Neutron reflectometry analysis: using model-dependent methods

Andrew R. McCluskey[1]

[1] Diamond Light Source & University of Bath. email: andrew.mccluskey@diamond.ac.uk

March 19, 2020

Neutron reflectometry analysis is an inherently ill-posed, which is to say that there are many possible solutions which agree equally well with the measured data. This leads to the application of model-dependent analysis, where information that we **know** about the system is integrated into our analysis. This tutorial briefly covers the mathematics underlying the use of model-dependent analysis in neutron reflectometry.

Introduction

The aim of model-dependent neutron reflectometry analysis is to create a **model** that accurately reproduces the collected data.[2] The collected data (once normalised) is described in terms of **reflected intensity**, $R(q)$, which depends on the wavevector, q, and is measured as,

$$R(q) = \frac{\text{specular reflected neutron intensity at } q}{\text{incident neutron intensity}}, \tag{1}$$

where the denominator in Equation 1 is the total neutron flux on the sample. It is this $R(q)$ that we want to calculate from some model and compare with our data.

By applying the **Born approximation**,[3] that there is only a single scattering event for each neutron, the reflected intensity can be approximated as follows,[4]

$$R(q) \approx \frac{16\pi^2}{q^4} \left| \int_{-\infty}^{+\infty} \rho'(z) \exp\left(-izq\right) dz \right|^2, \tag{2}$$

where, $\rho'(z)$ is the first derivative of the scattering length density profile.[5] However, this **kinematic** approach has a significant problem, which can be demonstrated by considering a the simple case of a bare silicon substrate, which can be modelled with a Heaviside function, shown in Figure 1(a),

$$\rho(z) = \begin{cases} 0 & \text{where } z < 0, \\ \rho_{\text{Si}} & \text{otherwise,} \end{cases} \tag{3}$$

where, $\rho_{\text{Si}} = 2.074 \times 10^{-6}\,\text{Å}^{-2}$. The first derivative of a Heaviside function is a scaled δ-function, shown in Figure 1(b),

$$\rho'(z) = \rho_{\text{Si}}\delta(z). \tag{4}$$

Figure 1: A graphical representation of the kinematic approach; (a) the Heaviside function describing the scattering length density profile for a bare silicon substrate, (b) the δ-function arising from the first derivative of the function in (a), and (c) the resulting reflectometry profile, where the orange line at $R(q) = 1$ identifies the breakdown of between experiment and theory.
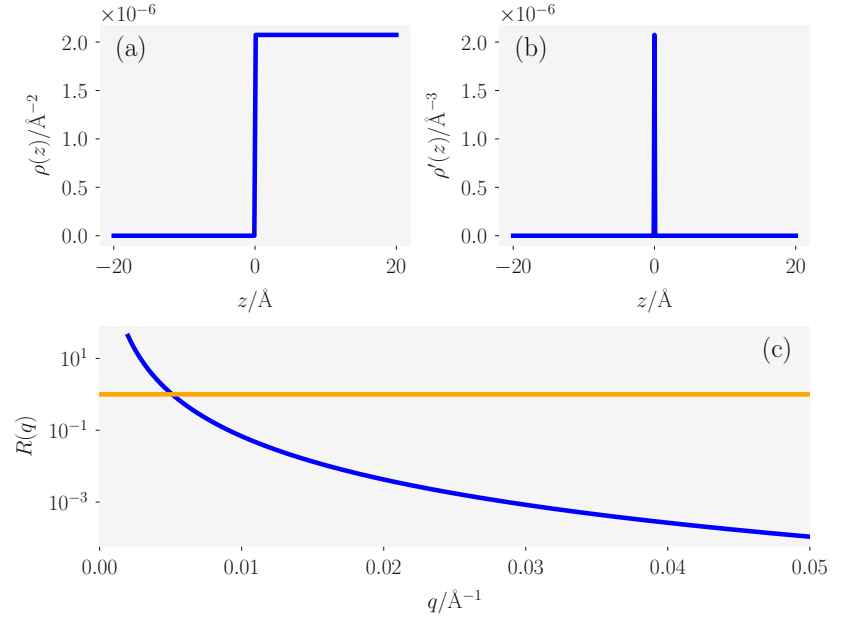
Then, using Equation 2, we can calculate the reflected intensity with respect to q,

$$
\begin{aligned}
R(q) &\approx \frac{16\pi^2}{q^4} \left| \rho_{\text{Si}} \int_{-\infty}^{+\infty} \delta(z) \exp\left(-izq\right) dz \right|^2 \\
&\approx \frac{16\pi^2}{q^4} \left| \rho_{\text{Si}} \exp\left(0\right) \right|^2 \\
&\approx \frac{16\pi^2 \rho_{\text{Si}}^2}{q^4}.
\end{aligned}
\tag{5}
$$

This reflectometry profile is shown in Figure 1(c), where it is clear that the agreement with the experimental profile would be poor as $q \to 0$.[6] For low values of q, the model reflectometry from this simple system is greater than 1, violating the physical condition imposed by Equation 1[7]. This breakdown of the kinematic approach is due to the assumption of the Born approximation, that each neutron is only scattered once. The reflectometry geometry, with the large path length that arises the shallow incidence angle, means that multiple scattering events are likely, rendering the kinematic approach invalid.

Recursive methods

The breakdown of the kinematic approach has lead to the application of the Abelès,[8] or Parratt,[9] recursive model for the reflection of light at a given number of stratified interfaces.[10] This is an alter-

[6] C. F. Majkrzak and N. F. Berk. *Phys. Rev. B*, 58(23):15416, 1998. DOI: 10.1103/PhysRevB.58.15416

[7] In the kinematic approach, more neutrons are reflected than are indicent in the first place!

[8] F. Abelès. *Ann. Phys.*, 12(3): 504–520, 1948. DOI: 10.1051/anphys/194812030504

[9] L. G. Parratt. *Phys. Rev.*, 95(2):359–369, 1954. DOI: 10.1103/PhysRev.95.359

[10] This method is also known as the dynamical approach.

native method for calculating the reflected intensity from a model and involves describing the system as a series of layers of different scattering length density. The neutrons can be either reflected or refracted at the interface between each of the layers. Figure 2 shows
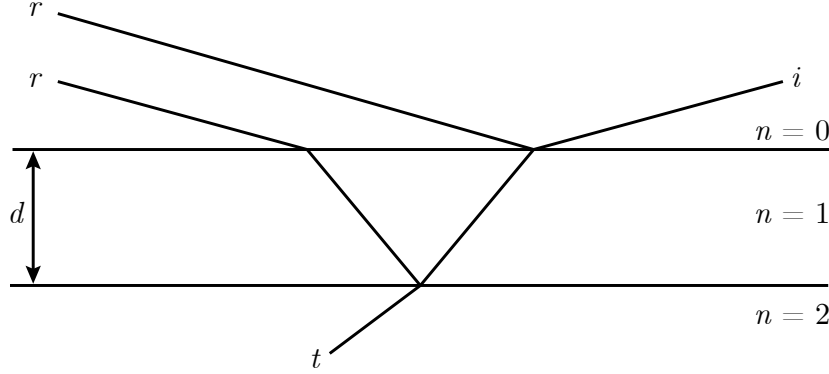

Figure 2: A schematic diagram showing the reflected (r) and transmitted (t) waves when an incident (i) wave enters an interface of thickness d.

the refraction and reflections from a simple two layer system, where layer 0 is a vacuum above the sample. It is clear how the two waves, labelled r, could interfere constructively or destructively depending on the thickness of layer 1, d.

The generalisation of this approach to any number of layers is possible and enables the reflected intensity to be calculated at each value of q that was measured. The incident neutrons will be refracted by each of the layers, giving wavevectors for each layer, k_n,

$$k_n = \sqrt{k_0 + 4\pi(\rho_n - \rho_0)}, \tag{6}$$

where, $k_0 = 0.5q$. The Fresnel equation coefficient between layers n and $n + 1$, $r_{n,n+1}$i, can then be found along with the phase factor, β_n, for the layer n,

$$r_{n,n+1} = \frac{k_n - k_{n+1}}{k_n + k_{n+1}}, \tag{7}$$

$$\beta_n = k_n d_n. \tag{8}$$

From the Fresnel equation coefficient and phase factor, the characteristic matrix for each layer, M_n, can be constructed,

$$M_n = \begin{matrix} \exp \beta_n & r_{n,n+1} \exp -\beta_n \\ r_{n,n+1} \exp \beta_n & \exp -\beta_n \end{matrix}, \tag{9}$$

and the resultant matrix B is found from the product sum of the matrices from each layer,

$$B = \prod_{n=0}^{n_{max}} M_n. \tag{10}$$

The final model reflected intensity at a given value of q is found from the following elements of the resultant matrix,

$$R(q) = \frac{B_{1,2}}{B_{1,1}}. \tag{11}$$

This algorithm models the layers as **perfectly** flat layers, which will not be strictly true. This has resulted in the use of correction terms to be added to Equation 7 to account for this **roughness**. The most common of these is Névot and Croce Gaussian broadening,[11] in which the Fresnel equation coefficient is evaluated as,

[11] L. Névot and P. Croce. *Rev. Phys. Appl. (Paris)*, 15(3):761–779, 1980. DOI: 10.1051/rphysap:01980001503076100

$$r_{n,n+1} = \frac{k_n - k_{n+1}}{k_n + k_{n+1}} \exp\left(-2k_n k_{n+1} \sigma_{n,n+1}^2\right), \tag{12}$$

where, $\sigma_{n,n+1}$ is the interfacial roughness between the layers n and $n+1$. Applying this **dynamical approach** is shown in Figure 3, where there is a clear difference between the kinematic and dynamical approaches as $q \rightarrow 0$, with the dynamical approach adhering to the physical constraint that causes the breakdown of the kinematic approach.

Figure 3: A comparison of the kinematic (blue) and dynamical (green) approaches to determine the reflected intensity from the material with the scattering length density profile given in Figure 1(a).
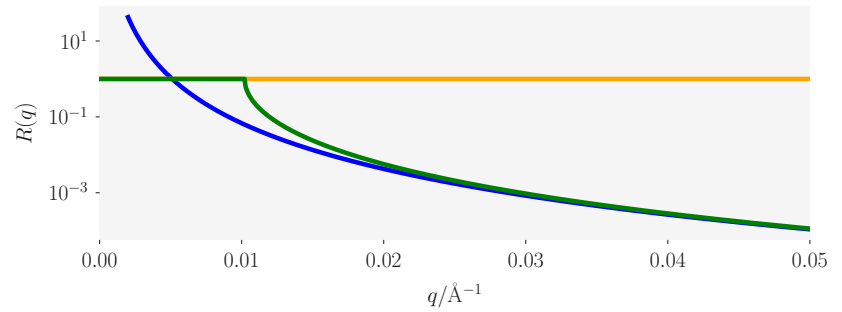
Global optimisation

The recursive method described above gives an accurate method to obtain a model reflected intensity. However, this is just the first step in the analysis of a neutron reflectometry dataset. Now we are interested in optimising our model such that the reflected intensity from it matches our experimental data as best as possible. This is the problem of parameter optimisation, which is a broad area of mathematics and computer science that we will not dwell on here.[12] However, we will introduce the basics of optimisation and mention the most common **global optimisation** method applied in reflectometry.

When we measure a reflectometry profile, we measure the reflected intensity (and some uncertainty in that measurement) at

[12] There are whole journals dedicated to the subject, see SIAM Journal on Optimization or Journal of Global Optimization.

discrete points in the wavevector, $R(q) \pm \delta R(q)$. Using the recursive method discussed above, we can calculate the model reflected intensity at these same q values, $R_m(q)$. We then aim to reduce the difference between the measured and modelled reflected intensity through the optimisation (maximisation) of the **likelihood** $\mathcal{L}$,

$$\mathcal{L} = \exp\left\{ -0.5 \sum_{q=q_{\min}}^{q_{\max}} \left[\frac{R(q) - R_m(q)}{\delta R(q)} \right]^2 + \ln[2\pi\delta R(q)] \right\}. \quad (13)$$

This parameter describes the **probability** that the model m accurately describes the observed reflectometry data. The aim in reflectometry fitting is to obtain model reflectometry at has the maximum likelihood possible for the given experimental data. Figure 4 shows the maximum likelihood model for some experimental data, in addition to another model which doesn't agree well with the data and has a lower likelihood as a result.
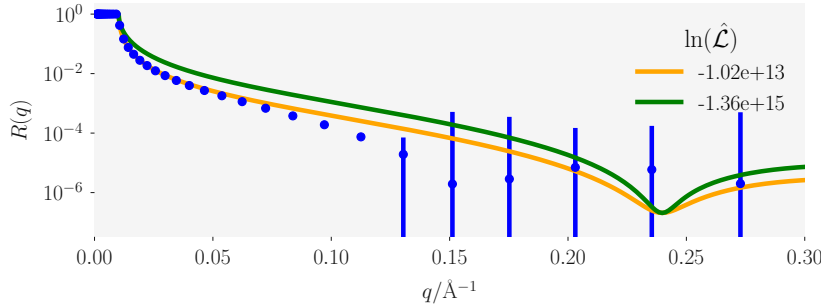

Figure 4: The maximum likelihood fit, the blue data points indicate some experimental data, while the yellow line shows the maximum likelihood model fit to this dataset and the green line show another model (which doesn't maximise the likelihood).

The global optimisation of a reflectometry model is a particularly difficult problem, this is due to the **ill-posed nature** of this data, this is where there are many reasonable solutions to a particular reflectometry profile.[13] However, a particular global optimisation method has shown substantial utility in the fitting of reflectometry data,[14] **differential evolution**.[15] This has lead to the inclusion of this method in many common reflectometry analysis packages.[16]

Differential evolution is an iterative, genetic algorithm, designed to mimic the evolution processes observed in biology.[17] The method consists of two vectors, the parent population **p**, and the offspring population, **o**. These vectors are of shape $(i \times j)$, where i is the number of parameters in the model and j is the number of candidate solutions being considered. The offspring population is generated as a result of some trail method.[18]

A classical trail method consists of two stages, mutation and recombination. The **mutation** stage involves the creation of a mutant population, **m**. The magnitude of the mutation is dependent on the

[13] This can also be thought of as *every reflectometry curve looks the same*, which you may have heard before.

[14] A. van der Lee, F. Salah, and B. Harzallah. *J. Appl. Crystallogr.*, 40(5):820–833, 2007. DOI: 10.1107/S0021889807032207

[15] M. Wormington, C. Panaccione, K. M. Matney, and D. K. Bowen. *Philos. Trans. R. Soc. London Ser. A*, 357(1761):2827–2848, 1999. DOI: 10.1098/rsta.1999.0469

[16] M. Björck. *J. Appl. Crystallogr.*, 44(6):1198–1204, 2011. DOI: 10.1107/S0021889811041446

[17] J. H. Holland. *Adaptation in Natural and Artificial Systems*. MIT Press, 2 edition, 1992. ISBN 978-0-262-58111-0

[18] Here we will only discuss a simple classical trail method, however, many of these exist.

first of our **hyperparameters**, the mutation constant, k_m,

$$\mathbf{m}_{i,j} = b_i + k_m(\mathbf{p}_{i,R1} - \mathbf{p}_{i,R2}),\tag{14}$$

where b is the candidate solution with the greatest likelihood, and $\mathbf{p}_{i,R1}$ and $\mathbf{p}_{i,R2}$ are randomly chosen members of the parent population. The mutation constant hyperparameter controls the size of the search space, with a large k_m corresponding to a wider search.

The **recombination** step creates the offspring population vector by taking a sample from either the parent or mutant population with some frequency, which depends on our second **hyperparameter**, the recombination constant, k_r,

$$\mathbf{o}_{i,j} = \begin{cases} \mathbf{m}_{i,j} & \text{where } X < k_r, \\ \mathbf{p}_{i,j} & \text{otherwise,} \end{cases}\tag{15}$$

where, X is a random number selected from a uniform distribution between 0 and 1. The recombination constant hyperparameter controls the mutation frequency in the offspring population.

The final stage is to compare the offspring and parent population vectors, in the **selection** stage, to create the parent population for the next iteration. Here, the likelihood is used to compare between subunits from the offspring or parent populations,

$$\mathbf{p}_{*,j} \leftarrow \begin{cases} \mathbf{o}_{*,j} & \text{where } \mathcal{L}_{\mathbf{o}_{*,j}} > \mathcal{L}_{\mathbf{p}_{*,j}}, \\ \mathbf{p}_{*,j} & \text{otherwise,} \end{cases}\tag{16}$$

where, the $*, j$ subscript notation indicates all objects from the population, j.

The differential evolution algorithm can be seen in action applied to the negative two-dimensional Ackley function,[19] in Figure 5.[20] This function can maximise the value of the negative Ackley function. While this does not offer a clear example of this algorithm's application to reflectometry analysis, the popularity of this method in the fitting of reflectometry profiles cannot be denied.[21]

Uncertainty sampling

Reflectometry measurements offer an **average** description of the out of place structure of a material. This means that it is pragmatic to describe the uncertainties in the values of our model parameters in some fashion. This is often performed using **Markov-chain Monte Carlo** (MCMC), which is a methodology used to sample the **posterior probability distribution** for each of our parameters.[22] Typically, MCMC is used on already optimised solutions to a particular problem, is in reflectometry analysis it is usually applied after the differential evolution has optimised the structure. In addition to being able

[19] D. H. Ackley. *A Connectionist Machine for Genetic Hillclimbing*. University of Michigan

[20] The Ackley function is a common function used in the assessment of global optimisation functions. This is due to there being a large number of local minima and only a single global minimum to this function. Here, we want to maximise the value, so the negative Ackley function is used.

[21] M. Björck. *J. Appl. Crystallogr.*, 44(6):1198–1204, 2011. DOI: 10.1107/S0021889811041446; and A. R. J. Nelson and S. W. Prescott. *J. Appl. Crystallogr.*, 52(1):193–200, 2019. DOI: 10.1107/S1600576718017296

[22] D. S. Sivia and J. Skilling. *Data Analysis: A Bayesian Tutorial*. Oxford University Press, 2 edition, 2006. ISBN 978-0-19-856832-2
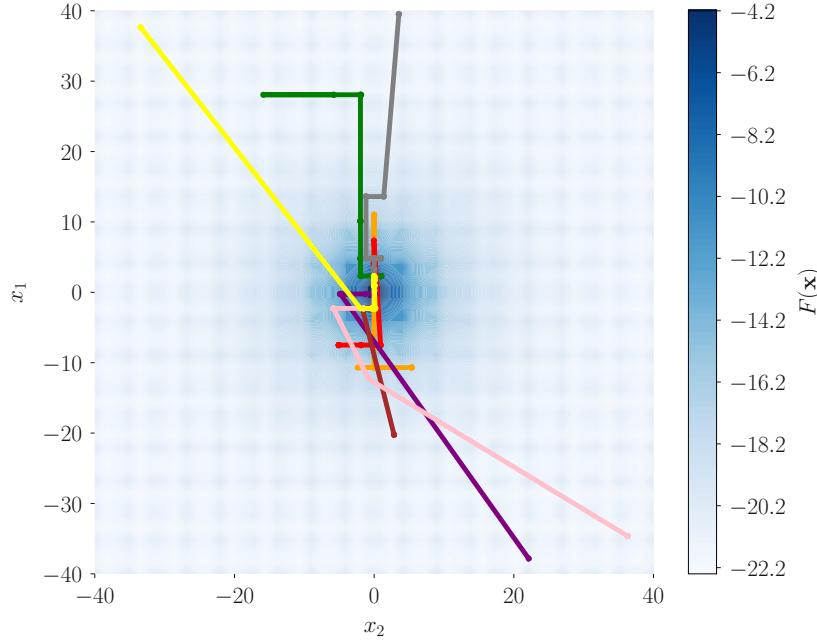
Figure 5: An example of a differential evolution algorithm applied to the negative of an Ackely function. In this implementation $k_m = 0.5$ and $k_r = 0.5$. Each line represents a different candidate solution. The optimisation was stopped after 100 iterations had run.

to quantify the **inverse uncertainties**,[23] MCMC also offers a more complete understanding of the correlations between different parameters,[24] which is particularly important in the ill-posed reflectometry analysis.

Once an optimised set of model parameters, θ, are obtained which maximise the likelihood, $\mathcal{L}$, some random perturbation is applied,

$$\Theta = \theta + aR, \tag{17}$$

where R is some normally distributed number centered on 0 with a standard distribution of 1 and a is the step size. A new $\mathcal{L}$ is found for Θ and the probability that this transition is will occur is found,

$$p = \exp\left[\frac{\mathcal{L}()\Theta) - \mathcal{L}(\theta)}{2}\right]. \tag{18}$$

This probability is then compared with a uniformly distributed random number from 0 to 1, n,

$$\theta \leftarrow \begin{cases} \Theta & \text{where } n < p, \\ \theta & \text{otherwise.} \end{cases} \tag{19}$$

This process is repeated until some desired number of samples has been obtained. It should be noted that it is important to allow the Markov chains to have some "burn-in" period, which is not included in the final samples. This allows the MCMC algorithm to settle into the search-space.

[23] This is the name given to the uncertainties in the model parameters.

[24] W. Gilks, S. Richardson, and D. Spiegelhalter. *Markov Chain Monte Carlo in Practice.* Chapman & Hall/CRC Interdisciplinary Statistics. CRC Press, 1995. ISBN 978-0-412-05551-5

[25] D. Foreman-Mackey, W. Farr, M. Sinha, A. Archibald, D. Hogg, J. Sanders, J. Zuntz, P. Williams, A. Nelson, M. de Val-Borro, T. Erhardt, I. Pashchenko, and O. Pla. *J. Open Source Softw.*, 4(43):1864, 2019. DOI: 10.21105/joss.01864

Figure 6 shows the result of an MCMC sampling for a pair of overlapping Gaussian functions, performed using the `emcee` Python package.[25] The posterior probability distributions that are available to each of the four parameters are shown along with the data, the optimised fit and a subset of models within the posterior distributions. All of the samples from the posterior distributions fit within the uncertainty error bars on the data. This shows the ability for MCMC to sample the range of the distribution that is **allowed** by the experimental uncertainty.

Figure 6: An example of the result of MCMC uncertainty sampling on data synthesized from two overlapping Gaussian functions. θ_1 and θ_2 are the integrals of the Gaussian functions while θ_3 and θ_4 are the positions. The data is shown with blue circles, the optimised model with an orange line, and samples from the posterior distributions with green lines.
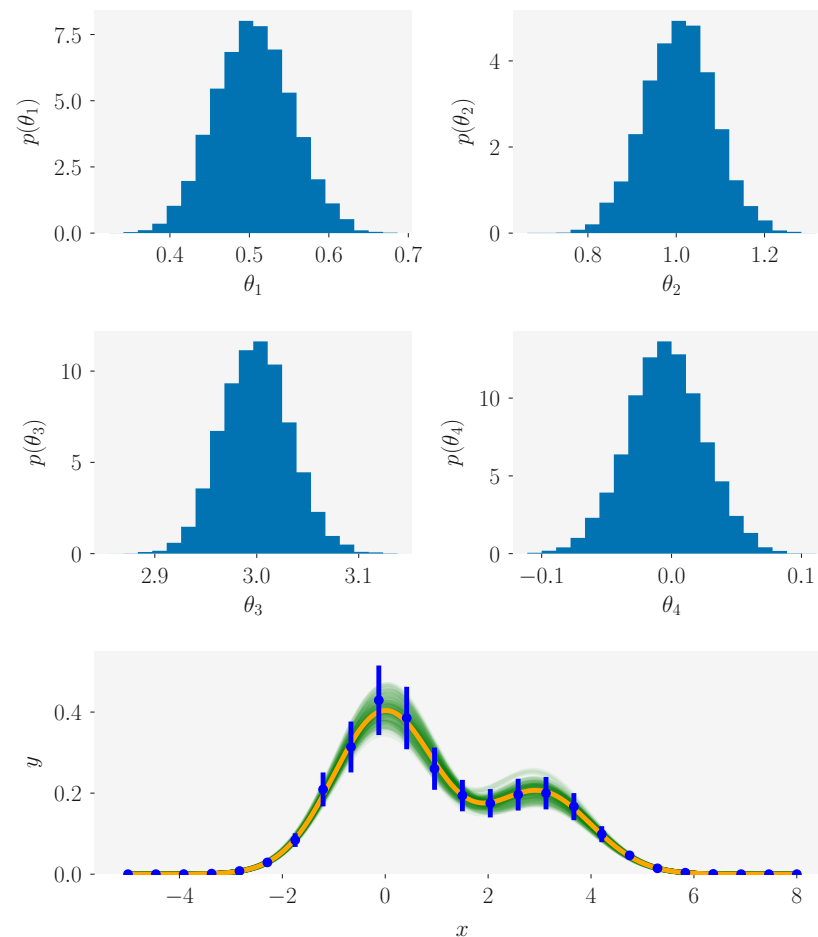
Conclusions

The aim of this document was to give an introduction to some of the mathematical concepts that underpin modern reflectometry model-dependent analysis. We have looked at how reflectometry

is calculated from a layer model description of the scattering length density profile, and why the kinematic approach fails. Then we have discussed the importance of the differential evolution algorithm in reflectometry analysis and detailed the operation of this algorithm. Finally, Markov-chain Monte Carlo was introduced in the context of uncertainty quantification for model-dependent analysis, with particular importance for reflectivity discussed. While not exhaustive, I hope that this document will give you the confidence in understanding to look in more detail into how the analysis of reflectometry measurements are performed.

References

F. Abelès. *Ann. Phys.*, 12(3):504–520, 1948. DOI: 10.1051/anphys/194812030504.

D. H. Ackley. *A Connectionist Machine for Genetic Hillclimbing*. University of Michigan.

M. Björck. *J. Appl. Crystallogr.*, 44(6):1198–1204, 2011. DOI: 10.1107/S0021889811041446.

M. Born. *Z. Physik*, 38:803–827, 1926. DOI: 10.1007/BF01397184.

D. Foreman-Mackey, W. Farr, M. Sinha, A. Archibald, D. Hogg, J. Sanders, J. Zuntz, P. Williams, A. Nelson, M. de Val-Borro, T. Erhardt, I. Pashchenko, and O. Pla. *J. Open Source Softw.*, 4(43):1864, 2019. DOI: 10.21105/joss.01864.

W. Gilks, S. Richardson, and D. Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC Interdisciplinary Statistics. CRC Press, 1995. ISBN 978-0-412-05551-5.

J. H. Holland. *Adaptation in Natural and Artificial Systems*. MIT Press, 2 edition, 1992. ISBN 978-0-262-58111-0.

M. R. Lovell and R. M. Richardson. *Curr. Opin. Colloid Interface Sci.*, 4 (3):197–204, 1999. DOI: 10.1016/S1359-0294(99)00039-4.

C. F. Majkrzak and N. F. Berk. *Phys. Rev. B*, 58(23):15416, 1998. DOI: 10.1103/PhysRevB.58.15416.

A. R. J. Nelson and S. W. Prescott. *J. Appl. Crystallogr.*, 52(1):193–200, 2019. DOI: 10.1107/S1600576718017296.

L. Névot and P. Croce. *Rev. Phys. Appl. (Paris)*, 15(3):761–779, 1980. DOI: 10.1051/rphysap:01980001503076100.

L. G. Parratt. *Phys. Rev.*, 95(2):359–369, 1954. DOI: 10.1103/Phys-Rev.95.359.

D. S. Sivia. *Elementary Scattering Theory: For X-Ray and Neutron Users.* Oxford University Press. ISBN 978-0-19-922868-3.

D. S. Sivia and J. Skilling. *Data Analysis: A Bayesian Tutorial.* Oxford University Press, 2 edition, 2006. ISBN 978-0-19-856832-2.

A. van der Lee, F. Salah, and B. Harzallah. *J. Appl. Crystallogr.*, 40(5): 820–833, 2007. DOI: 10.1107/S0021889807032207.

M. Wormington, C. Panaccione, K. M. Matney, and D. K. Bowen. *Philos. Trans. R. Soc. London Ser. A*, 357(1761):2827–2848, 1999. DOI: 10.1098/rsta.1999.0469.