

As discussed previously, the phases are unknown during the data reduction stage and hence the only quantity that can be inferred is the amplitude of the reflection. The unknown phase can be integrated over equations 4.3.4 and 4.3.5 (marginalisation) to obtain the probability of the structure factor amplitudes (transition probability):

$$P_a(|\boldsymbol{F}_{i+1}|;|\boldsymbol{F}_i|) = \frac{2|\boldsymbol{F}_{i+1}|}{\varepsilon\sigma^2}\exp\left(-\frac{|\boldsymbol{F}_{i+1}|^2+\boldsymbol{D}^2|\boldsymbol{F}_i|^2}{\varepsilon\sigma^2}\right)I_0\left(\frac{2|\boldsymbol{F}_{i+1}|\boldsymbol{D}|\boldsymbol{F}_i|}{\varepsilon\sigma^2}\right), \tag{4.3.6}$$

$$P_c(|\boldsymbol{F}_{i+1}|;|\boldsymbol{F}_i|) = \left[\frac{2}{\pi\varepsilon\sigma^2}\right]^{1/2}\exp\left(-\frac{|\boldsymbol{F}_{i+1}|^2+\boldsymbol{D}^2|\boldsymbol{F}_i|^2}{2\varepsilon\sigma^2}\right)\cosh\left(\frac{|\boldsymbol{F}_{i+1}|\boldsymbol{D}|\boldsymbol{F}_i|}{\varepsilon\sigma^2}\right) \tag{4.3.7}$$

where $I_0(\cdot)$ is the zero order modified Bessel function of the first kind and $\cosh(\cdot)$ is the hyperbolic cosine function. Note equation 4.3.6 is known as the Rice distribution and equation 4.3.7 is known as the Woolfson distribution (**??**).

The mean (process function), $\mu_{Rice}$, and variance (process variance), $\sigma^2_{Rice}$, for acentric structure factor amplitudes can are calculated by integrating equation 4.3.6 to give

$$\mu_{Rice} = \sigma\sqrt{\frac{\pi}{2}}L_{1/2}\left(-\frac{\boldsymbol{D}^2|\boldsymbol{F}_i|^2}{2\sigma^2}\right), \tag{4.3.8}$$

$$\sigma^2_{Rice} = 2\sigma^2+\boldsymbol{D}^2|\boldsymbol{F}_i|^2-\frac{\pi\sigma^2}{2}L^2_{1/2}\left(-\frac{\boldsymbol{D}^2|\boldsymbol{F}_i|^2}{2\sigma^2}\right), \tag{4.3.9}$$

where

$$L_{1/2}(x) = \exp(x/2)\left[(1-x)I_0\left(\frac{-x}{2}\right)-xI_1\left(\frac{-x}{2}\right)\right] \tag{4.3.10}$$

is the Laguerre polynomial and $L^2_{1/2}$ denotes the square of the Laguerre polynomial $L_{1/2}$. $I_1(\cdot)$ is the first order modified Bessel function of the first kind (**?**). For strong reflections the

corresponding Rice distribution for the amplitude can be approximated well with a Gaussian function. In this case the process function and covariance become

$$\mu_{Gauss} = \boldsymbol{D}\boldsymbol{F}_i \tag{4.3.11}$$

$$\sigma^2_{Gauss} = \left(1 - |\boldsymbol{D}|^2\right)\Sigma_N. \tag{4.3.12}$$

For centric reflections the covariance is simply twice the variance for acentric reflections (**?**). The process function for centric reflections is assumed identical to the process function for acentric reflections for the sake of simplicity. For strong reflections this assumption is valid, since the Gaussian distributions (equations 4.3.4 and 4.3.5) differ only in the variance. For weak reflections this assumption breaks down and the expected value of the Woolfson distribution should be explicitly calculated.

### 4.3.5 Observation function and covariance

In addition to the process function, it is necessary to define the process by which diffraction images are generated from the crystal state. This is known as the observation function. In an analogous manner to the process function, the observation function is taken as the expected value of a conditional probability distribution (emission probability) describing the probability of the intensity of a reflection $I_i$, given the structure factor amplitude $|\boldsymbol{F}_i|$ at time i. The observation model is given by (**?**)

$$I = K|\boldsymbol{F}|^2, \tag{4.3.13}$$

where $K$ is the scale factor. However, due to the measurement process being inherently noisy, the process is better approximated as a probability distribution. Assuming a normally distributed measurement error, the emission probability is given by

$$P(I_i; |\boldsymbol{F}_i|) = \frac{1}{\sigma_m\sqrt{2\pi}} \exp\left(-\frac{(I_i - K|\boldsymbol{F}_i|^2)^2}{2\sigma_m^2}\right), \tag{4.3.14}$$

where $\sigma_m^2$ is the variance of the measurement process, which is given as a result of the integration process. Not all reflections are observed on a diffraction image, so these observations are not given a variance value by the integration software. The variance for these

missing reflections is therefore given a large value (e.g. $10^{10}$) to effectively represent an infinite uncertainty on the observation for the image.

### 4.3.6 Obtaining parameter values for the process and observation functions

The process and observation functions are parameterised by $\boldsymbol{D}$ and $K$ respectively and hence the values of these parameters must be determined. The multiplier $\boldsymbol{D}$ is given by (**??**)

$$\boldsymbol{D} = \exp\left(-2\Delta B \frac{\sin^2(\theta)}{\lambda^2}\right) \cos\left(2\pi \boldsymbol{s} \cdot \Delta \boldsymbol{r}\right). \tag{4.3.15}$$

where $\Delta B$ is the change in scaling B factor and $\Delta \boldsymbol{r}$ is the average coordinate error from time point i to time point i + 1, $\theta$ is the Bragg angle and $\lambda$ is the wavelength of the incident X-ray. (All references to the B factor in the rest of the chapter refer to the scaling B factor unless otherwise stated). Implicit to the form of $\boldsymbol{D}$ in equation 4.3.15 is the fact that the B-factor is assumed to be isotropic. Furthermore, it is assumed that the irradiation process only changes the B factor (not the coordinate error of the atoms i.e. $\Delta \boldsymbol{r} = 0$) and the change in B-factor is the same for every atom in the structure (the Wilson B-factor). This reduces equation 4.3.15 to

$$\boldsymbol{D} = \exp\left(-2\Delta B \frac{\sin^2(\theta)}{\lambda^2}\right). \tag{4.3.16}$$

Thus $\boldsymbol{D}$ is ultimately determined by the change in $B$ factor, $\Delta B$.

The B and scale factor can be determined using the scaling equation

$$I_{obs} = K \sum_j f_j^2 \exp\left(-2B \frac{\sin^2(\theta)}{\lambda^2}\right), \tag{4.3.17}$$

where $I_{obs}$ is the observed intensity and $f_j$ is the atomic scattering factor for an atomic species within the unit cell, which can be calculated for a given reflection using the Cromer-Mann coefficients (**?**). Taking the natural logarithm of both sides and rearranging yields.

$$\ln\left(\frac{I_{obs}}{\sum_j f_j^2}\right) = \ln(K) - 2B \frac{\sin^2(\theta)}{\lambda^2}. \tag{4.3.18}$$

Plotting $\ln\left(\frac{I_{obs}}{\sum_j f_j^2}\right)$ against $\frac{\sin^2(\theta)}{\lambda^2}$ gives a straight line where $\ln(K)$ is the intercept and $-2B$ is the gradient. This procedure can be performed for each image to obtain a sequence of $K$

and $B$ values. These values will be noisy because no single image contains the entirety of reciprocal space (i.e. no image contains all reflections). Assuming the scale and B functions to be smooth, continuous functions can be fitted through the values obtained from the data to get estimates of the true scale and B factors for each image. For simplicity the B factor is assumed linear and the scale factor is assumed constant. The assumption of linearity for the B factor is valid for low dose ranges (**???**) and hence $\Delta B$ can be given as the difference in B factor between images. On the other hand the constant scale factor assumption is not valid for general MX experiments, but may be a suitable approximation for the case where a crystal is completely immersed in a top-hat profile X-ray beam, as for instance was the case for the data collection described in Chapter **??**.

### 4.3.7 Convergence of the forward-backward algorithm

The initial estimate of the structure factor amplitude required to start the forward-backward algorithm may not be close to the true value. The algorithm will thus propagate the incorrect amplitude value through the filter until the time point when the first actual observation is made. From then on the estimates should be closer to the true values of the amplitude in the experiment. In particular, the backwards pass should result in a better estimate of the true initial amplitude. The improved initial amplitude estimate can then be provided to the forward-backward algorithm again to obtain better estimates of the amplitude evolution of a reflection, including a further improved initial structure factor amplitude value. Hence the procedure is iterative.

The question then becomes *"When has the solution reached convergence?"*

#### Log Likelihood

One way to determine when the solution has converged is to determine the point at which the increase in the log likelihood is smaller than a given tolerance. The likelihood is a function describing how likely the data observed are to occur given the current model. For the Kalman filter it is defined as (**?**)

$$L = P(F_0) \prod_i P(I_i|F_i) \times P(F_i|F_{i-1}), \tag{4.3.19}$$

where $F_i = |\boldsymbol{F}_i|$ is the structure factor amplitude of a reflection and $I_i$ is the intensity of the reflection at time point $i$. Initially it seems that $P(I_i|F_i)$ and $P(F_i|F_{i-1})$ should represent the emission and transition probabilities respectively. However this is not exactly the case because the UKF propagates Gaussian models. Thus the mean and covariance of the states calculated at each time point in the HMM are used as parameters for the Gaussian distribution that is used as an approximation of the true crystal state (Figure 4.3). Hence $P(I_i|F_i)$ and $P(F_i|F_{i-1})$ are Gaussian distributions. The log likelihood is computationally more convenient to deal with (and often analytically too) than the likelihood, and hence it is the log likelihood that is calculated instead of the likelihood.

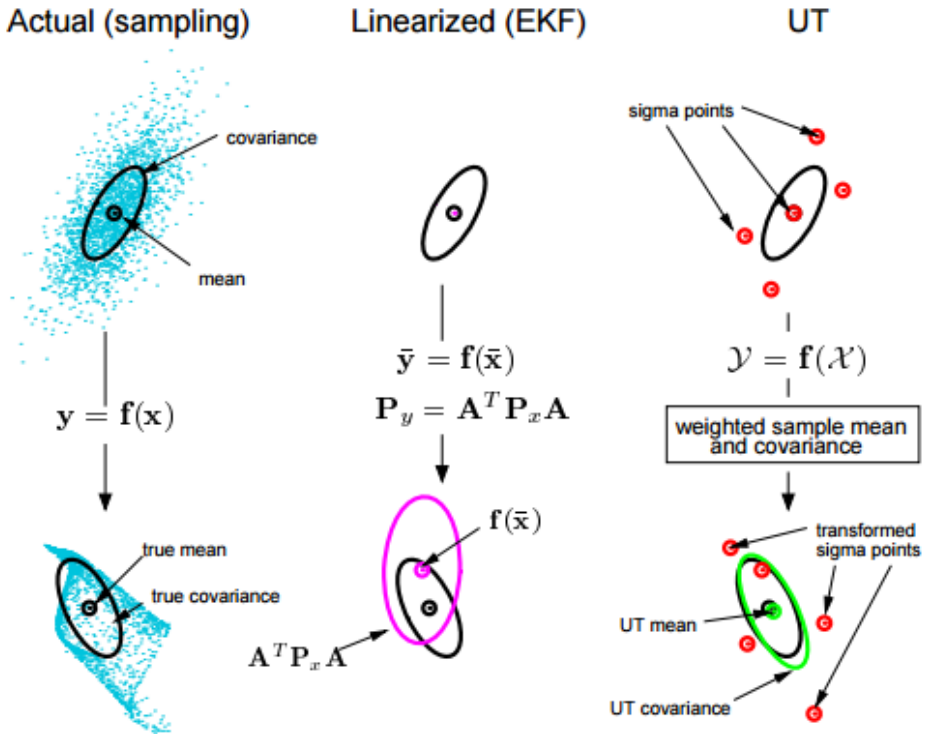



Figure 4.3: Propagation of states $\mathbf{x}$ through a non linear function $\mathbf{f}$ for different transformation techniques. The left plot shows the true mean and covariance propagation which uses Monte-Carlo sampling. The middle shows the linearisation performed by the extended Kalman filter (EKF), another non-linear Kalman filter. The right plot shows the propagation performed with the unscented transform (UT) used by the UKF. The performace of the UKF is superior to that of the EKF and uses many fewer sampling points (called sigma points) than Monte-Carlo sampling.

### 4.3.8 Summary of hidden Markov model formulation

In summary, in the current work the diffraction experiment has been represented as a hidden Markov model where a process function describes how the (hidden) state of the crystal at a particular time point i generates the consecutive crystal state due to X-ray irradiation. An observation model also describes how the crystal, which is in a particular state, generates

the observed intensities. The UKF and the URTSS algorithms together give the forward-backward algorithm, which is designed to give the optimal sequence of crystal states that best describe the observed data, consistent with the defined process and observation functions. The process and observation (co)variances quantify the level of uncertainty of the corresponding processes. The forward-backward algorithm is applied iteratively to obtain the improved estimates of the structure factor amplitude of a reflection with each iteration. The log likelihood is calculated for each cycle, and when the improvement of the log likelihood is smaller than a given tolerance value, the solution is considered to have converged. The final result of the algorithm is a sequence of the set of structure factor amplitudes for each time point in the data collection experiment.

## 4.4 Extraction and treatment of reflection intensity data

### 4.4.1 Allocating observations to images

The algorithm presented was applied to the data that were collected on a crystal of bovine pancreatic insulin (Crystal ID 0259) as described in Chapter **??**. However, before the forward-backward algorithm can be applied, the data for each full observation have to be extracted and allocated to a single diffraction image. An MTZ file containing the integrated data from the set of diffraction images was produced by processing the images with MOSFLM (**?**). MTZDUMP, a program from the CCP4 software suite (**?**), was used to extract the data from the integrated MTZ file, with additional commands to obtain the space group symmetry and image (batch) information. A custom script was written to parse the MTZDUMP output and extract the observation information. Each image is given a *rotation start angle* (RSA) and a *rotation end angle* (REA) and each observation has a *rotation centroid* value. An observation is allocated to an image if its centroid value lies between the image's RSA and REA. For fully recorded reflections this is straight forward because the entire observation is recorded on a single image. This is not the case for partially observed reflections i.e. when a single reflection observation is partially measured on multiple images. Each detection of a partially measured reflection is given an estimate of the rotation centroid of the full observation. In theory this centroid value should be the same for each measurement of the same reflection, but in practice this is rarely the case. To determine the actual centroid value, the mean average of all centroid values of each measurement is calculated, and this value is used to allocate a partial reflection observation to a given image.

At the beginning and end of a data collection experiment, some reflections have not been completely traversed, resulting in observations where the calculated rotation centroid lies outside the oscillation range of the data collection. In these cases the reflection is allocated to either the first or last image if the calculated centroid is smaller than the RSA of the first image or the REA of the last image respectively.

### 4.4.2 Treatment of intensity data

#### Extracting intensity values for fully traversed reflections

When reflection observations are integrated with MOSFLM, two intensity estimates are calculated: a profile fitted intensity and a summed intensity. The profile fitted intensity value is generally a better estimate and this is especially true for weak data. On the other hand, the summed estimate should be more accurate for the strongest reflections (`http://www.ccp4.ac.uk/html/aimless.html`). In exactly the same manner as utilised in AIMLESS (**?**), the custom written parser can use either of the two intensity estimates but defaults to using a combination of the two. The approach of combining the two is to calculate a weighted average of the two intensity estimates such that the profile fitted estimate is weighted higher for weak reflections and vice versa for strong reflections. The equation used to extract the combined intensity, $I_{com}$ is

$$I_{com} = wI_{pr} + (1 - w)I_{sum}, \tag{4.4.1}$$

where $I_{pr}$ is the profile fitted intensity estimate, $I_{sum}$ is the summed intensity estimate and $w$ is the weight defined as

$$w = \frac{1}{1 + \left(\frac{I_{raw}}{I_{mid}}\right)^{I_{pow}}}. \tag{4.4.2}$$

In AIMLESS, $I_{pow}$ defaults to 3, $I_{raw}$ is the intensity value before Lorentz-Polarisation (LP) correction and $I_{mid}$ is optimised to give the best overall $R_{meas}$ value. The custom parser uses $I_{mid} = (I_{pr} + I_{sum})/2$, $I_{raw} = I_{mid}\times$LP and $I_{pow} = 3$.

The intensity value (either $I_{com}, I_{pr}$ or $I_{sum}$) is calculated for each measurement of a reflection observation. For full reflections the resulting intensity value is used as the full observation intensity. For partial reflections this value has to be summed for each measurement of the same reflection observation to obtain the full estimate.

#### Estimating the intensity of non-fully traversed reflections

Some reflection observations are not fully traversed and hence the intensity values for these reflections are incomplete. Estimates of the true intensity of these reflections can be made by assuming a standard uniform shape for the reflection. If the reflection is assumed spher-

ical in shape then the measured fraction of the reflection is a spherical cap, shown in Figure 4.4. The ratio of the volume of the spherical cap to the volume of the sphere, $p$ is given

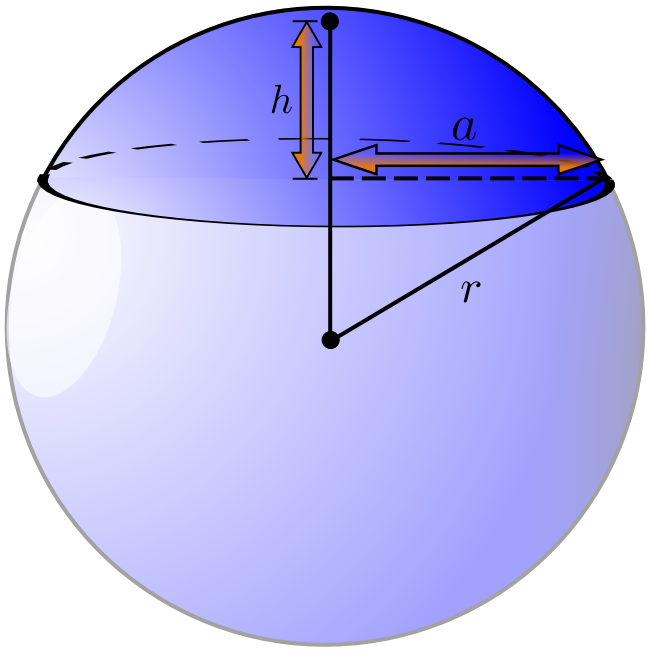



Figure 4.4: Model of the spherical cap traversed in a data collection experiment. The translucent volume is the volume that was not recorded. $h$ represents the height of the spherical cap, $a$ represents the radius of the circular base of the cap and $r$ is the radius of the sphere.

by

$$p = \frac{h^2}{d^3}(3d - 2h), \tag{4.4.3}$$

where $d$ is the diameter of the sphere and $h$ is the height of the spherical cap. If the height of the spherical cap is given as a fraction of the diameter, denoted $q$, and the diameter is set to 1, then equation 4.4.3 becomes

$$p = 3q^2 - 2q^3, \tag{4.4.4}$$

which is the same formula as given in Rossman *et al.* (1979). Angles are used as a proxy for the actual lengths $h$ and $d$ because the lengths are not given. For reflections where the calculated centroid is before the RSA of the first image, $h$ is approximated as the absolute difference between the RSA of the first image and the mid point of the RSA and REA of the last image on which the reflection was observed. The spherical diameter of the reflection, $d$ is approximated as twice the absolute difference between the mid point of the RSA and REA of the last image on which the reflection was observed and the rotation centroid. The analogous values are used for reflections where the reflection centroids were calculated

beyond the REA of the last image.

### Quantifying additional uncertainty in the observation variance

The standard deviation for each measurement is also calculated and provided in the output by MOSFLM. Again, for full reflections this value can be used as the final standard deviation for each measurement. However, for partial reflections the standard deviations for each partial measurement have to be combined. This is achieved by summing the variances for each partial measurement giving a total variance denoted $\sigma^2_{sum}$.

However, two more factors complicate the variance calculation. Firstly, it is acknowledged that the crystal is in a slightly different dose state at each image: hence in combining the variances it is also necessary to increase the uncertainty due to the change in crystal state between images. This additive uncertainty factor is calculated as

$$\sigma^2_{im} = \sum_{i}^{\text{images}} (1 - (\boldsymbol{D}_i^{im})^2) I_i^{im}, \tag{4.4.5}$$

where the sum is over all images on which the measurements of an observation are recorded, $I_i^{im}$ is the measured intensity of the observation a image $i$ and $\boldsymbol{D}_i^{im}$ is defined as

$$\boldsymbol{D}_i^{im} = \exp(-2|\Delta B_i^{diff}| \sin^2(\theta)/\lambda^2)), \tag{4.4.6}$$

where $\Delta B_i^{diff}$ is the difference in B factor between image $i$ and the centroid image. The explicit calculation and results of the B factor analysis is given in sections 4.6.1 and 4.6.2.

Secondly, the total fraction of each measurement is calculated for each reflection (denoted `FRACTIONCALC` in the MTZ column from MOSFLM). The sum of these values for partial measurements of an individual observation should be equal to 1. This is rarely the case because these values are not accurate. In AIMLESS the criteria for an observation to be regarded as fully recorded over its partial measurements is if the sum of the `FRACTIONCALC` is bewteen 0.95 and 1.05. The criterion used by the custom parser script only flags observations where the sum is less than 0.95. If this is the case then the observations can either be rejected, or the variance of the intensity value can be further inflated by a value proportional to the

inverse of the total calculated fraction. Explicitly the additive factor is calculated as

$$\sigma_{fr}^2 = \varepsilon \times \Sigma \times (1 - fr_{tot}), \tag{4.4.7}$$

where $\varepsilon$ is multiplicity of the reflection, $\Sigma$ is the expected intensity value and $fr_{tot}$ is the sum of the individual calculated fractions for each partial measurement.

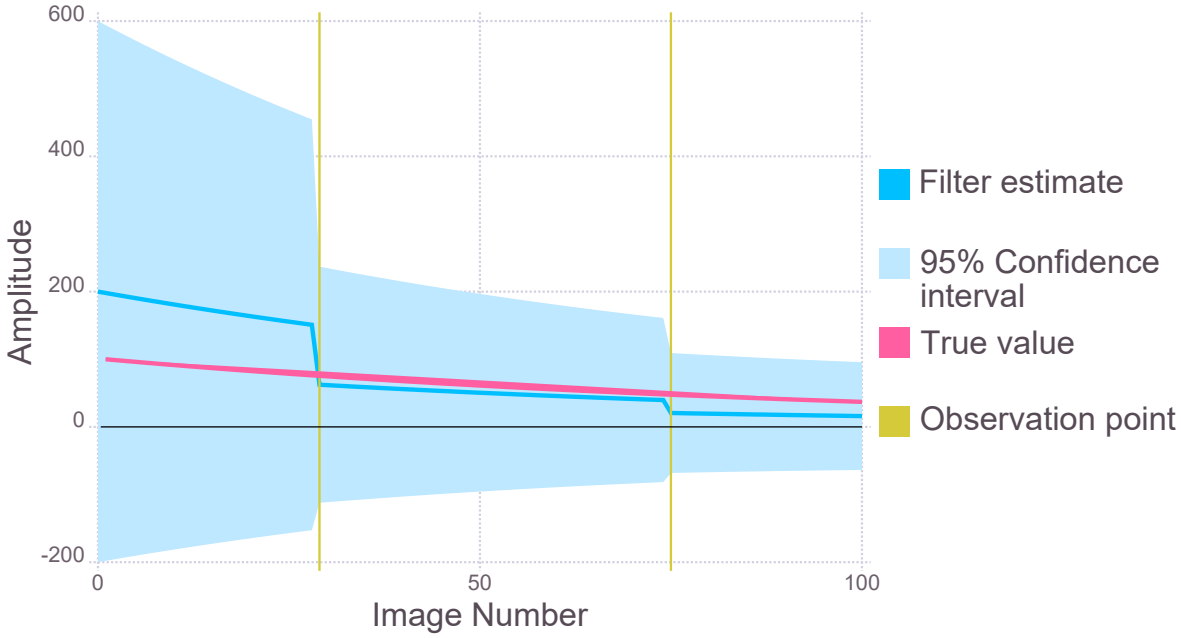
The final variance for a single observation is then given by

$$\sigma^2 = \sigma_{sum}^2 + \sigma_{im}^2 + \sigma_{fr}^2. \tag{4.4.8}$$

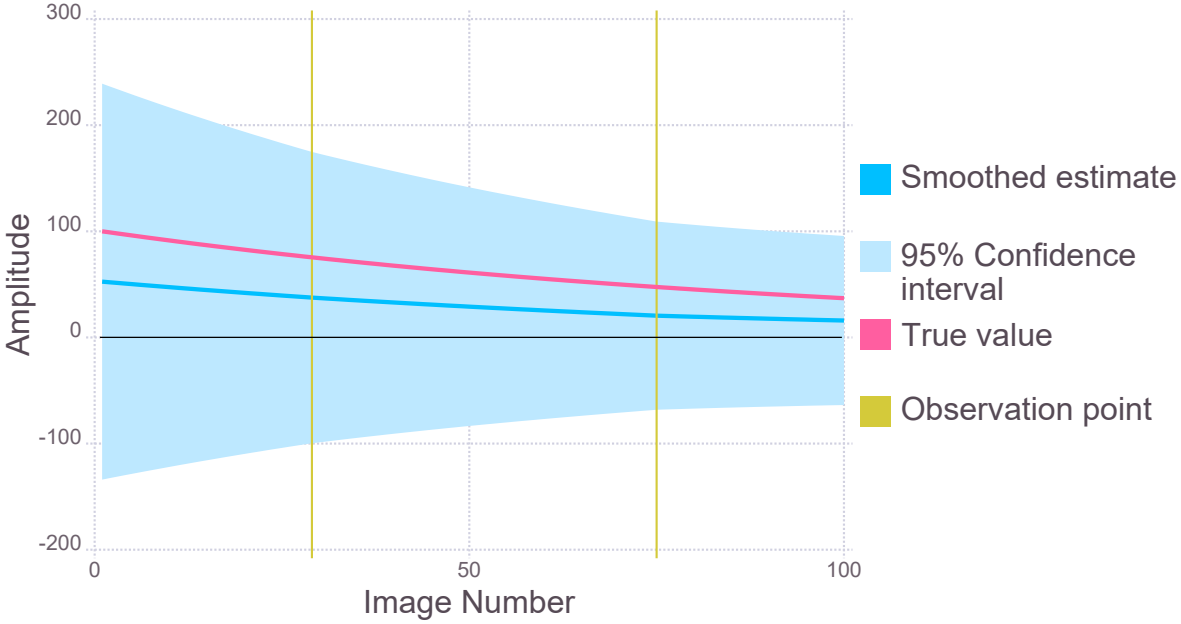
## 4.5 Simulation results

A simulation of the behaviour of a reflection in a diffraction experiment was performed to test the performance of the forward-backward algorithm. In the simulation, 100 images were recorded in a diffraction experiment where the intensity of a particular reflection was observed twice, once on the 27th image and again on the 76th image with simulated Gaussian noise. The observed intensities were 71.43 and 40.13 on images 27 and 76 respectively. The true structure factor amplitude of the reflection is initially 100 and it decays by 1% after each image is collected (whether it is observed or not). The forward-backward algorithm is applied where the process function is defined such that the amplitude decays by 1% for each image and the observation function is defined as in equation 4.3.13. The estimate supplied to the forward-backward algorithm for the initial amplitude is 200, double the value of the true value of 100. The results of the forward-backward algorithm for cycles 1, 2 and 10 are shown in Figure 4.5.

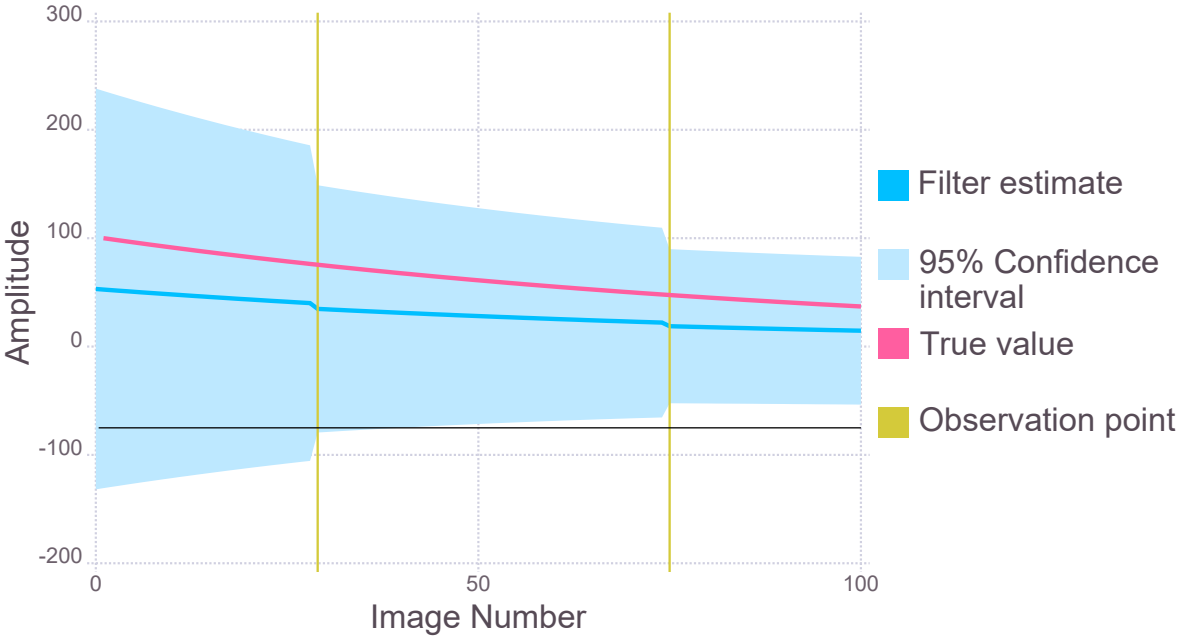
At the very beginning of the algorithm, the filtering estimates (solid blue line Figure 4.5a) do not predict the true value (solid pink line) very well. This is because no observations are made until the 27th image, therefore the estimates propagate according to the defined process function. At image 27 the first observation is made, represented by the vertical solid gold line, which is when the filtering estimate approaches the amplitude value required to produce the observed intensity according to the observation equation 4.3.13. The forward pass continues to propagate the estimates according to the process function until it reaches the second observation and sharply changes value to what it deems optimum. The smoothing algorithm attempts to consolidate the optimal values found during the forwards pass whilst maximising the probability of the estimate of the state at time i producing the state at time i + 1. This results in smoother state predictions, as evidenced by the smoother blue solid line in Figure 4.5b, and a reduced overall uncertainty (the light blue shaded region represents the 95% confidence interval). The initial value found during the smoothing is also closer to the true value. The second forward-backward pass shows the same characteristics, whereby the smoother gives a smaller uncertainty estimate and smoothes the values from the filtering pass (Figures 4.5c and 4.5d). After 10 forward-backward cycles the smoothed estimates are very close to the true values (Figure 4.5f). Importantly the uncertainty values are smallest at the points where the observations are made, as expected.



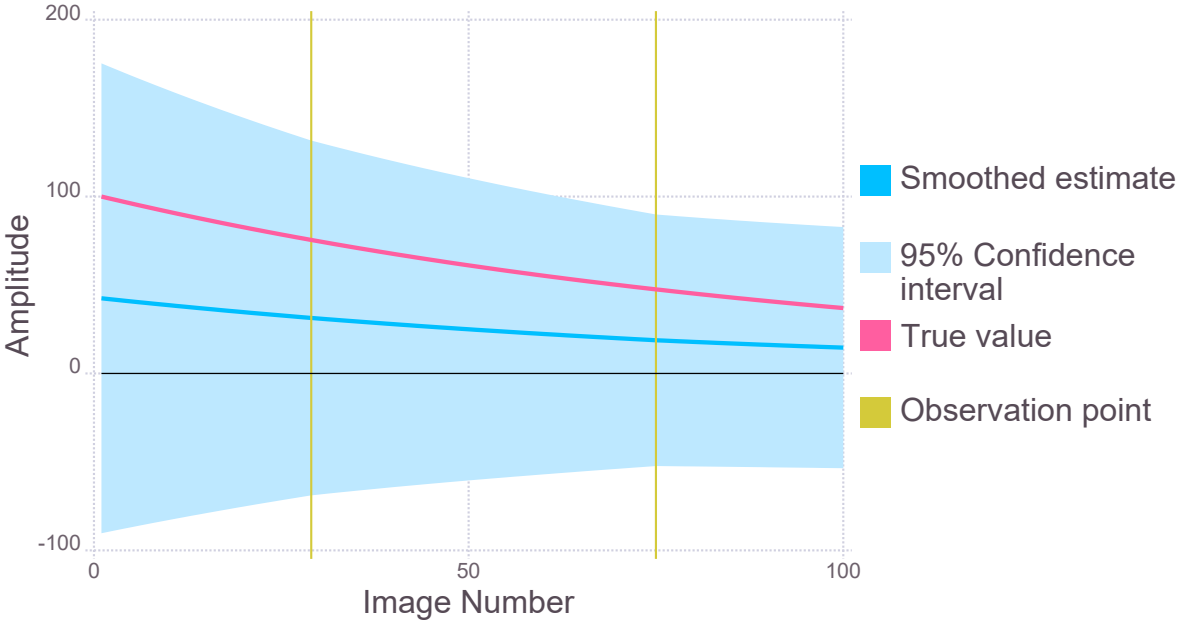

(a) Cycle 1: forward pass.



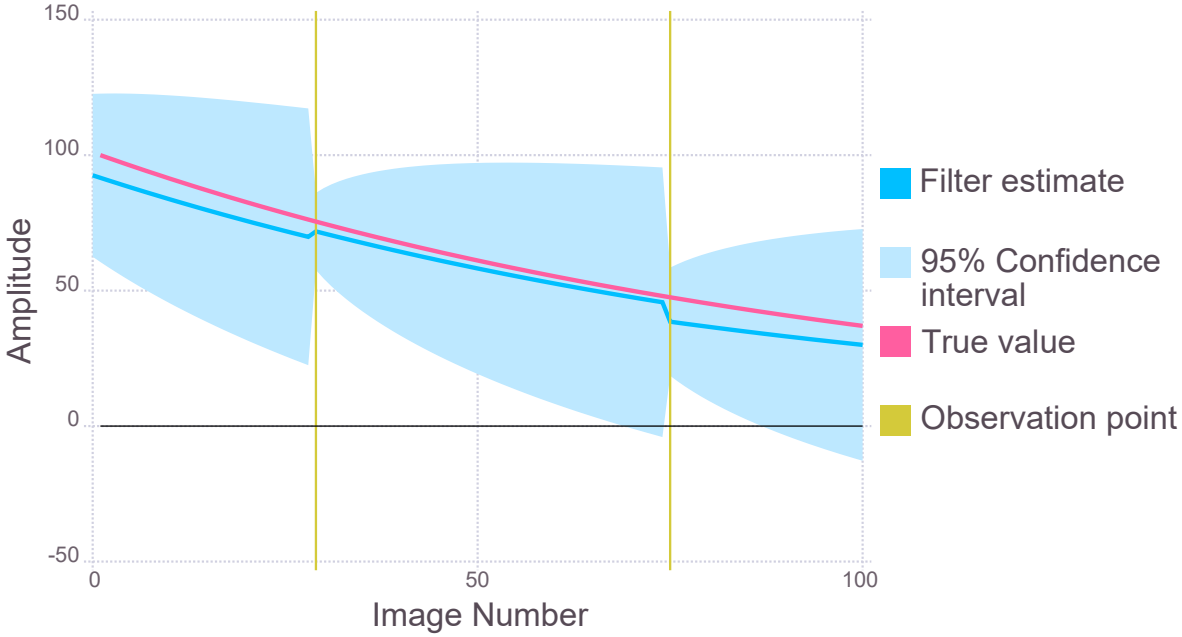

(b) Cycle 1: backward pass.



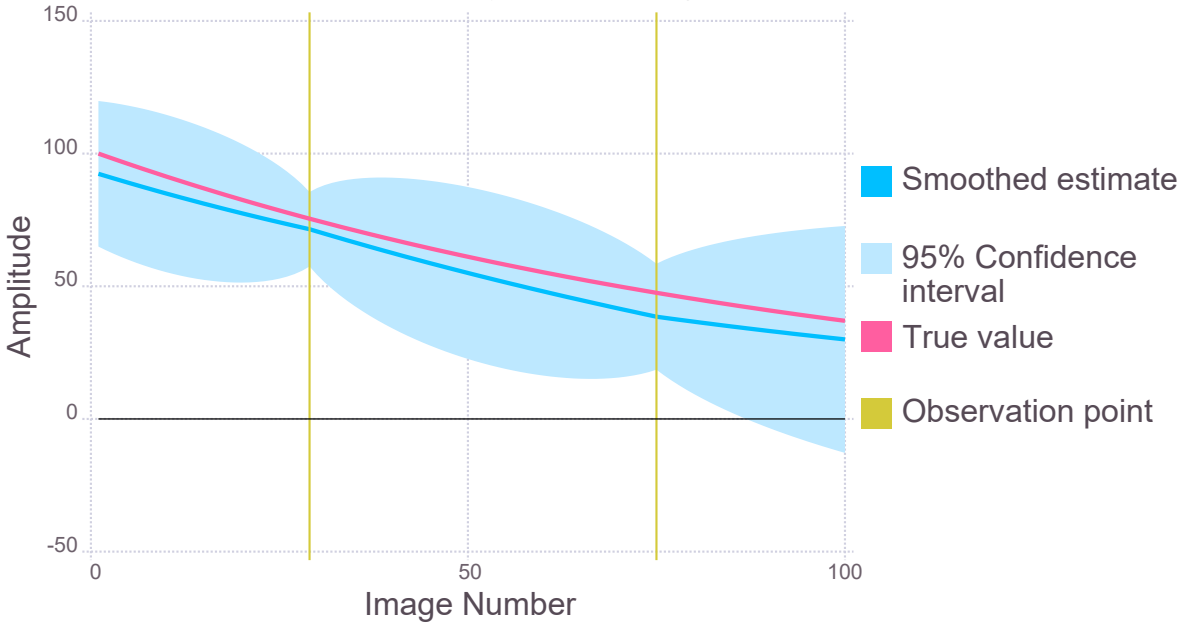

(c) Cycle 2: forward pass.




(d) Cycle 2: backward pass.




(e) Cycle 10: forward pass.




(f) Cycle 10: backward pass.

Figure 4.5: Forward-backward algorithm results for a simulated reflection observed on image 27 and 76 (solid gold lines) of a dataset consisting of 100 images. As the number of cycles increases, the forward-backward estimate (blue solid line) approaches the true value (pink solid line) and the 95% confidence interval decreases (light blue shaded region).

The log likelihoods for the 10 cycles, calculated using the natural logarithm of equation 4.3.19, were also computed and are shown in Figure 4.6.

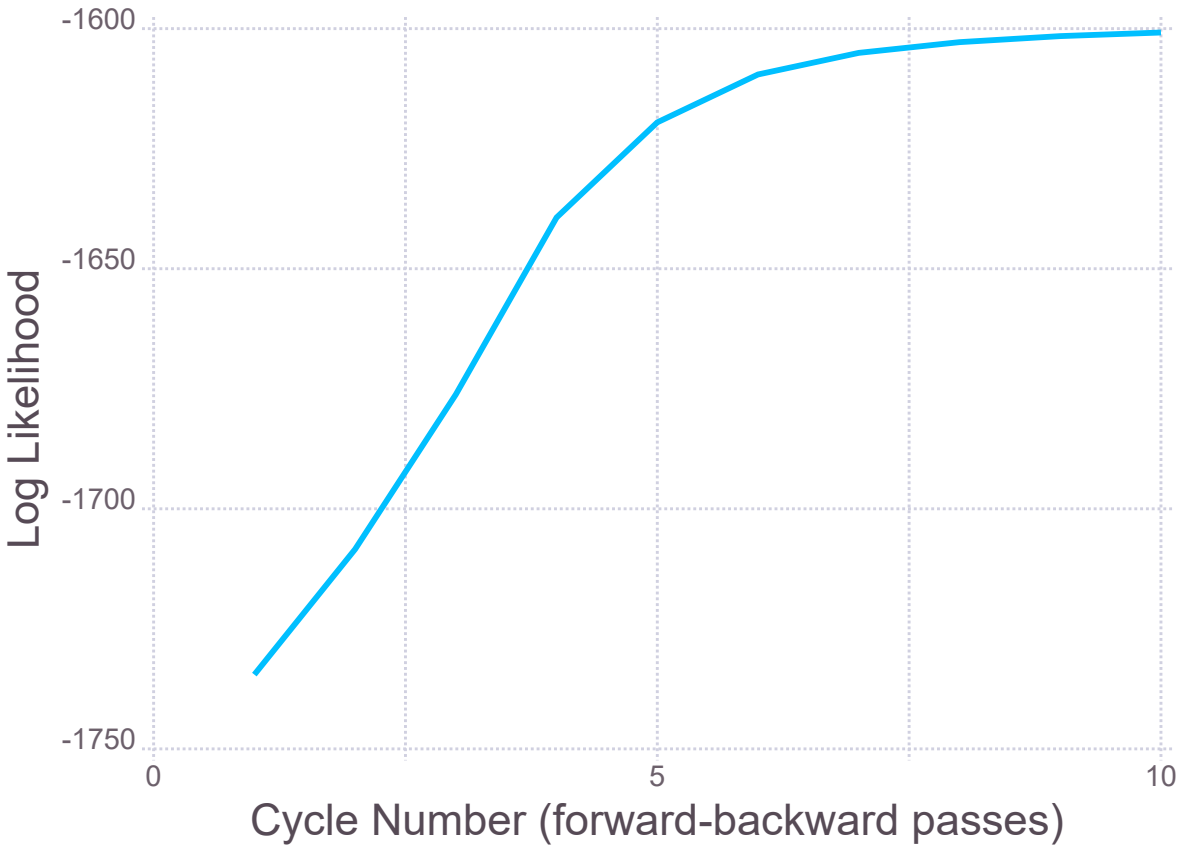



Figure 4.6: Log likelihood values calculated for each smoothing pass. As the cycle number increases the log likelihood starts to plateau meaning that the forward-backward solution is converging on a final solution.

It can be seen that as the number of forward-backward passes increases, the rate of increase (gradient) tends to zero thus showing that the smoothed estimates are converging to the optimal solution for the observed data.

### 4.5.1 Weak data

Simulations of the forward-backward algorithm showed that the method works very well for strong reflections. A further simulation was performed to see how effective the algorithm would be for weak reflections. Again the simulation consisted of 100 images where the intensity of a particular reflection was observed twice, this time on the 41st image and the 99th image with simulated Gaussian noise. The true structure factor amplitude of this reflection is initially 10 and again decays by 1% after each image is collected (whether it is observed or not). The same process and observation functions are defined but the estimate

supplied to the forward-backward algorithm for the initial amplitude is 20. Additive zero-mean Gaussian noise was applied to the observation model to obtain observed intensities of -1.92 and 2.91 on images 41 and 99 respectively. The results of the forward-backward algorithm for cycles 1 and 8 are shown in Figure 4.7.

The first cycle of the forward-backward algorithm looks promising as the estimate of the initial amplitude is slightly closer to the true value (Figure 4.7b). However, at cycle 8 (Figure 4.7d) it is clear that the forward-backward algorithm is converging to a point where every amplitude estimate is zero, which is not representative of the true amplitude. The log likelihood confirms that the estimates are getting worse as the values decrease with the cycle number (Figure 4.8).

To circumvent the problems caused by using the forward-backward algorithm on weak reflections, Bayesian inference is performed on the initial amplitude estimate at the end of each cycle. In particular the expected value of the posterior distribution is used as the initial amplitude estimate. The posterior distribution of interest is

$$P(F_c|F_0) = \frac{P(F_0|F_c) \times P(F_c)}{P(F_0)}, \tag{4.5.1}$$

where $P(F_0|F_c)$ is defined for centric and acentric reflections as

$$P_a(F_0|F_c) = \frac{2F_0}{\varepsilon\sigma_0^2} \exp\left(-\frac{F_0^2 + \boldsymbol{D}^2F_c^2}{\varepsilon\sigma_0^2}\right) I_0\left(\frac{2F_0\boldsymbol{D}F_c}{\varepsilon\sigma_0^2}\right), \tag{4.5.2}$$

$$P_c(F_0|F_c) = \left[\frac{2}{\pi\varepsilon\sigma_0^2}\right]^{1/2} \exp\left(-\frac{F_0^2 + \boldsymbol{D}^2F_c^2}{2\varepsilon\sigma_0^2}\right) \cosh\left(\frac{F_0\boldsymbol{D}F_c}{\varepsilon\sigma_0^2}\right), \tag{4.5.3}$$

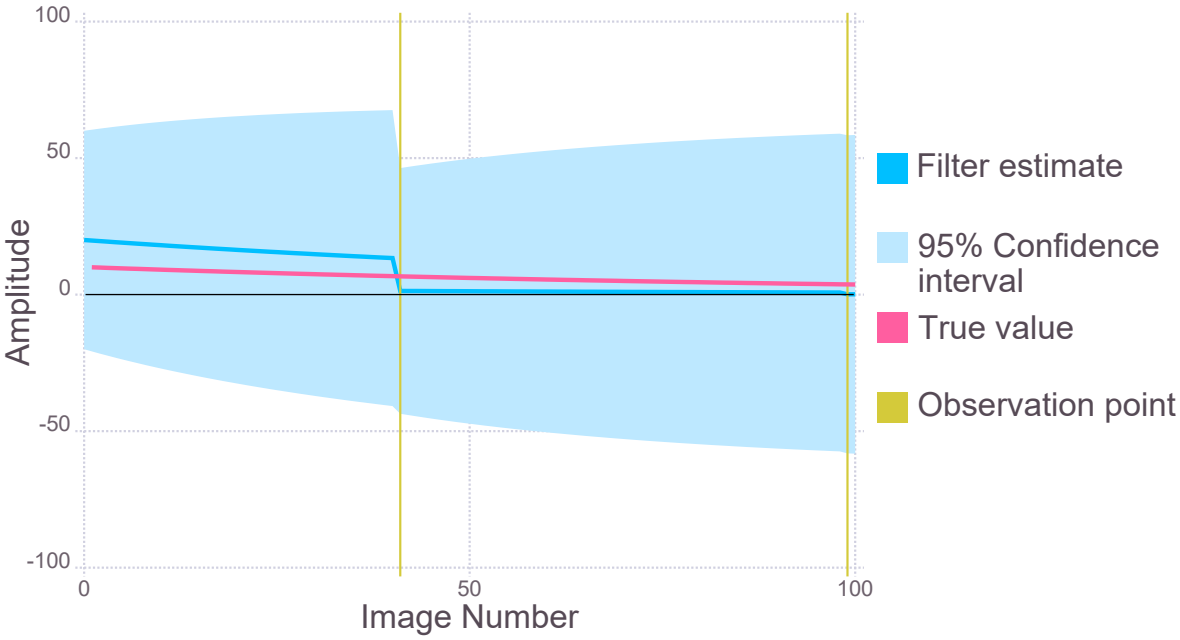
and $P(F_c)$ is also defined for centric and acentric reflections as

$$P_a(F_c) = \frac{2F_c}{\Sigma^2} \exp\left(-\frac{F_c^2}{\Sigma^2}\right), \tag{4.5.4}$$

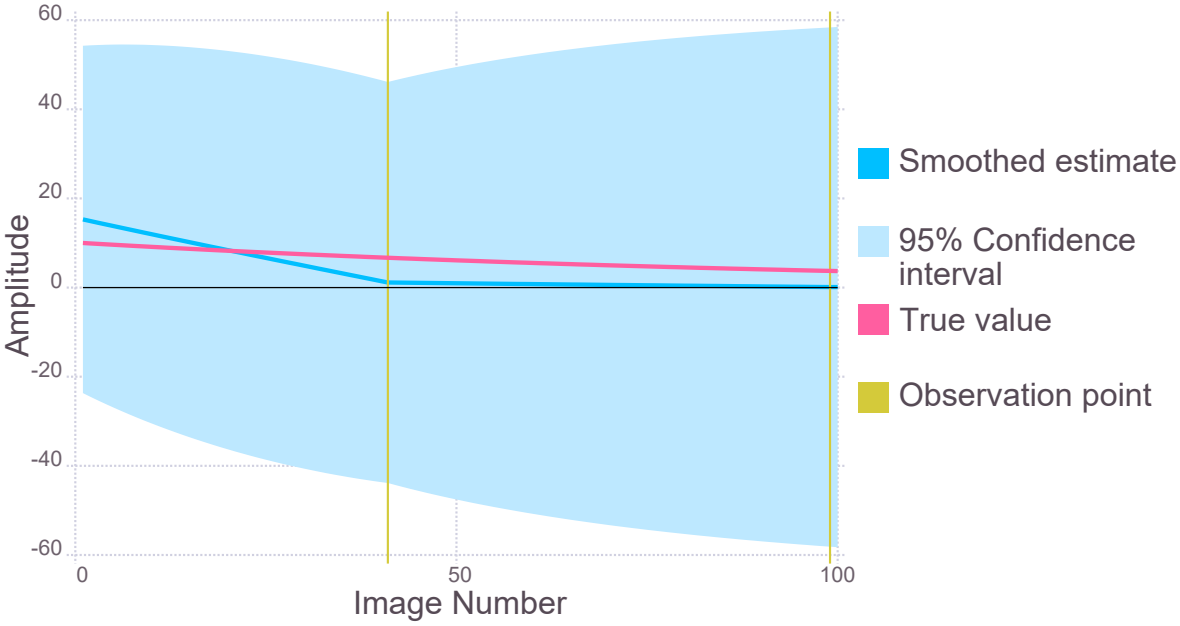
$$P_c(F_c) = \sqrt{\frac{2}{\pi\Sigma^2}} \exp\left(-\frac{F_c^2}{2\Sigma^2}\right), \tag{4.5.5}$$

where $P(F_c)$ is the Wilson distribution for amplitudes, $F_0$ and $\sigma_0^2$ are the initial amplitude and variance estimates from the forward-backward cycle, and $F_c$ is the amplitude to be calculated. As discussed in section **??**, the denominator of equation 4.5.1 can be given as:

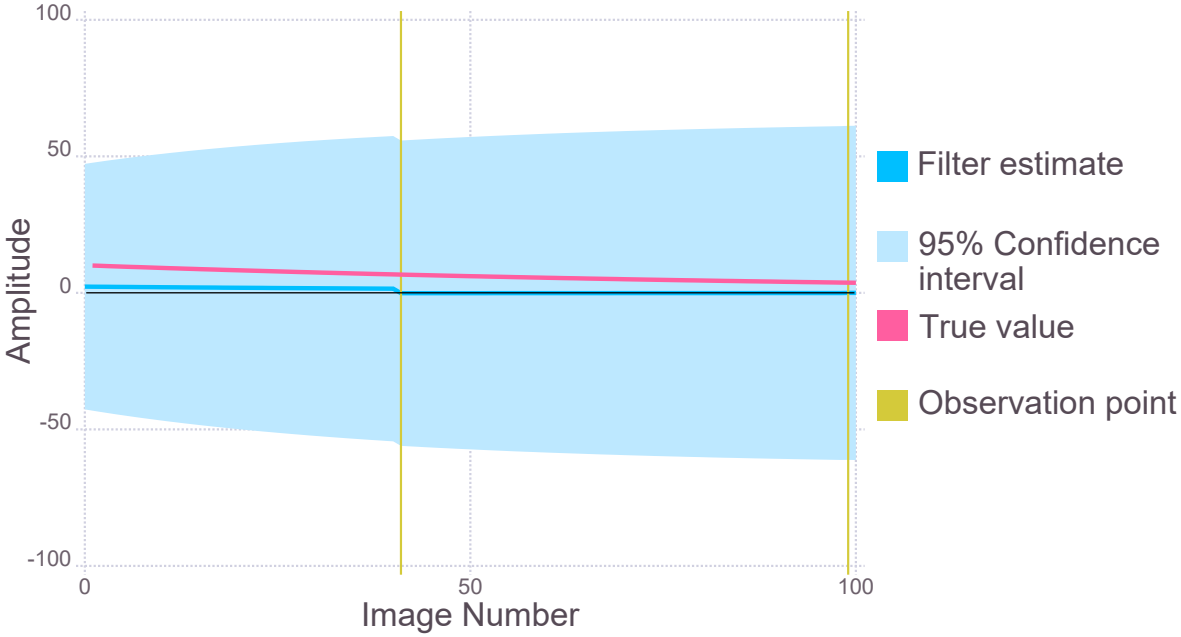
$$P(F_0) = \int_0^\infty P(F_0|F_c) \times P(F_c)\,\mathrm{d}F_c. \tag{4.5.6}$$



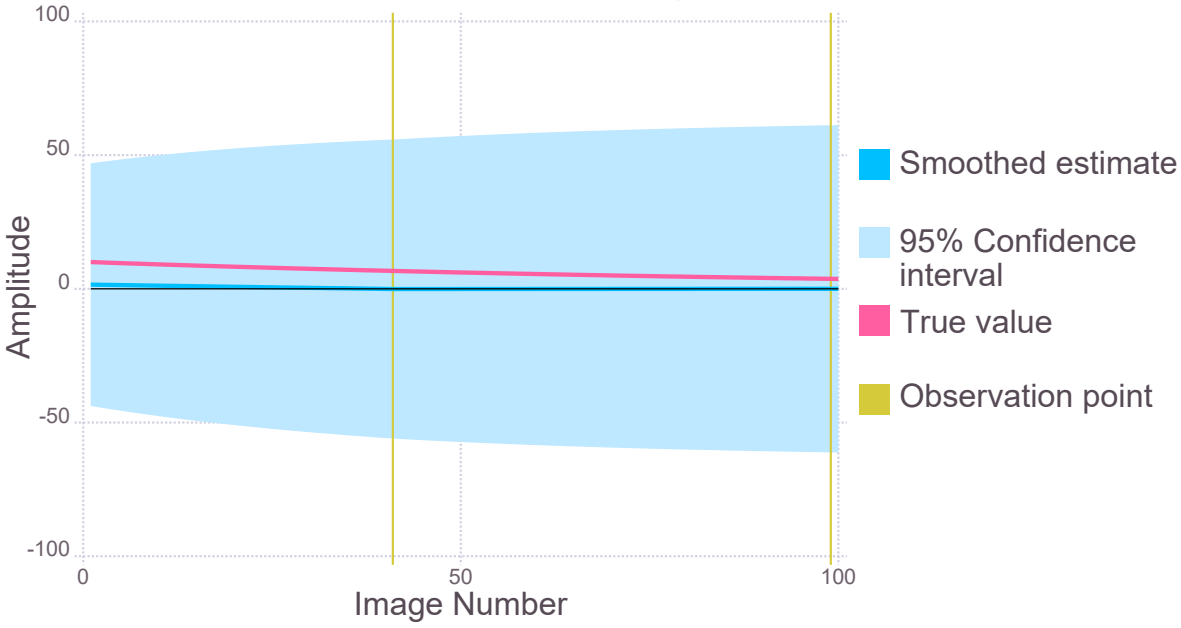

(a) Cycle 1: forward pass.




(b) Cycle 1: backward pass.




(c) Cycle 8: forward pass.




(d) Cycle 8: backward pass.

Figure 4.7: Forward-backward algorithm results for a simulated weak reflection observed on image 41 and 99 (solid gold lines) of a dataset consisting of 100 images. The forward-backward pass for cycle 1 looks like it may lead to a good estimate of the true value. However, by cycle 8 the estimates have converged to zero and the 95% confidence interval has not improved.

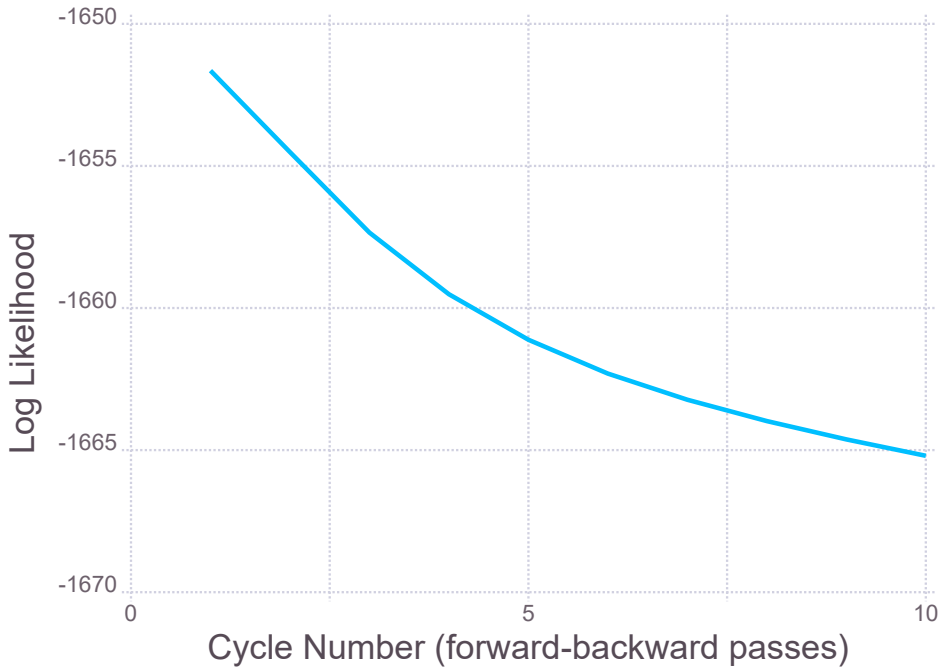



Figure 4.8: Log likelihood values calculated for each smoothing pass. Visually the forward-backward passes seemed to give worse estimates as the number of cycles increased (Figure 4.7). The log likelihood values confirm this as evidenced by the decrease in the values.

This procedure ensures that the initial amplitude value for the next forwards pass is positive. If the final value for the amplitude at the end of a forwards pass is negative, then that value is set to zero for the smoother. Systematic testing is required to determine whether this is a robust solution to the problem

## 4.6 Protein structure results

### 4.6.1 Bovine pancreatic insulin

#### Scale and B factors

Data were collected on a crystal of bovine pancreatic insulin (crystal ID 0259) as described in chapter **??**. The atomic composition used to provide expected intensity values, was obtained from the insulin structure with PDB code 2BN3. The B factors could then be calculated for each image according to equation 4.3.18 and are shown in Figure 4.9. It can be seen that there are a couple of points that may be regarded as outliers in Figure 4.9a. To remove the outliers, the mean and standard deviation of the B factors were calculated and any B factor that was more than two standard deviations from the mean was removed. The resulting B factor distribution is plotted in Figure 4.9b.

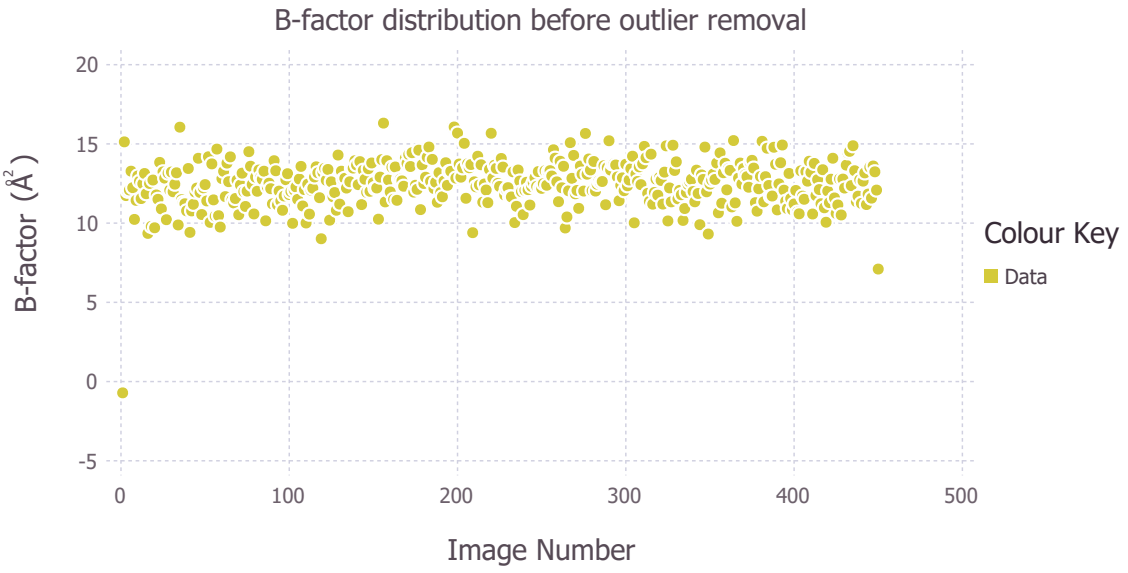
To ensure that the B-factors exhibited linear behaviour, "damage corrected" B factors, $B_{dc}$, were calculated by rearranging the linear formula for the B factor increase.

$$B_{dc}^{i} = B_i - \Delta B \times i, \tag{4.6.1}$$

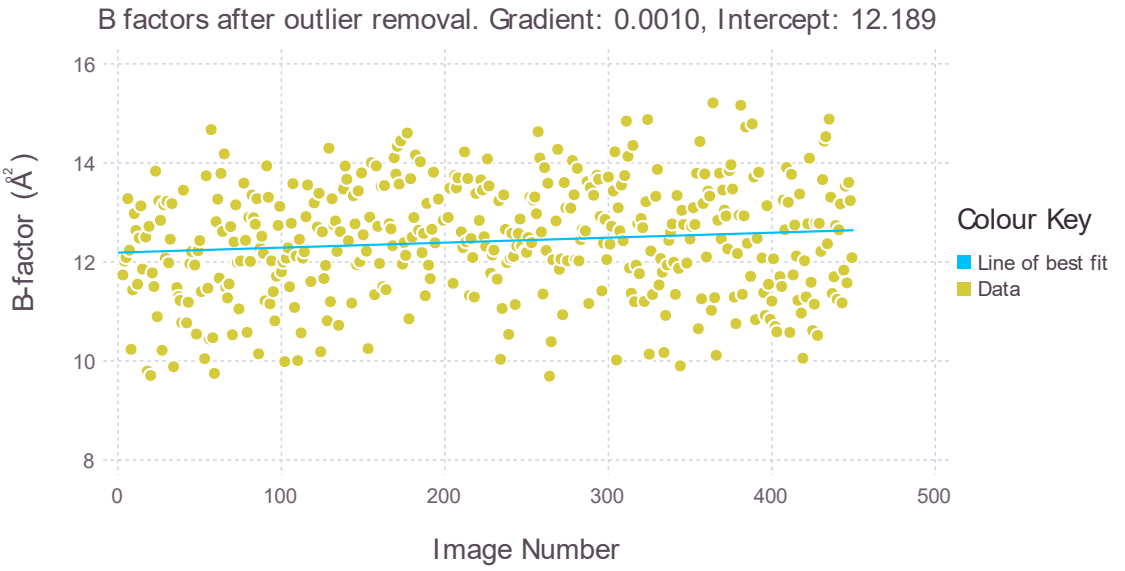
where $B_i$ is the B factor calculated at image $i$, and $\Delta B$ is the gradient of the line fitted to the data. A histogram of the "damage corrected" B factor distribution was then plotted, which should be a Gaussian distribution centred on the intercept of the line in Figure 4.9b. Additionally a QQ plot[†] was used to ensure that the data were normally distributed (Figure 4.10b). The gradient $\Delta B$ was calculated to be 0.001 Å$^2$.

The set of scale factors calculated from the images, $\{s_{images}\}$, are shown in Figure 4.11a. There did not appear to be any outliers from visual inspection, so there was no outlier rejection method performed for the scale factors. The scale factors that corresponded to images that were removed from the B factor outlier rejection analysis were also omitted for consistency. The resulting scale factors, $\{s^{*}_{images}\}$, are shown in Figure 4.11b.

[†]A QQ plot is graphically determines whether two datasets come from the same distribution. It plots the quantiles of the first dataset against the quantiles of the second. If the two datasets come from the same distribution the points should fall on a 45° reference line which is typically also plotted.

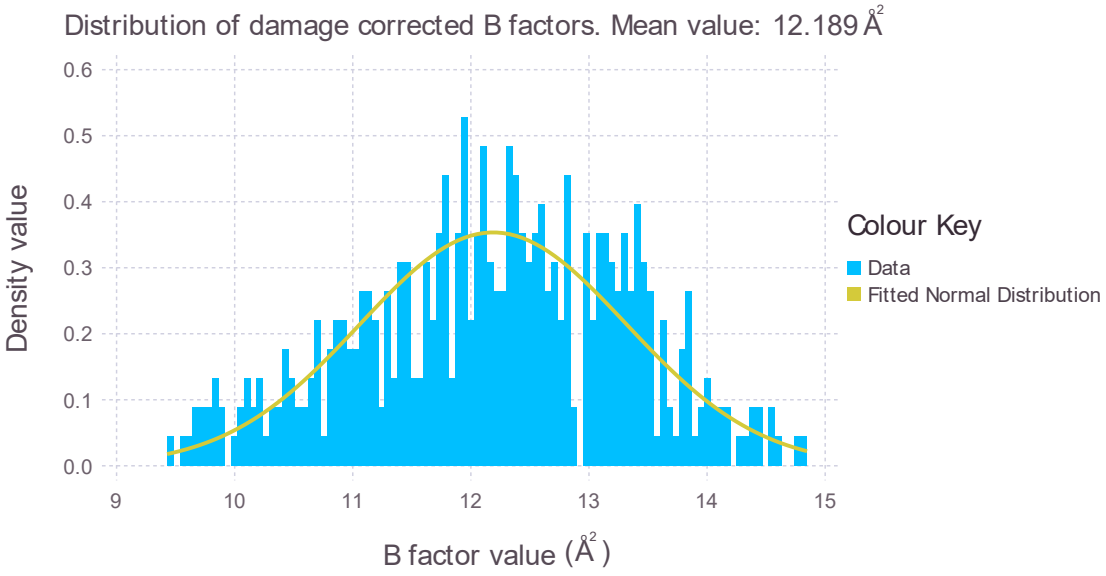



(a)

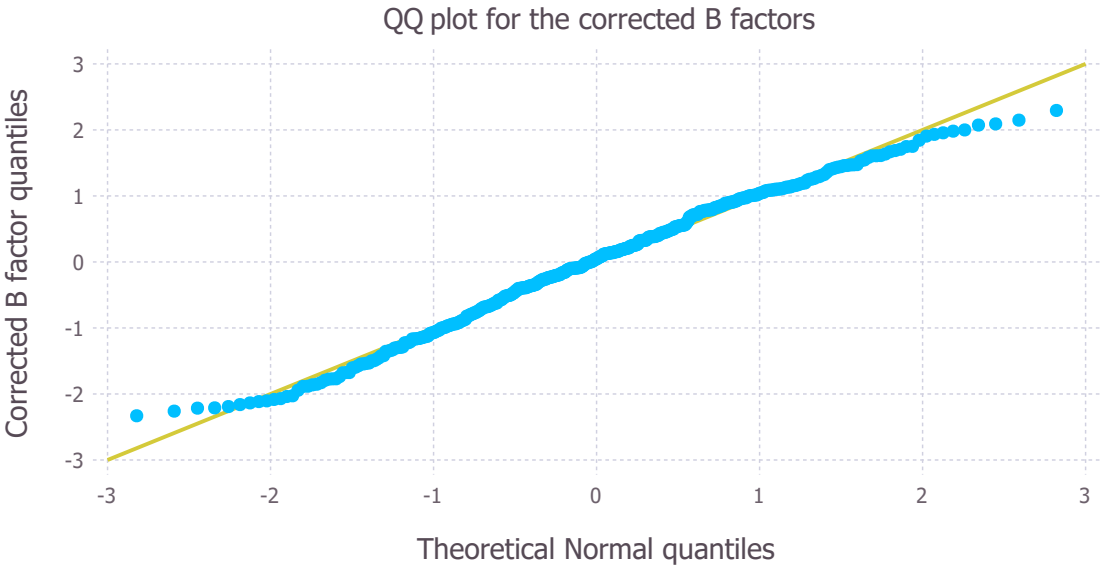



(b)

Figure 4.9: Calculated B factors for each image in the insulin dataset. (a) Distribution before outlier removal. (b) Distribution after outlier removal. The line of best fit (blue solid line) with gradient, $\Delta B$ = 0.001 Å$^2$ and intercept = 12.189 Å$^2$, is overlaid on the data. Note the differing $y$ axis scales.

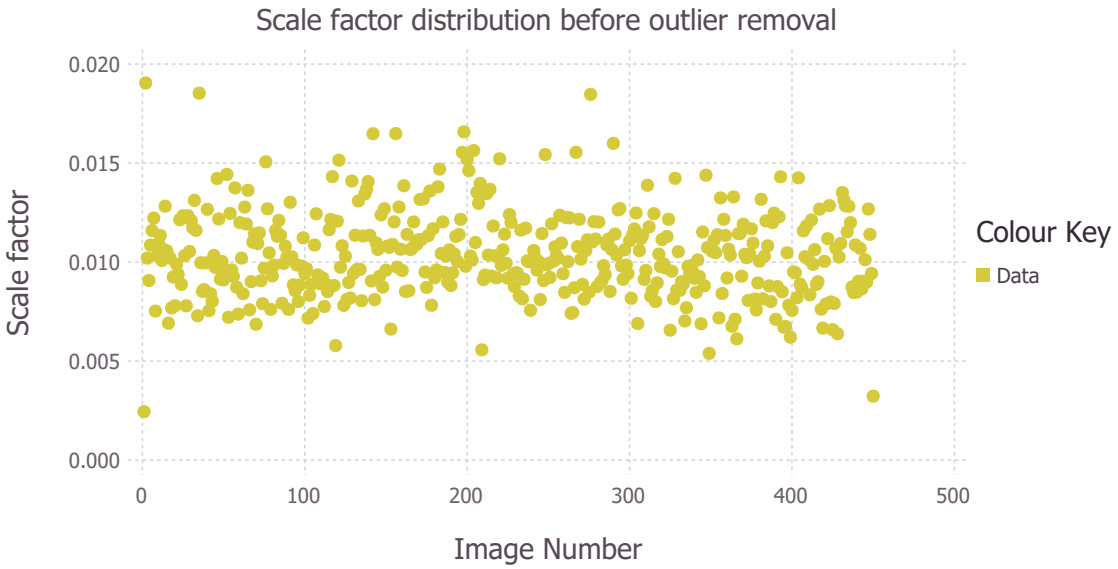



(a)

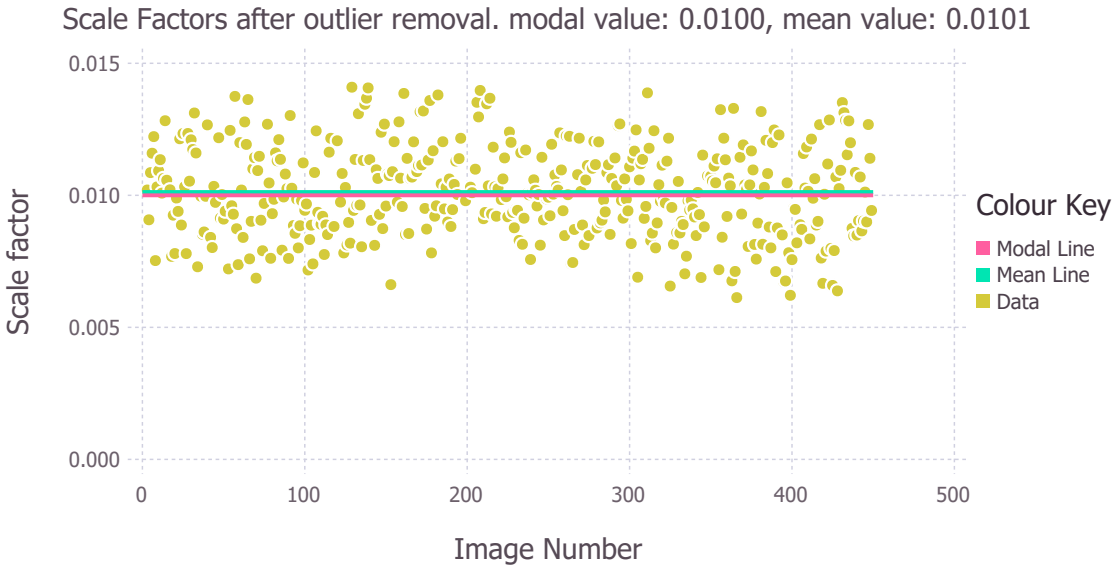



(b)

Figure 4.10: (a) Histogram of damage corrected B factors. The Gaussian shape suggests that a linear assumption for the behaviour of B factors is suitable. (b) QQ plot for the damage corrected B factors. The linearity of the points confirm that the distribution is actually Gaussian.




(a)




(b)

Figure 4.11: Calculated scale factors for each image in the insulin dataset. (a) Distribution before outlier removal. (b) Distribution after outlier removal. The solid green and solid pink lines represent the mean and mode of the distribution respectively. The fact that the mean and mode are close in value suggest that the distribution is Gaussian.

Figure 4.12 presents the distribution of scale factors as a histogram. The fact that the mode and mean are very similar in value also suggests that the scale factor distribution is Gaussian. However there is no guarantee that the scale factor distribution for a general experiment will be Gaussian and hence there are no checks for normality included in the algorithm for the scale factors. The mean scale factor value of the distribution, $s_{mean}$ = 0.010 was used in the forward-backward algorithm, which is assumed to be constant throughout the experiment. This assumption is not true in the general case, but it should be suitable for the diffraction experiment concerned, for which the insulin crystal was completely immersed in a tophat beam for a 45° rotation. This is because the transmission through the crystal and the illuminated volume should remain constant.

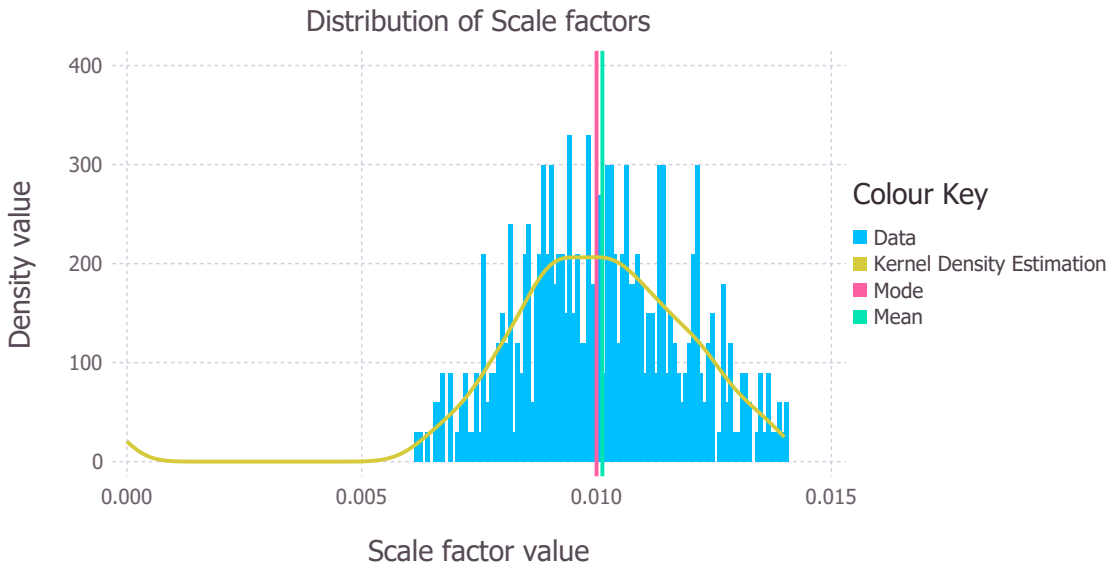



Figure 4.12: Histogram of scale factors with the mean (solid green line), mode (solid pink line) and kernel density estimation (solid gold line) overlaid.

### Forward-backward algorithm

The forward-backward algorithm was applied with the following parameters defined:

- the minimum and maximum number of forward-backward cycles for an individual reflection was 5 and 200 respectively.
- the forward-backward cycles were regarded to have converged if:
    1. the maximum number of cycles had been reached (200)

2. the absolute change in log likelihood between consecutive cycles was less than 0.1
3. the change in initial amplitude value between consecutive cycles was less than 1

- reflections for which the Rician distribution was used for the amplitude process function (equations 4.3.8 and 4.3.9) were defined as reflections where $F_0/\sigma(F_0) < 3$, where $F_0$ is the initial amplitude estimate.

Amplitude estimates resulting from applying the forward-backward algorithm for 4 reflections are shown in Figure 4.13. It can be seen that for these reflections, the initial amplitude value estimated using CTRUNCATE is within the 95% confidence region as estimated by the forward-backward algorithm. This suggests that the constant scale factor assumption used for the simple scaling procedure used for this study is valid for this experiment.

Figures 4.13a and 4.13b exhibit the expected behaviour of an average refection, since both of these reflections decay relatively smoothly as exposure time increases. Figure 4.13c shows a reflection whose amplitude increases as the exposure increases. This demonstrates that the forward-backward algorithm is capable of capturing the different behaviours of various reflections despite the process function describing a monotonic decay of the amplitudes.

Figure 4.13d shows a reflection where the behaviour is very irregular and not very smooth. In this case it is very likely not to be caused by a physical phenomenon and is probably due to incorrect scaling of the data. To prevent this problem, restraints should be imposed during the scaling procedure in a similar manner to those of existing scaling methods to ensure sufficient smoothness of reflection amplitudes (**??**). It should be noted that not all sharp changes in amplitudes relate to noise/incorrect processing. Mechanical or chemical changes of the structure can occur within a few seconds (**?**).

Another undesirable feature of the forward-backward algorithm is the fact that the initial amplitude estimate is solely influenced by the amplitude estimate at the point where the observation is made. This effect is prominent in Figures 4.13b and 4.13c. Thus the initial amplitude estimate is not influenced by the multiplicity and hence erroneous estimates are more likely to arise if the first observation of a reflection is an outlier. This can be overcome by performing the forward-backward algorithm on each observation separately and merging the entire amplitude curves for equivalent reflections. The relative uncertainties at each

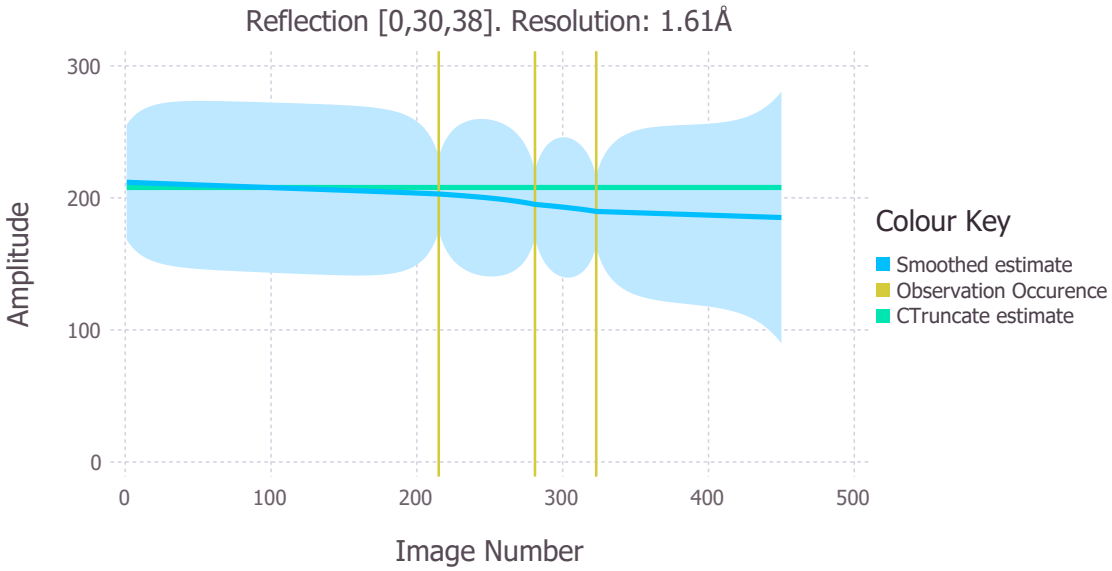
point in the data collection experiment will be different because the observations will be collected on different images. This should ensure that the (possibly non-linear) behaviour of the reflection should still be captured by this method.

**Refinement results**

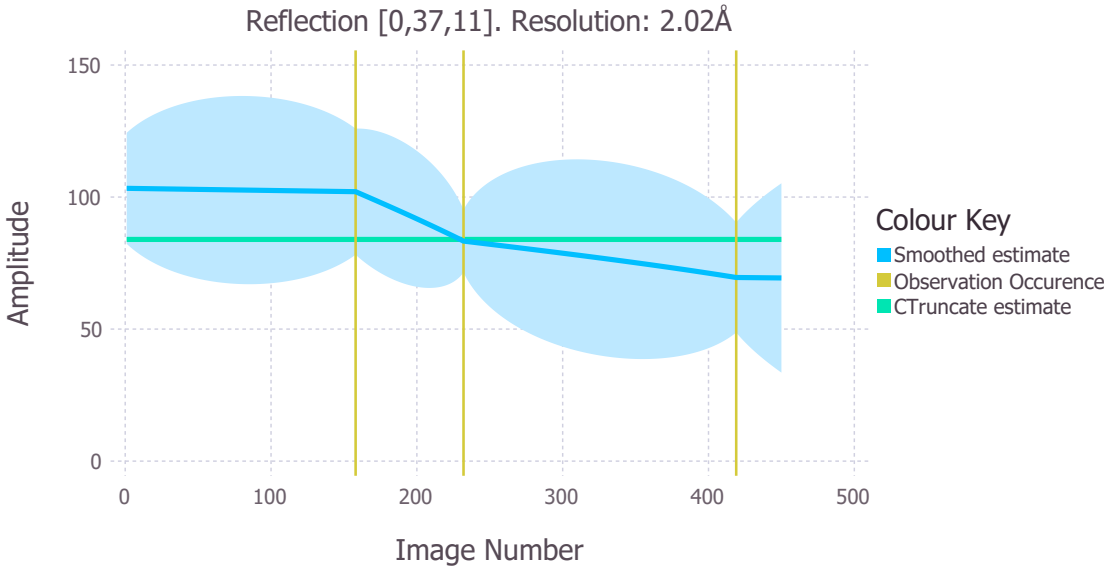
The initial amplitude estimates resulting from the forward-backward algorithm (FBA) were combined with the phases from a deposited insulin structure (PDB code 2BN3) and refined with REFMAC (**?**) (10 cycles of rigid body refinement followed by 10 cycles of restrained refinement). The same refinement procedure was performed with data processed using AIMLESS (**?**) and CTRUNCATE (ACT pipeline)(**?**) (i.e. no processing with the forward-backward algorithm). The resulting electron density maps contoured at the $3\sigma$ at 1.38 Å for selected residues are shown in Figure 4.14. The maps are practically identical for the two different data reduction pipelines, which is the case for the rest of the structure.

A difference map was also calculated to locate the major differences between the results of the two different data reduction pipelines. The differences were calculated between the resulting amplitudes from the ACT and FBA pipelines, rather than the calculated amplitudes after refinement. Phases were obtained from the final model resulting from refinement using the phases from PDB 2BN3 and the ACT amplitudes. Figure 4.15 displays the resulting difference map contoured at the $3\sigma$ level along with the full insulin structure from which the phases were obtained. The difference electron density is distributed quite uniformly over the unit cell rather than showing large differences overlapping the structure. This supports the result shown above that the two methods give practically identical electron density for the structure.

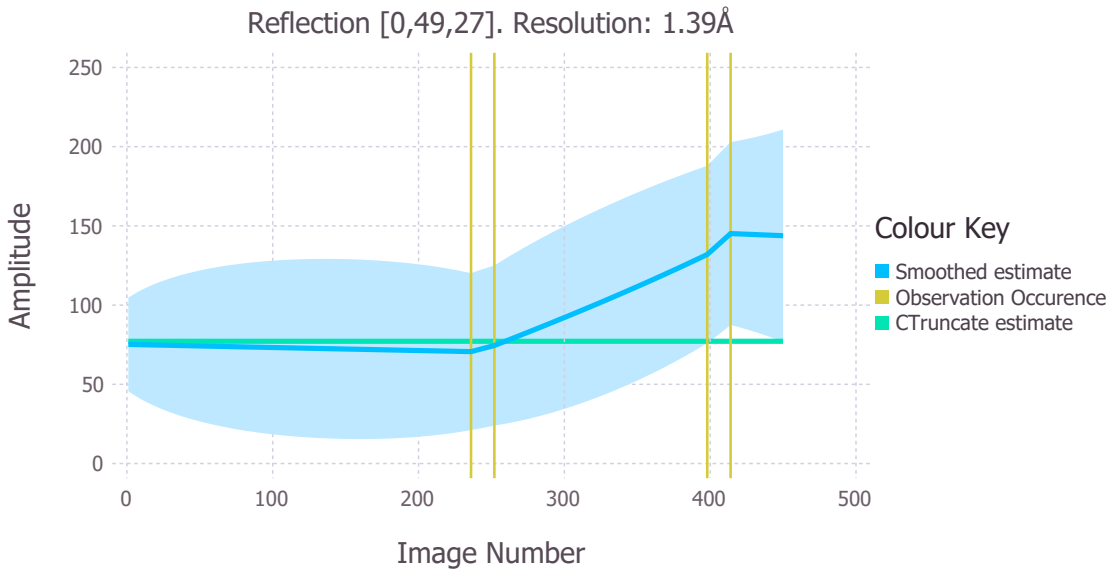
Refinement statistics for both pipelines are shown in Table 4.1. Overall the statistics are quite similar but they are consistently better for the ACT pipeline. It is likely that the FBA results can be improved by merging the amplitude estimates for symmetry related reflections after the algorithm has been applied to each observation individually, as described in section 4.6.1. Another difference between the two pipelines is that the FBA pipeline does not include any outlier rejection whereas AIMLESS does. This is likely to be the reason why there were more reflections at the end of data reduction using the FBA pipeline (16261) compared to that when using the ACT pipeline (16233).


Reflection [0,30,38]. Resolution: 1.61Å
Amplitude
Image Number
0
100
200
300
400
500
Colour Key
Smoothed estimate
Observation Occurence
CTruncate estimate

(a)


Reflection [0,37,11]. Resolution: 2.02Å
Amplitude
Image Number
0
50
100
150
200
300
400
500
Colour Key
Smoothed estimate
Observation Occurence
CTruncate estimate

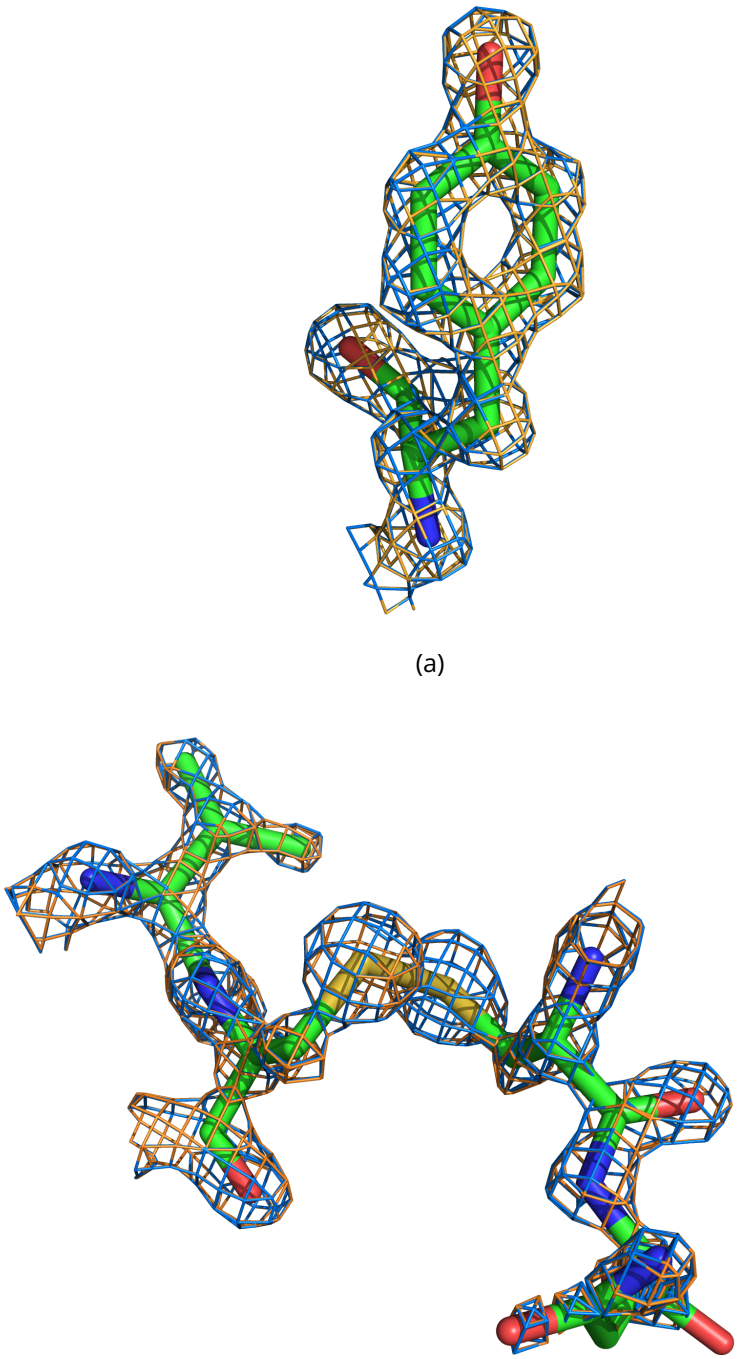
(b)




(c)


(d)

Figure 4.13: Amplitude estimates for four different reflections observed in the insulin dataset using the forward-backward algorithm (blue solid line). The estimate produced by CTRUNCATE is shown in green. The estimates all agree within the 95% confidence interval (light blue shaded region) determined by the forward-backward algorithm. (a), (b) and (c) exhibit somewhat smooth changes in the amplitude behaviour, whereas (d) shows very sharp and irregular changes, which are likely to be unphysical.



(a)

(b)

Figure 4.14: $2F_o$ - $F_c$ electron density maps contoured at the $3\sigma$ level at 1.38 Å resolution for the ACT pipeline (blue) and the FBA pipeline (orange) with the insulin structure obtained after refinement with REFMAC with data processed via the ACT pipeline. (a) Tyrosine residue. (b) Disulphide bond. The electron density maps are practically identical and these are representative of the density around the entire structure.

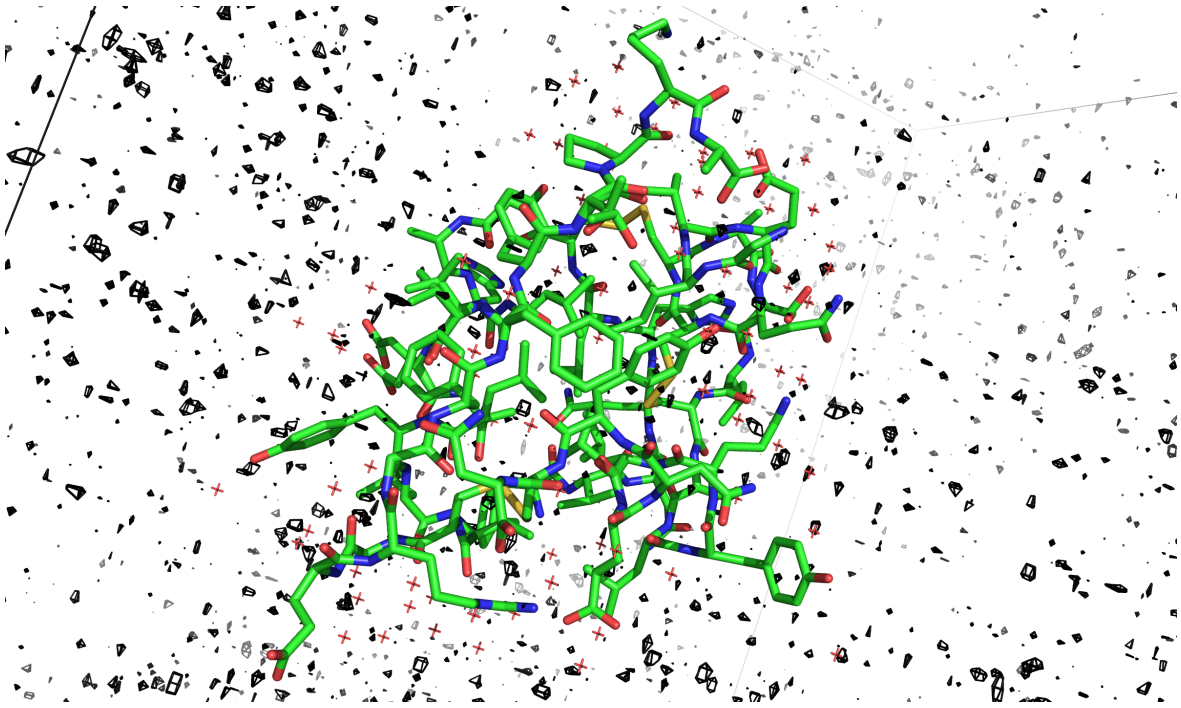


Figure 4.15: Difference electron density map (black mesh) contoured at the $3\sigma$ level between the amplitudes resulting from the ACT pipeline and the FBA pipeline using the phases obtained from the model resulting from refinement with the data processed using the ACT pipeline. The insulin structure from which the phases were obtained is also shown as a green stick model. The difference density is uniformly distributed throughout the unit cell and no large differences can be seen overlapping the structure. This suggests that the two methods would result in the same structure as evidenced by the electron density maps in Figure 4.14.

Table 4.1: Final refinement statistics for data processed with the ACT and FBA pipelines

| | ACT | FBA |
|---|---|---|
| R work | 0.165 | 0.171 |
| R free | 0.177 | 0.182 |
| RMS bond length (Å) | 0.029 | 0.030 |
| RMS Bond Angle (°) | 2.493 | 2.552 |

### 4.6.2 Protein-DNA complex - C.Esp1396I

#### Scale and B factors

Data were collected from a crystal of the bacterial protein-DNA complex (C.Esp1396I) as described in Bury *et al.* (2015) (Figure 4.16). Notably the crystal (30 $\mu m$ × 30 $\mu m$ × 10 $\mu m$) was exposed to a 25 $\mu m$ circular Gaussian profile beam (FWHM dimensions before the 25 $\mu m$ diameter pinhole are 0.212 $mm$ × 0.279 $mm$), with the crystal oriented such that the 10 $\mu m$ dimension was aligned parallel to the beam direction. A single dataset consisted of 100 frames with each frame generated from a 1° rotation (100° total wedge). Thus not all of the crystal was immersed in the beam and the rotation led to the X-ray beam path length and illuminated volume through the crystal changing throughout the experiment. The atomic compo-

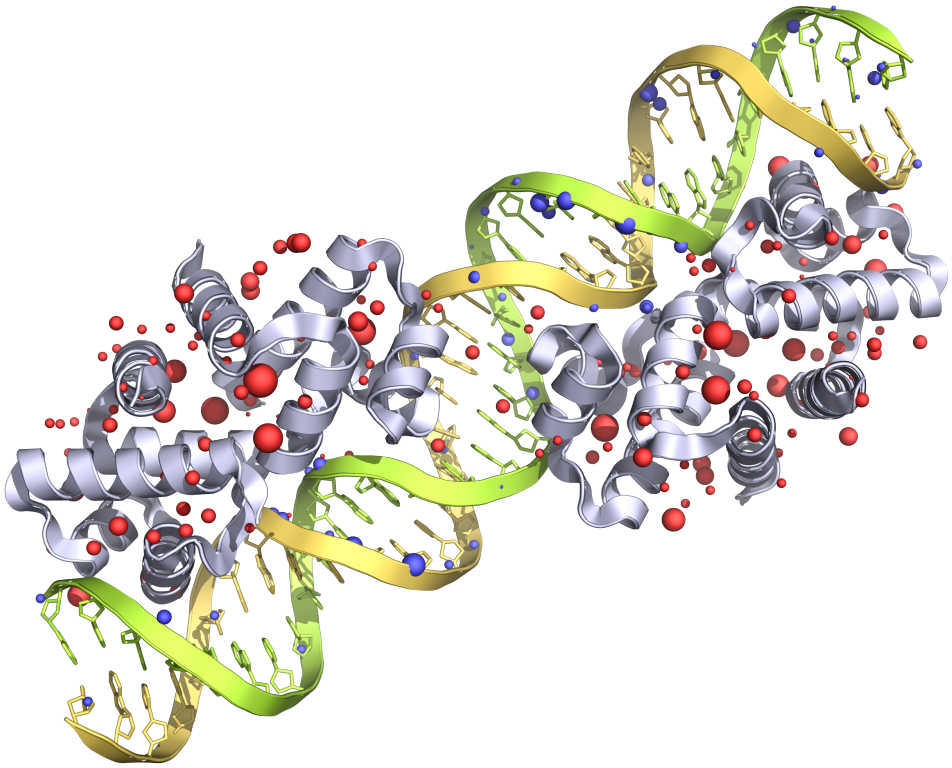
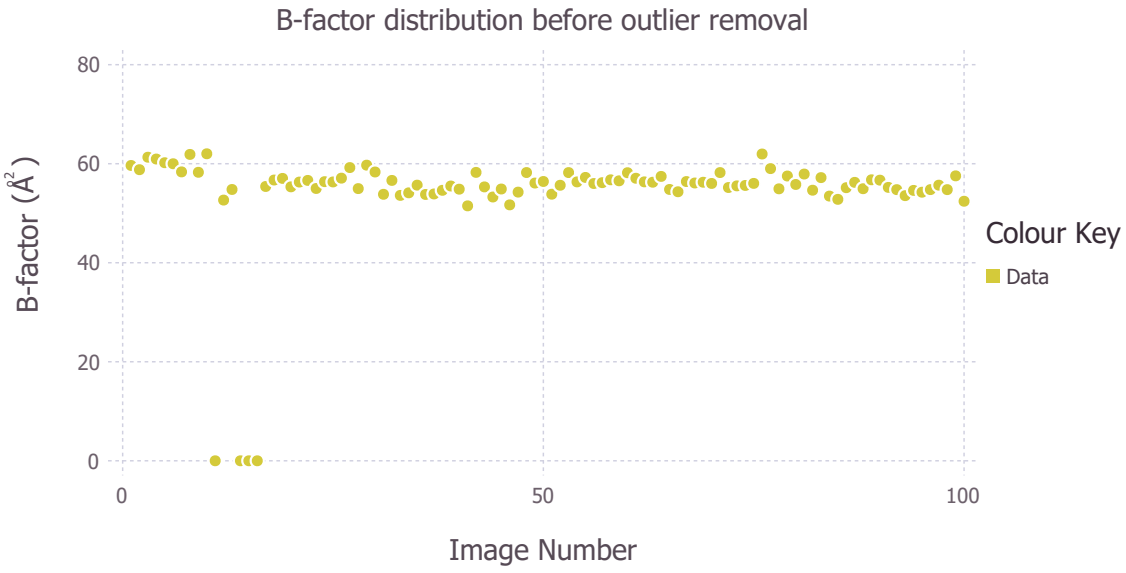


Figure 4.16: Structure of the C.Esp1396I protein-DNA complex. The spheres show sites of specific radiation damage at a dose of 44.6 MGy. The radii of the spheres are proportional to the electron density loss and the spheres closer/further than 2 Å from the DNA strands are coloured blue/red (**?**).

sition used to provide expected intensity values was obtained from the PDB structure 4X4B. The B factors were calculated as described above and are shown in Figure 4.17. Four reflections that were removed in the outlier rejection procedure are clearly visible in Figure 4.17a with B factor values of zero. The initial B factor is much higher for the C.Esp1396I structure (57.69 Å$^2$) than it was for the insulin structure (12.19 Å$^2$). In contrast to the behaviour observed with the insulin dataset, the B factor for the C.Esp1396I structure decreases linearly throughout the dataset (Figure 4.17b). The assumption of a linear behaviour of the B-factor is again justified (Figure 4.18).

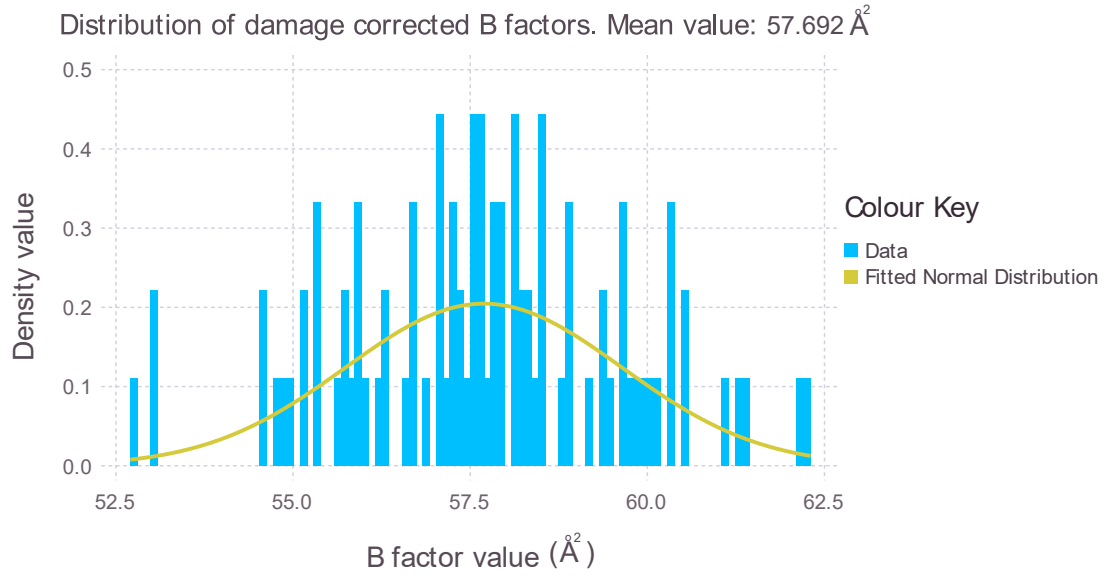
The calculated scale factor distribution, $\{s_{images}\}$, is shown in Figure 4.19. As with the insulin dataset, no specific outlier rejection was carried out on the scale factors, so the only ones rejected were those that corresponded to the same images on which the rejected B factors were found, i.e. the four scale factors with a value of 1 in Figure 4.19a. The resulting scale factors, $\{s^*_{images}\}$, calculated from the images (Figure 4.19b) do not exhibit a constant behaviour. This implies that the assumption of a constant scale factor (as was assumed to be the case for insulin) is likely to give incorrect results. The kernel density estimate in



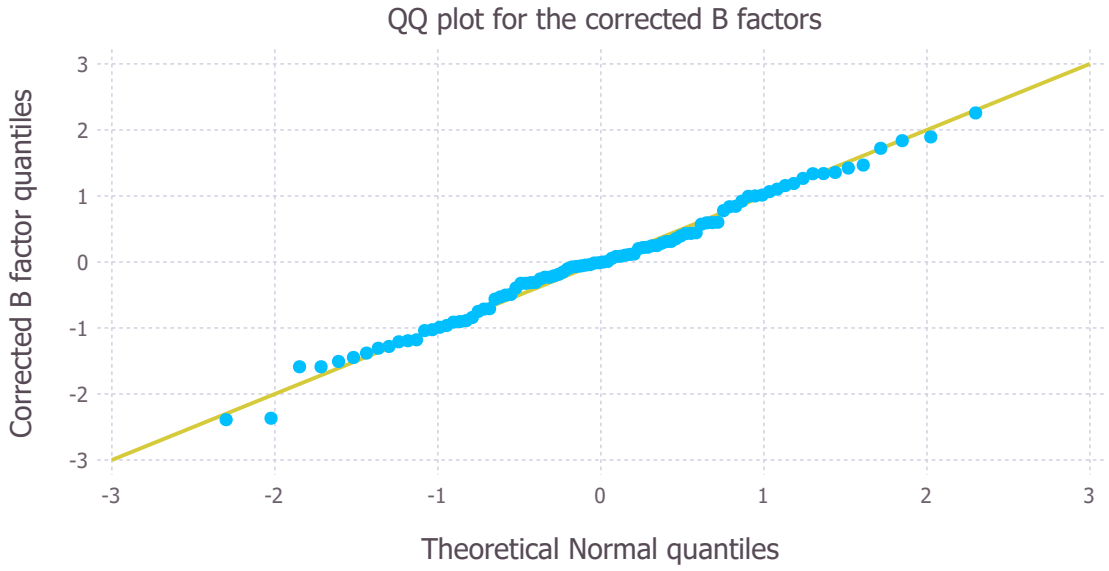

(a)


(b)

Figure 4.17: Calculated B factors for each image in the C.Esp1396I dataset. (a) Distribution before outlier removal. (b) Distribution after outlier removal. The line of best fit (blue solid line) with gradient, $\Delta B$ = -0.0296 Å$^2$ and intercept = 57.592 Å$^2$, is overlaid on the data.




(a)




(b)

Figure 4.18: (a) Histogram of damage corrected B factors for the C.Esp1396I structure. (b) QQ plot for the damage corrected B factors. The linearity of the points in the QQ plot supports the Gaussian approximation in the histogram, suggesting that the B factors change linearly.

Figure 4.20 shows that the scale factor distribution is bimodal and hence a single (constant) value does not represent the distribution adequately. A varying scale factor has not yet been implemented into the forward-backward algorithm and hence the mean value of the calculated scale factor distribution, $s_{mean}$ = 0.0147, was used as the scale factor in the processing.

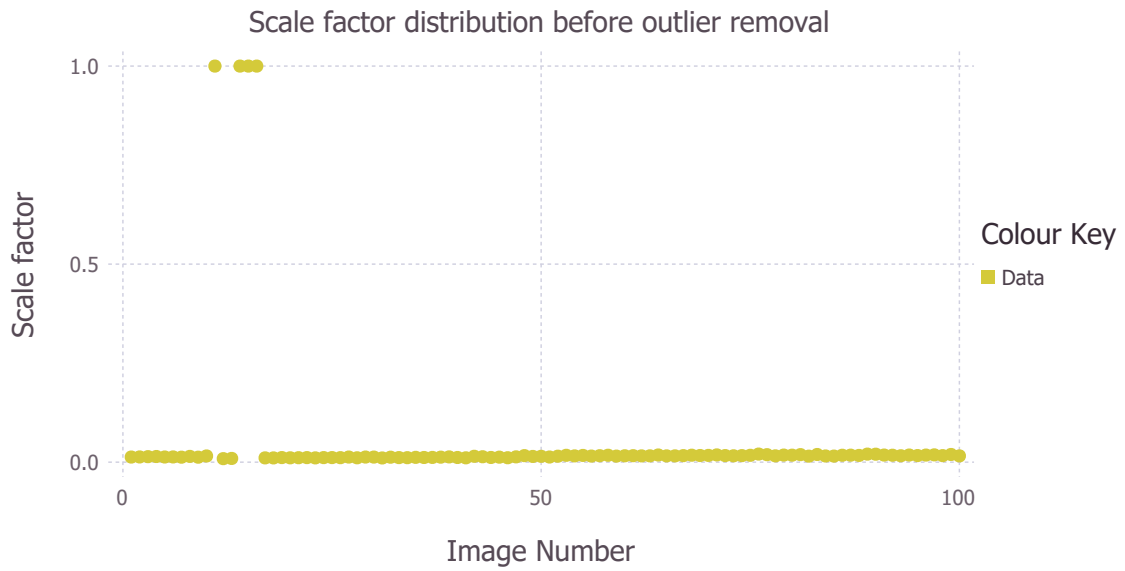
**Forward-backward algorithm**

The forward-backward algorithm was carried out with the same parameters as defined for the insulin structure. The only difference was that a Gaussian approximation was used as the process function for every reflection. Furthermore the Bayesian inference method described in section 4.5.1 was not performed, because the algorithm suffered numerical issues when calculating the Rician approximation. This is likely to be due to the calculation of the modified Bessel function of the first kind of order zero for arguments with high values. Mathematically this function increases in an exponential manner, especially for large argument values, and ultimately reaches a point where an error is flagged. The Gaussian approximation does not rely on evaluating this function which is why it was used instead for all reflections. Addressing this issue will require the algorithm to check the size of the argument before evaluating it.

The amplitude estimates for two reflections resulting from the processing are shown in Figure 4.21. As was the case with the insulin dataset, some reflections exhibit smooth behaviour (Figure 4.21a), whereas others show sharp changes that are likely to be due to imperfect scale factors (Figure 4.21b). Due to the incorrect scale factor used for the images in the dataset, the CTRUNCATE and the FBA amplitude values do not agree.

**Refinement results**

The initial amplitude estimates resulting from FBA were combined with the phases from the deposited C.Esp1396I structure (PDB code 3CLC) and refined with REFMAC (**?**) (20 cycles of rigid body refinement followed by 20 cycles of restrained refinement). The same procedure was also performed using the amplitudes calculated using the ACT pipeline. The resulting electron density maps contoured at the 3$\sigma$ at 2.4 Å for selected residues are shown in




(a)


(b)

Figure 4.19: Calculated scale factors for each image in the C.Esp1396I dataset. (a) Distribution before outlier removal. (b) Distribution after outlier removal. The solid green and solid pink lines represent the mean and mode of the distribution respectively. The scale factor is clearly not constant and shows an increasing trend throughout the experiment. This leads to differences in the mean and modal values of the distribution.

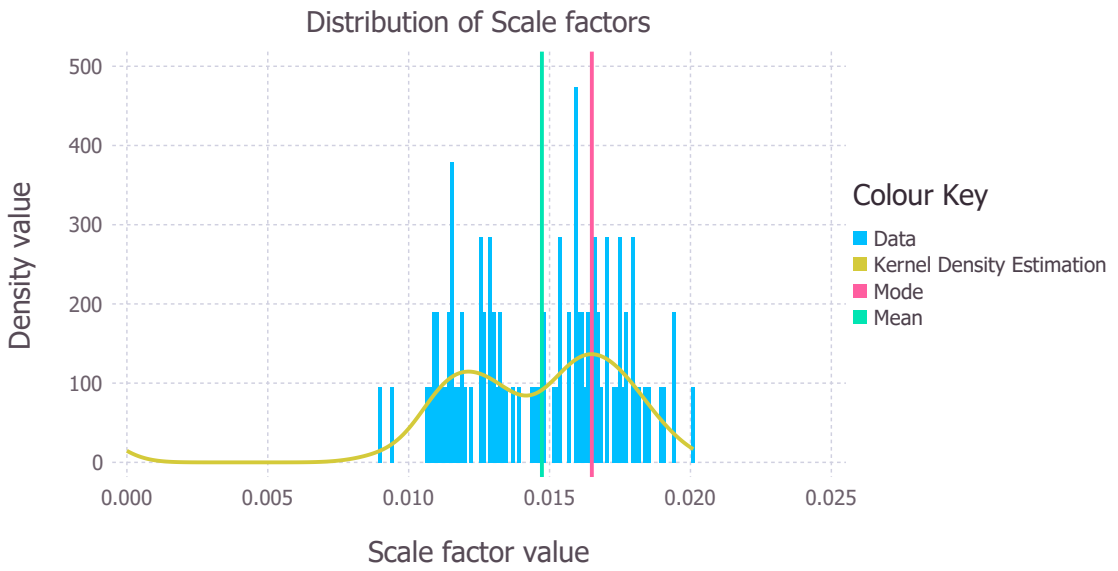


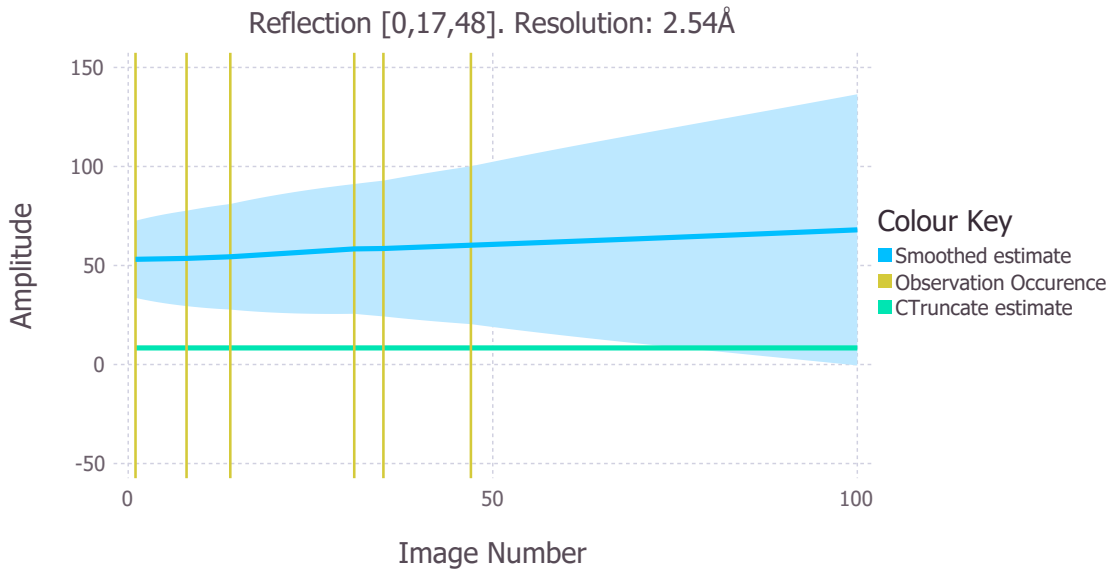

Figure 4.20: Histogram of scale factors with the mean (solid green line), mode (solid pink line) and kernel density estimation (solid gold line) overlaid. The bimodal distribution of the scale factor is clear from the kernel density estimate.

Figure 4.22. Once again the maps generally agree structurally but the overlap is less pronounced in this structure than it was for insulin. This is expected because the amplitude values did not agree as well, largely due to an incorrect scale factor used for the FBA.

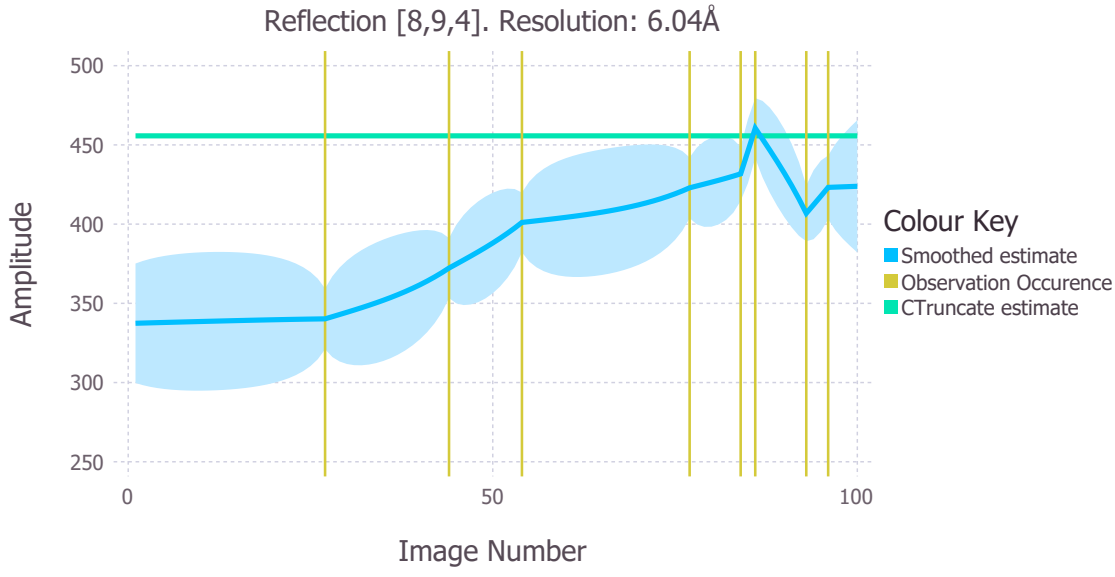
The difference map between the amplitude values calculated from the ACT and FBA pipelines, using phases from the final model after refinement with the 3CLC phases and ACT amplitudes, is shown in Figure 4.23. The difference density is no longer random and in fact is located around where the model is located in the unit cell. This suggests that the two different pipelines could lead to different structural models.

Again the total number of reflections differ between datasets. The FBA pipeline results in 32944 reflections at the end of data reduction, whereas the ACT pipeline results in 32898 reflections.

Overall refinement statistics using both pipelines are shown in Table 4.2. The R values are slightly better for the ACT pipeline but the RMS values are lower for the FBA pipeline. The statistics obtained using the FBA method are much better than expected considering the scale factor used in the FBA was non-optimal. If the amplitude values resulting from the FBA pipeline are indeed inaccurate then it is likely that the refinement process is significantly “mopping up" the errors that are made during the data reduction stage. It is also possible

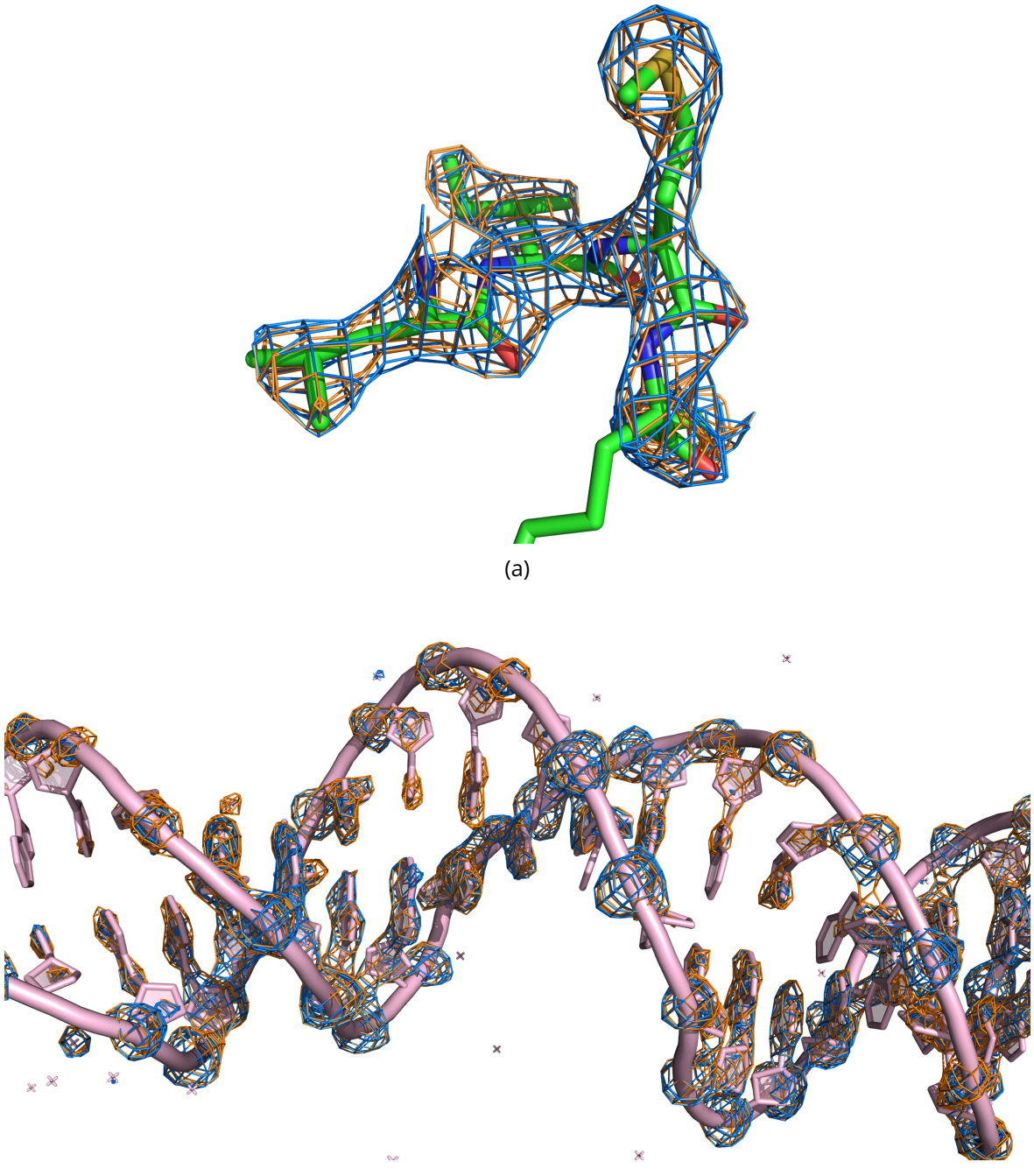



(a)




(b)

Figure 4.21: Amplitude estimates for two different reflections observed in the C.Esp1396I dataset using the forward-backward algorithm (blue solid line). The estimate produced with CTRUNCATE is shown in green. The estimates using the two different pipelines do not agree and this is likely to be due to the incorrect scale factor used for the forward-backward algorithm. Reflection 8,9,4 in (b) also exhibits sharp changes in the amplitude, which are likely to be noise.



(a)

(b)

Figure 4.22: $2F_o$ - $F_c$ electron density maps contoured at the $3\sigma$ level at 2.42 Å for the ACT pipeline (blue) and the FBA pipeline (orange). The model was obtained after refinement with REFMAC with data processed via the ACT pipeline. (a) Leu-Ile-Met-Lys-Gly residues of the protein from the C.Esp1396I protein-DNA complex. (b) DNA section of the C.Esp1396I protein-DNA complex.

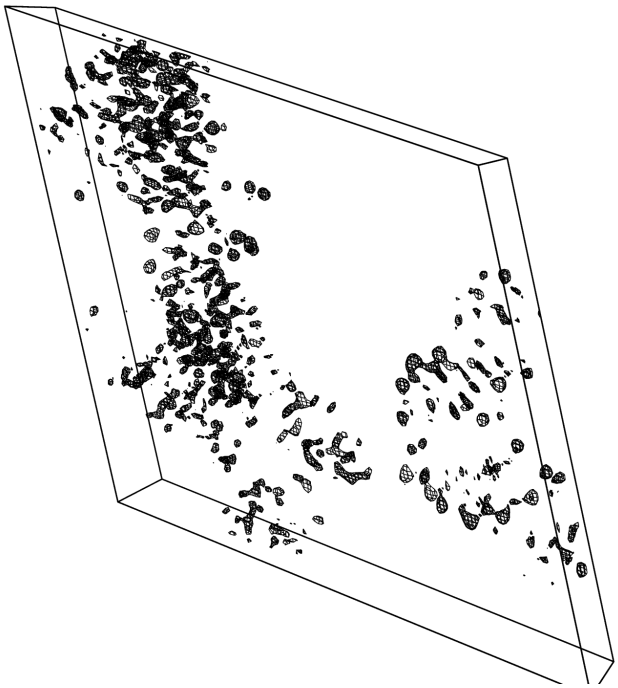


Figure 4.23: Difference electron density map (black mesh) contoured at the $3\sigma$ level between the amplitudes resulting from the ACT pipeline and the FBA pipeline using the phases obtained from the model given by refinement with the data processed using the ACT pipeline. The difference density is not random as it was for the insulin structure. Instead most of the difference density is located around the structure, suggesting that the refined models are likely to differ in several regions.

that the statistical improvement by manually refining the model may be more limited with the data for the FBA pipeline compared to that of the ACT pipeline.

Table 4.2: Final refinement statistics for data processed with the ACT and FBA pipelines.

| | ACT | FBA |
|---|---|---|
| R work | 0.257 | 0.259 |
| R free | 0.282 | 0.288 |
| RMS bond length (Å) | 0.014 | 0.012 |
| RMS Bond Angle (°) | 1.937 | 1.806 |

## 4.7 Discussion

### 4.7.1 Overview of the forward-backward algorithm

The work presented in this chapter introduces a representation of the data collection experiment as a hidden Markov model (HMM). The fundamental idea is that each image obtained in a diffraction experiment is generated from a different crystal. The crystals are related by the fact that the atomic composition of the crystal is the same, and the structural changes due to the X-ray exposure between images are small. The Markov property makes the assumption that the state of one crystal is only dependent on its previous state, not on its entire history. With these assumptions, the Luzzati distribution (**??**) can then be used to mathematically describe how the crystal state evolves (process function). Furthermore the intensity observations are directly proportional to the square of the structure factor amplitude giving the observation function. With this representation, the UKF and URTSS together (FBA) are used to find the optimal amplitude estimates during the experiment. The log likelihood of the resulting amplitude estimates can then be used to determine whether the FBA has converged for each reflection.

Before FBA is applied, the reflection observation data had to be pre-processed to the required format, which required some data manipulation. The main objective was to allocate each observation to an image. For fully recorded reflections there is no ambiguity but for partial reflections this was achieved by matching the calculated centroids of the reflections to the images. Additionally, intensity and partiality estimates were made for reflection observations that were not fully traversed in the experiment. Further to the error values calculated from the integration software, more uncertainty had to be added to the intensity estimates arising from the collapsing of several partial measurements of an observation to a single image collected at one point in time.

The FBA algorithm was tested on simulated reflection data and the results were extremely good for strong data. The resulting amplitude estimates were close to the true values, and the confidence intervals were sensible, decreasing in width when observations were made and increasing further away from observations. For weak data the FBA estimate did not perform as well, so a correction to the initial amplitude value using Bayesian inference (in a

similar manner to the French and Wilson truncation algorithm) was created.

The FBA was successful when applied to real crystallographic data. It generated amplitudes that led to interpretable electron density maps for both insulin and the C.Esp1396I protein-DNA complex. Furthermore, the final model refinement statistics are on a par with those that result from using current data reduction pipeline programs (AIMLESS and CTRUNCATE).

**Advantages of the FBA**

One of the major advantages of the FBA algorithm is that the error estimates are calculated explicitly at each time point in the data collection experiment. These error estimates are a combination of the uncertainty in the integrated measurement (observation covariance) and the uncertainty of the changes suffered by the crystal (process covariance) propagated through time. Since the amplitudes are found directly by the FBA, it abrogates the need to use existing truncation algorithms including the commonly used French and Wilson algorithm (**?**). This is an advantage, because for large experimental uncertainties the French and Wilson algorithm produces error estimates that resemble the Wilson distribution, which result in weak measurements having a significant influence on a structural model (**?**).

The other major advantage of the FBA is the fact that the amplitude estimates are time/dose resolved, so that several electron density maps can be obtained from a single diffraction dataset. In theory, the set of structure factor amplitudes given at each point in time could lead to slightly different models showing structural changes throughout the experiment (e.g. due to radiation damage). However it is unclear how sensitive downstream processing (such as refinement) is to small amplitude changes in a subset of reflections, and whether the processing will influence the resulting models enough to hinder the observation of altered conformations. For example, if it is the case that the applied restraints in refinement are weighted highly, then subtle structural deviations from the “ideal” conformation may be missed. Further investigation will be required to assess this issue.

An additional, albeit minor, advantage to the FBA is that the framework is very modular by design. The process and observation functions are simply based on the current theoretical understanding of structure factor statistics and diffraction theory. If the description of the crystal evolution process were to change, then this could simply be incorporated by changing

the process function and covariance functions, and it would not require a refactoring of the code. Similarly, if the detector were to operate differently and the theory of how the observations (intensities or not) were to alter, then this would only change the process and observation functions. An example of this is that both the Gaussian and Rician process functions exist in the code as separate functions, and either one can be called very simply. Thus it is also very easy to compare different processes.

**Disadvantages of the FBA**

The main disadvantage of the FBA is that in its current implementation, it is computationally expensive. The insulin dataset which consists of 450 images took about 16 hours to process through the algorithm, whereas the C.Esp1396I dataset which only had 100 images took around 4 hours to run despite containing around twice the number of reflections. This suggests that the computational time is largely dependent on the total number of images. There are many ways in which the performance can be improved. An obvious one would be to write it in a faster language such as C++, but the code would still need to be written is a more optimal manner. For example (although not the biggest bottleneck), to read reflection information the program runs MTZDUMP and then parses the resulting text to retrieve the information. This can be improved if the binary MTZ file contents are read directly using the MTZLIB. Another way to speed up the code would be to run the FBA in parallel, which is possible because the reflections can be assumed to be independent.

A fundamental feature of the HMM is that the scale factor is simply a parameter concerned with the observation and it is not intrinsic to the crystal. At first this seems at odds with the current scaling assumption that the scale factor contains terms that are intrinsic properties of the crystal e.g. unit-cell volume, and the fact that the radiation damage parameter(s) are refined simultaneously with the other scaling factors (absorption, detector response to X-ray photon, etc.). However, it should be theoretically possible to separate the factors that are affected by the crystal changes and those that are a property of the observation method (i.e. observing intensities on a Pilatus detector). A simple thought experiment to demonstrate this is to consider a crystal that is irradiated by X-rays with no detector present. The crystal is still going to change due to radiation damage regardless of whether intensity measurements are made. Hence a description of the crystal changes must (in theory) be

possible without reference to a scale factor that describes how the intensity observations are made. Therefore it can be inferred that the HMM representation would operate optimally in the ideal scenario where the scale factor is known. This is not that case as "the only information we have is the measured difference between symmetry-related observations" (**?**).

### 4.7.2 Improvements and extensions

The algorithm presented has yielded promising results but there are still several improvements that could be made to the software, one of the obvious being to improve the scaling procedure. The current method is very primitive and better methods to obtain the scale and B factors have already been proposed (**?**). Furthermore, the program should accommodate methods to handle varying scale and B factors. Non-parametric regression methods such as Gaussian process regression (GPR) could be utilised to do this. GPR makes no assumption about the functional form to describe the evolution of the scale or B factors. An additional benefit is that the Gaussian errors in the regression are also calculated, which allows for the explicit propagation of errors in the algorithm so that a more accurate uncertainty represented by the process and observation covariance values can be obtained. Rather than defining parameters for a certain family of functions (e.g. the gradient and intercept of linear curves) as is the case for parametric regression, non-parametric regression methods usually require the user to loosely define more general properties of a function such as the smoothness and covariance (**?**). Restraints would have to be applied during the regression to ensure that the behaviour of resulting amplitude estimates were "sensible". This may require iterative feedback between the amplitude values and the scale and B factors.

Ultimately, the scaling procedure would benefit from utilising the methods that are currently implemented in software programs such as AIMLESS and XSCALE rather than reinventing the wheel. These are very mature algorithms that reflect the current knowledge and best practice in scaling. Therefore an immediate improvement would be to run AIMLESS and output unmerged, scaled intensity values with B factor correction turned off. Thus the scale factor can be assumed to take the value 1 for every image and the FBA would only then be required to track the resulting changes in the crystal state. The other advantages of doing this are that the current method implemented to extract reflection intensities (described in section 4.4)

would effectively be carried out by AIMLESS. Additionally AIMLESS incorporates an outlier rejection algorithm so another one would not necessarily have to be written for the FBA. In the case where AIMLESS is used to scale the data, the FBA algorithm could be used simply as a truncation algorithm giving dose resolved amplitude estimates with sensible error estimates for all reflections.

One of the assumptions that was made in deriving the process function is that the changes in the structure factor amplitude resulting from the coordinate errors could be absorbed into the temperature factor term. This assumption may not hold and investigation into the coordinate error distributions should also be carried out to determine the true effect of it.

To extend the use of the algorithm, the FBA could be used to merge data from several crystals. First, pairwise cross-covariance matrices between sets of amplitude values resulting from applying the FBA to several different crystals should be calculated. The elements of the resulting matrices can be used to determine whether the data from different crystals can be merged (Garib Murshudov, personal communication).

As mentioned above, one of the major features of the algorithm is that it produces dose resolved amplitude estimates, which provides a unique opportunity to perform radiation damage correction. This means that the behaviour of reflections can be tracked explicitly, particularly for reflections that are observed multiple times. If the assumption is made that reflections in the same resolution bin behave similarly, then the behaviour of reflections that were only observed once can be predicted by determining the “average” behaviour of multiply observed reflections in the same resolution bin. The obstacle with this method that must be overcome is how to average ‘behaviour’ irrespective of the varying scales on which the reflections are observed.

### 4.7.3 Future work

It should be noted that the main goal of the FBA algorithm is to improve diffraction data for experimental phasing, in particular the uncertainty estimates for weak reflections. Additionally, correction for radiation damage should also improve the phasing signal. Therefore the next steps are to apply the algorithm to SAD data (with the extensions/alterations listed above) to determine whether these improvement gains can be realised.