LiquidDiffract: software for liquid total scattering analysis

Benedict J. Heinen1, James W. E. Drewitt2

Received: 17 September 2021 / Accepted: 31 December 2021 / Published online: 3 May 2022
© The Author(s) 2022

Abstract
LiquidDiffract is an open source, Python-based graphical application for X-ray total scattering analysis of liquids and disordered solids. The software implements procedures to obtain information on macroscopic bulk properties and local atomic-scale structure of monatomic or polyatomic samples from X-ray total scattering data. LiquidDiffract provides an easy to use interface with tools to perform background subtraction; calculation, normalisation, and refinement of the reciprocal-space structure factor and real-space correlation functions; and the extraction of structural information such as bond lengths, coordination number, and bulk density. The software is well suited to investigations of amorphous materials at extreme conditions, such as studies of high-pressure melt structure, polyamorphic phase transitions, and liquid equations of state. The open-source distribution and graphical interface will be of particular benefit to researchers who are new to the field. In this article we describe the distribution, system requirements, and installation of LiquidDiffract, and detail the data processing workflow and underlying numerical methods.

Keywords Computer programs · Liquid structure · Total scattering analysis

Introduction
Despite a lack of long-range order, the structure of liquids and glasses exhibit a high-degree of ordering on short- to medium-range length scales, originating from chemical interactions that lead to arrangements of local structural motifs (Kitaigorodskiy and Chomet 1967). The study of liquid and glass structure provides insight into the behaviour and physical properties of amorphous materials that is important in many fields of research. At extreme conditions of high temperature ($T$) and pressure ($p$) an understanding of liquid structure and polyamorphic (liquid–liquid) phase transitions has important industrial applications, including novel phase discovery, and modelling of industrial melt processes (Drewitt 2021). Fluids and melts are also key to our understanding of the dynamic Earth, occurring from the crust to the core and controlling mass transfer in the deep interior, and geothermal, volcanic, and ore-forming processes at the near surface.

For liquids and disordered solids, X-ray diffraction provides direct information on the average atomistic structure. The total intensity of scattered radiation arises from coherent, incoherent (Compton), and self-scattering effects. These components can be separated out, and the structure factor, $S(Q)$, extracted following a normalisation procedure. The structure factor is a measure of the coherent scattering by atoms found at different positions in the sample, and hence contains the useful structural information about the material. $S(Q)$ is a reciprocal-space function, where $Q$ is the scattering vector, usually measured in Å$^{-1}$, which is related to the wavelength of the incident radiation, $\lambda$, and the scattering angle, $2\theta$, by:

$$Q = \frac{4\pi \sin \theta}{\lambda}. \tag{1}$$

It is generally more useful to interpret the structure of disordered materials by considering correlation functions in real-space, the primary example being the pair distribution function, $g(r)$, which describes variations in the microscopic...
pair density of a sample with distance, $\rho(r)$, proportional to the average (bulk) number density ($\rho_0$):

$$g(r) = \frac{\rho(r)}{\rho_0}. \quad (2)$$

These variations arise when correlations between pairs of atoms are more likely to occur at a specific separation distance $r$. Peaks in the $g(r)$ are therefore centered around the average bond lengths of coordination shells, and the $g(r)$ approaches 1 at high $r$ as variations from the bulk density become negligible. The pair distribution function is related to the structure factor via a sine Fourier transform:

$$g(r) = 1 + \frac{1}{2\pi^2 r \rho_0} \int_0^\infty Q |i(Q)| \sin(Qr) dQ. \quad (3)$$

where the interference function, $i(Q)$, is a simple transformation of the structure factor such that $\lim_{Q \to \infty} i(Q) = 0$ (Eq. 8).

LiquidDiffract is a data analysis application that computes these functions from x-ray diffraction data, and provides a range of tools to extract structural information in an easy to use, graphical interface. It can be used to obtain information on macroscopic bulk properties and local atomic-scale structure of monatomic or polyatomic samples from x-ray total scattering data.

While there is a wealth of tools available to analyse liquid structures obtained from simulations (e.g., ISAACS, Le Roux and Petkov 2010; MDAnalysis, Michaud-Agrawal et al. 2011; STRFACT, Bernardes 2021; Correlation, Rodriguez et al. 2021; POLYANA, Dimitroulis et al. 2015; GROMACS, Van Der Spoel et al. 2005), there is not a wide range of software available for total x-ray scattering analysis. Some of the tools available as part of XPDFSuite (e.g., PDFgetX3; Juhás et al. 2013) may provide similar functionality to LiquidDiffract, however this is non-free software. XPDFSuite is also optimised for disordered crystalline materials, and does not implement correct normalisation of the $S(Q)$/$g(r)$ for liquids or amorphous solids. The software RAD and RAD-2 (Petkov 1989) may similarly provide some overlapping functionality, but is also non-free and appears obsolete, with binaries only available for the Windows 7 operating system. The software GudrunX (Soper 2017) provides routines for data reduction from 2D diffraction images, complex background refinement, and calculation of PDF functions, but is also distributed under a non-free license. It does not implement methods for extracting density or methods of optimising the $S(Q)$ normalisation that are suitable for high-pressure experimental data. GudrunX is also not particularly accessible, as binaries are only available for Windows operating systems, and the code is difficult to install on other platforms. Briggs et al. (2019) and Yu et al. (2019) cite the use of the program Glassure in their data analysis. Glassure (Prescher 2017) is libre, fairly simple to install (it is also python-based) and can be used to calculate and properly normalise the $S(Q)$ and $g(r)$ from diffraction data. However, some features, such as density refinement, are not fully functional in the latest available version. Unlike LiquidDiffract, Glassure also does not implement methods to investigate the radial distribution function, RDF($r$), or extract estimates of coordination number and bond-length.

There is a clear need for software such as LiquidDiffract, which provides an open-source and libre tool for x-ray total scattering analysis of liquids and disordered solids. The procedures implemented to extract bulk liquid number density will be of particular benefit to researchers performing experiments at extreme conditions, as the equations of state of many liquids and glasses at high-$p$ are poorly known (Drewitt 2021). The efficacy of LiquidDiffract to produce accurate structure factors and equation of state data of liquids under extreme conditions has been successfully demonstrated in published studies of liquid gallium to 25 GPa (Drewitt et al. 2020), and liquid tantalum shock-released from several hundred GPa (Katagiri et al. 2021). We hope that the accessible distribution and simple graphical interface of LiquidDiffract will be of particular benefit to researchers who are new to the field.

**Distribution, licence, installation, and dependencies**

LiquidDiffract is a pure-Python application, ensuring that installation requires no compiling by the user, and that the software is platform-independent. A Python version $\geq$3.5 is required. LiquidDiffract makes use of the open source scientific Python ecosystem for data processing and numerical operations, including the SciPy (Virtanen et al. 2020) and NumPy (Harris et al. 2020) libraries. ‘lmfit’, an extension package to the curve-fitting functionality in SciPy, is used for curve-fitting procedures as it provides a higher-level interface to minimisation routines that promotes easier extensibility, either in future versions of LiquidDiffract, or user defined code that makes use of LiquidDiffract’s core functionality (Newville et al. 2014). The graphical user interface (GUI) is written using the PyQt5 library (https://www.riverbankcomputing.com/software/pyqt5/), which provides Python bindings for version 5 of the Qt C++ cross-platform application framework. The PyQtGraph library (http://www.pyqtgraph.org/) is used to display and plot data. PyQtGraph is a pure-python graphics library built on PyQt that is intended for scientific use cases, and achieves fast updating of displays due to its heavy leverage of NumPy arrays and Qt’s GraphicsView framework. PyQtGraph was chosen for this speed performance, as well as for the in-built collection of tools to provide intuitive interaction controls to the user. In addition, the importlib_resources package (a backport of the
Python 3.9 standard library module importlib.resources) is required for Python versions <3.7.

LiquidDiffract is freely available and distributed under the GNU General Public License v3.0 (GPLv3). The source-code is hosted on a public GitHub repository located at https://github.com/bjheinen/LiquidDiffract, and the package is primarily distributed via The Python Package Index, PyPI (https://pypi.org/project/LiquidDiffract/). Python’s standard package installer, pip, automatically tracks the PyPI so this allows users to install LiquidDiffract from any computer with a Python installation and an internet connection with the command: $ pip install LiquidDiffract. Alternatively, pip can also be used to install LiquidDiffract from the source code downloaded to a local directory. All of the dependencies are automatically handled and installed by pip when installing LiquidDiffract.

Application description and workflow

The user interface of LiquidDiffract is designed around the normal steps and procedures in an experimental data processing pipeline to enable an efficient workflow. The interface is organised into four main tabs which each provide a selection of toolboxes for data operations at sequential stages of the workflow. These are: (1) Background subtraction, (2) Data operations and structure factor refinement, (3) Calculation and output of PDF functions, and (4) Extraction of coordination numbers. The data are automatically visualised graphically and the tabs updated as operations are made through the workflow (Fig. 1).

Background scaling and subtraction

The first tab allows data and optional background files to be loaded into the software and inspected. LiquidDiffract expects $Q$-space diffraction data but a toolbox is provided to convert experimental files saved with $2\theta$ values to $Q$-space. The measured or background intensity profile, $I_{\text{Measured}}(Q)$ or $I_{\text{Background}}(Q)$, is uniformly sampled from a cubic spline interpolation of the raw data, which is also used to extrapolate low-$Q$ values to $Q = 0$. The sampling rate ($Q$-step) can be set by the user and the background profile scaled and subtracted if present. A background auto-scale feature (that roughly scales the background to match the data profile) is provided to speed up workflow but is not recommended for a close fit. Subtracting a scaled background pattern from the measured intensity gives the scattering intensity of the sample, $I_{\text{Sample}}(Q)$:

$$I_{\text{Sample}}(Q) = I_{\text{Measured}}(Q) - bI_{\text{Background}}(Q).$$

More complex $Q$-dependent background corrections that are sometimes required for high-pressure experiments and

![Fig. 1](image.png) The user interface of LiquidDiffract showing data operations being made on sample data in the refinement tab.
other complex sample environments are not currently implemented. In these cases pre-processed, background corrected data can be loaded in.

**Data operations and S(Q) refinement**

The second tab is where most of the data processing is performed. The sample composition and data processing options can be set to calculate the structure factor \( S(Q) \) and corresponding real-space \( g(r) \) function. This tab also offers an option for numerical refinement of both the \( S(Q) \) normalisation and average sample density. Functionality is provided by several toolboxes which group operations for different stages of the workflow.

The *Composition Toolbox* allows the user to set the atomic species present in the sample, and set the atomic number density, \( \rho_0 \). If the user later chooses to refine the sample density then this value of \( \rho_0 \) is also passed to the solver as an initial estimate.

The *Data Options Toolbox* then allows the user to set the majority of important options for data pre-processing and calculation of the normalised total structure factor, \( S(Q) \). An option is provided to smooth noise in the data, which applies a Savitzky-Golay filter. The length of the filter window and order of the polynomial used to fit samples can be set in an *Additional Preferences* dialog, accessible from the *Tools* menu.

The \( Q_{\text{min}} \) cut-off setting allows the user to mask spurious peaks or oscillations in the low-\( Q \) region of the data which can arise from the experimental set-up (e.g. from a slightly mislocated beam stop). Toggling this option sets \( I(Q < Q_{\text{min}}) = I(Q_{\text{min}}) \).

The \( Q_{\text{max}} \) cut-off setting allows the user to truncate the data at high-\( Q \). At high \( Q \)-values experimental data often becomes increasingly noisy because the relative contribution of coherent scattering to the experimental signal is inversely related to the scattering vector. The increasingly noisy signal can lead to dramatic and anomalous oscillations near the first peak in the real space correlation functions. It can therefore be beneficial to truncate experimental data at high-\( Q \).

Irrespective of the \( Q_{\text{max}} \) value chosen by the user, experimental data is always effectively truncated due to the finite \( Q \)-range available to experimental measurements. This truncation is essentially an ideal low-pass brick-wall filter, and so results in spurious oscillations in the real-space correlation function via the Gibbs phenomenon. The effect corresponds to the time-domain ‘ringing’ artifacts seen in signal processing, and is equivalent to convolution of the real-space correlation function with the impulse response of the filter - the real space peak function \( P(r) \). For a simple truncation the \( P(r) \) is a normalised sinc function, i.e.:

\[
P(r;Q_{\text{max}}) = 2Q_{\text{max}} \sin \left( \frac{2\pi r}{Q_{\text{max}}} \right) \frac{\sin \left( \frac{\pi r}{Q_{\text{max}}} \right)}{\pi r}.
\]

The interference function, \( \imath(Q) \), tends to zero at \( Q = \infty \), so the effect of truncating at very high-\( Q \) \(( \gtrsim 30\, \text{Å}^{-1}) \) is often negligible. However, the effect is much more significant for high-pressure studies due to the limited accessible scattering angle when working with high-pressure instruments (Drewitt 2021). Furthermore, truncating at a lower \( Q_{\text{max}} \) corresponds to a broadening of the sinc function, and these broader oscillations are often difficult to distinguish from real structural information (Lorch 1969; Shen et al. 2004).

A filter with a smoother roll-off can be used to suppress FT artifacts in the real-space functions. This is done by applying a window function to the low-pass filter, equivalent to multiplying the \( Q \)-space data by a modification function, \( M(Q) \), which damps the data cut-off at \( Q_{\text{max}} \) (i.e., reduces the amplitude of the discontinuity). Two window functions are available in LiquidDiffract: the Lanczos (1966) window proposed by Lorch (1969, commonly called the Lorch function in total scattering literature):

\[
M(Q) = \begin{cases} 
\sin \left( \frac{\pi Q}{Q_{\text{max}}} \right) & \text{if } Q < Q_{\text{max}} \\
0 & \text{if } Q > Q_{\text{max}} 
\end{cases}
\]

and a cosine window function (i.e., a Hann window; Blackman and Tukey 1958; Drewitt et al. 2013):

\[
M(Q) = \begin{cases} 
1 & \text{if } Q < Q_{\text{Window}} \\
0.5 \left[ 1 + \cos \left( \frac{\pi x}{N-1} \right) \right] & \text{if } Q_{\text{Window}} < Q < Q_{\text{max}} \\
0 & \text{if } Q > Q_{\text{max}} 
\end{cases}
\]

where \( N \) is width of the window function and \( x \) is an integer with values from 0 to \((N-1)\) across the window.

After setting these options, the ‘Calculate \( S(Q) \)’ button will calculate the total structure factor, \( S(Q) \), and display the interference function, \( \imath(Q) \), and the differential correlation function, \( D(r) \), to the user. LiquidDiffract provides the option to use either the Faber–Ziman (FZ; Faber and Ziman 1965) or Ashcroft–Langreth (AL; Ashcroft and Langreth 1967) formalism of the total structure factor \( S^{\text{FZ}}(Q) \) and \( S^{\text{AL}}(Q) \) respectively; see also: Keen 2001). These are detailed fully in Appendix A (Eqs. 32, 37, and associated).

In both cases, the total structure factor is converted to atomic (normalised) units using the method of Krogh-Moe (1956) and Norman (1957). The interference function, \( \imath(Q) \), is simply related to \( S(Q) \):

\[
\imath(Q) = S(Q) - S_{\infty}.
\]
with $S_{\infty} = 1$ for all samples in the FZ formalism and for monatomic samples in the AL formalism (Eq. 42). The differential correlation function, $D(r)$, describes the difference between the (local) microscopic pair density and the average density:

$$D(r) = 4\pi r [\rho(r) - \rho_0].$$

(9)

It is related to $g(r)$, the pair-distribution function (Eq. 2), by:

$$D(r) = 4\pi \rho_0 [g(r) - 1]$$

(10)

or, following from Eq. 3:

$$D(r) = \frac{2}{\pi} \int_0^{Q_{\text{max}}} \mathcal{Q}(Q) \sin( qr ) dQ.$$

(11)

In LiquidDiffract, $D(r)$ is computed from the sine Fourier transform of $\mathcal{Q}(Q)$ using a standard FFT algorithm. As the transform is sine-only, the $Q$-space function, $\mathcal{Q}(Q)$, is mirrored with its odd image and the imaginary components of the Fourier transform used. $\mathcal{Q}(Q)$ is also zero-padded for computational efficiency and interpolation in real-space.

The length of the transformed array, $2^N$, can be set by the user via the Additional Preferences dialog. Increasing $N$ increases the density of points in the real-space array, which may be necessary when using a large $Q$-range, which, for the same value of $N$, would otherwise give a lower real-space resolution.

An option is then provided to optimise the calculated $i(Q)$ and $g(r)$ through the numerical iterative procedure developed by Eggert et al. (2002), that follows from the work of Kaplow et al. (1965). The procedure is based on the assumption that, in real-space, a minimum distance, $r_{\text{min}}$, can be defined, which represents the largest distance ($r = 0$ to $r = r_{\text{min}}$) where no atoms can be found. In a liquid, this corresponds to the region below the first interatomic distance in the $1^\text{st}$ coordination shell. Correspondingly, no oscillations or peaks should be observed in the real-space correlation functions within this region. As a result, the function $D(r) = -4\pi r \rho_0$ for $r < r_{\text{min}}$. However, oscillations are commonly observed in this region, due to the effects of Fourier transforming finite $Q$-space data described above, along with systematic errors in the scattering factors and the normalisation factor, $\alpha$. These errors can be minimised by taking the difference between the observed and modelled behaviour of $D(r)$ below $r_{\text{min}}$. Fourier transforming this difference to reciprocal space, and subtracting from the $i(Q)$ (with a scaling factor applied based on the definition of $S(Q)$ used) to compute a refined $i(Q)$. This refined $i(Q)$ can then be fed back into Eq. 12, and the final refined $i(Q)$ computed after a set number of iterations, $n_{\text{iter}}$. A minimum of 3 iterations is normally required for convergence. The value of $r_{\text{min}}$ used in the refinement should be set carefully, as it exerts a strong influence (Shen et al. 2004). This value should correspond to the minimum preceding the first physical peak in $D(r)$, which is generally obvious in the preliminary $D(r)$ (prior to optimisation), particularly when using a modification function to suppress FT artefacts (i.e., unphysical peaks). In some cases the user may need additional knowledge of their sample or its structure to set the $r_{\text{min}}$ correctly. The iterative procedure is as follows:

1. $D_{(n)}(r) = \frac{2}{\pi} \int_0^{Q_{\text{max}}} \mathcal{Q}(Q) \sin( qr ) dQ$ (12)

2. $\Delta D_{(n)}(r) = D_{(n)}(r) - S_{\infty}(-4\pi \rho_0 r)$, for $r < r_{\text{min}}$ (13)

3. $i_{(n+1)}(Q) = i_{(n)}(Q) - \frac{1}{Q} \left[ \frac{i_{(n)}(Q)}{z(Q)} + 1 \right] \times \int_0^{r_{\text{min}}} \Delta D_{(n)}(r) \sin(Qr)dr$ (14)

where for the Ashcroft-Langreth formalism:

$$z^{\text{AL}}(Q) = S_{\infty} + J_0(Q)$$

(15)

and for the Faber–Ziman formalism:

$$z^{\text{FZ}}(Q) = 1.$$  

(16)

$J_0(Q)$ and $S_{\infty}$ are defined in Eqs. 41 and 42 respectively.

LiquidDiffract also provides the option to use this refinement procedure to extract the average number density of the sample, $\rho_0$. As $\rho_0$ is an independent variable in the refinement procedure its optimum value can be found by minimising a $\chi^2$ figure of merit, defined as a function of $\rho_0$:

$$\chi^2_{(n)}(\rho_0) = \int_0^{r_{\text{min}}} \left[ \Delta D_{(n)}(r) \right]^2 dr.$$ 

(17)

LiquidDiffract supports several solvers implemented in the SciPy libraries to do this minimisation. The solver to be used, along with specific options like convergence criteria and number of function iterations can be set from the Additional Preferences dialog. The solvers currently supported by LiquidDiffract are L-BFGS-B (Byrd et al. 1995; Zhu et al. 1997), SLSQP (Kraft 1988) and COBYLA (Powell 1994, 1998, 2007), which are fast, robust, and represent a good cross-section of the different derivative-free and gradient-based approaches available in the SciPy library. All solvers require upper and lower bounds on the density to be set. The optimisation is typically very fast (~0.05 s on a mid-range laptop), although the accuracy of the result depends on the initial estimate of $\rho_0$ and the range of the limits used. A useful recipe to use if a good initial estimate of $\rho_0$ cannot be made is to first run the optimisation with only 1 iteration in the refinement procedure to extract a better initial estimate.
of $\rho_0$. The optimisation can then be run again using a higher number of iterations that provides full convergence. The bounds on $\rho_0$ can also be progressively narrowed to encourage true convergence. Alternatively, a global minimisation option is provided. This is a basin-hopping algorithm which runs the optimisation several times, randomly perturbing the initial estimate of $\rho_0$ between each run, and accepting or rejecting each minimisation based on the standard Metropolis criterion (Li and Scheraga 1987; Wales and Doye 1997). This option can be useful in some cases of otherwise poor convergence, but is much slower.

The density, $\rho_0$, is the only optimisation parameter in the refinement procedure implemented in LiquidDiffract, however the $\chi^2$ figure of merit can be used to simultaneously refine the background scaling factor, $b$, alongside $\rho_0$ (Eggert et al. 2002). In high-pressure experiments additional care in the background scaling (such as $Q$-dependent scaling functions) is often required, and in these cases naive optimisation of both $\rho_0$ and $b$ can lead to unphysical results. As LiquidDiffract was primarily designed for data collected at extreme conditions, simultaneous refinement of $\rho_0$ and $b$ is not currently implemented, but this is a planned feature for future releases of the software. Nevertheless, as $b$ and $\rho_0$ are not strongly correlated (Eggert et al. 2002), the background can be independently scaled by other methods prior to refinement of $\rho_0$, which is equivalent to a step-wise optimisation across the full $b$-$\rho_0$ parameter space.

It is also important to note that a well defined minimum in $\chi^2_{\text{ew}}(\rho_0)$ relating to the sample density only exists for $n < \Xi$, where $\Xi$ is typically $\sim 10$. This is a result of the way the iterative procedure works, and the definition of $\chi^2_{\text{ew}}(\rho_0)$. After a large number of iterations the refinement procedure can always force the data to closely fit the modelled slope through unreasonable manipulation (e.g., by massively inflating low-$Q$ values). This results in a drastically reduced $\chi^2$ at large values of $n$. Furthermore, at values of $n > \Xi$, $\chi^2_{\text{ew}}(\rho_0)$ will decrease with decreasing $\rho_0$ because the absolute values used to calculate $\chi^2_{\text{ew}}(\rho_0)$ are smaller as the magnitude of the model slope decreases. However, a minimum generally exists at $\rho_0 > 0$ because the iterative procedure struggles to force the $D(r)$ to the horizontal. Computing the function $\chi^2(\rho_0;n)$ illustrates that above $\Xi$ iterations a well defined minimum in $\chi^2(\rho_0)$ no longer exists, and the density region in which the iterative procedure can fit the data rapidly increases (Fig. 2).

The value of $\Xi$ will depend on the data and the value of $r_{\text{min}}$. The change in behaviour of $\chi^2(\rho_0;n)$ above and below $\Xi$ can be illustrated by plotting a map of the function (Fig. 2c). The rapid asymptotic convergence in just a few iterations to the true minimum in $\chi^2$ can be seen, as can the sharp shelf in the function at $n = \Xi$. Above $n = \Xi$ the true density minimum is briefly preserved as a local minimum in the function, with the global minimum at lower $\rho_0$ physically meaningless (Fig. 2b, d). Care must be taken in the density refinement to ensure that $n < \Xi$, as any solver would naively find the physically meaningless minimum at $n > \Xi$. Prior to the $n = \Xi$ limit, the density minimum already begins to be shifted away from the true value of $\rho_0$ by the widening plateau at low $\rho_0$ (Fig. 2d). At high enough $\rho_0$ resolution the number of iterations itself can be optimised, as this is seen as a maximum in the value of $\rho_0$ along the minimum in $\chi^2(\rho_0;n)$ (Fig. 2d). This is a result of the trade-off between the minimum asymptotically approaching the true value of $\rho_0$ with increasing number of iterations, and the shift of the minimum towards lower $\rho_0$ as $n$ approaches $\Xi$. An example script to compute maps of the $\chi^2(\rho_0;n)$ function is provided with the LiquidDiffract source code. Greater values of $n$ can be used to optimise the $S(Q)$ and real-space correlation functions at a set density, but the value of $\chi^2$ used to extract density becomes meaningless.

The next workflow tabs are automatically updated with the resulting PDF functions after the refinement of $i(Q)$ has run, or after the initial calculation of $i(Q)$ if the $S(Q)$ pass-through option is selected. If a modification function is selected the default behaviour is to only apply this to display the initial $i(Q)$ and to calculate the final $g(r)$ after the refinement procedure has completed. An option is provided to use the modification function within the refinement procedure, although this is not recommended for most use cases. Figure 3 shows an example of using the iterative refinement procedure in LiquidDiffract to obtain a properly normalised $S(Q)$ and $g(r)$.

**PDF output**

This tab displays the optimised structure factor, $S(Q)$, pair distribution function, $g(r)$, and radial distribution function, $RDF(r)$, following operations made in the refinement tab. The functions can be inspected within the software, or output to text files for further analysis or graphing by the user. If a modification function has been used in the data treatment then information on this is saved along with the raw $S(Q)$. Information on the sample composition and density is also saved automatically. The $S(Q)$ is defined as in Eqs. 32 and 37, and the $g(r)$ as the finite integral version of equation 2:

$$g(r) - 1 = \frac{1}{2\pi^2 r \rho_0} \int_0^{Q_{\text{max}}} Q i(Q) \sin(Qr) dQ.$$  

(18)

The RDF($r$) describes the average number of particles that can be found within a sphere of radius $r$. It is defined as:

$$RDF(r) = 4\pi r^2 \rho(r) = 4\pi r^2 \rho_0 g(r).$$

(19)

or, by combining with Eq. 18:
Structural information: integration and curve fitting

The final tab provides several methods of extracting average coordination numbers of atomic pairs from the calculated radial distribution functions. Two toolboxes are provided: an *Integration Toolbox* (which is generally used for monatomic samples), and a *Curve-fitting Toolbox* (which is generally used for polyatomic samples).

An average coordination number, $\bar{N}_{\alpha\beta}$, can be defined as the average number of atoms $\beta$ around an atom $\alpha$. This can be calculated by integrating the pair-distribution function, $g(r)$, over a spherical shell defined by an inner and outer radius, $r_1$ and $r_2$, and multiplying by the bulk density:

$$ RDF(r) = \left[ \frac{2r}{\pi} \int_0^{Q_{\text{max}}} Ql(Q) \sin(Qr) dQ \right] + 4\pi r^2 \rho_0. $$

(20)

**Fig. 2**

(a) Function $\chi^2(\rho_0;n)$ computed for sample data at increasing values of $n$. The lack of a well-defined minimum at high values of $n$ can be seen, as can the increasing density region in which the iterative procedure can fit the data, but the sharp change in behaviour at $n \geq \Xi$ is not obvious. The minimum within the widening plateau region at $\rho_0 \sim 0.027$ that exists at high values of $n$ is also not obvious. The local minimum towards $\rho_0 = 0$ becomes the global minimum. Immediately beyond the $n = \Xi$ limit the local minimum is still well-defined enough to find with a solver, given a bound-constrained minimisation that excludes the new global minimum from the search space.

(b) Close up of the function $\chi^2(\rho_0;n)$ across the limit $\Xi = 11$. At $n = 11$ the local minimum towards $\rho_0 = 0$ becomes the global minimum. Immediately beyond the $n = \Xi$ limit the local minimum is still well-defined enough to find with a solver, given a bound-constrained minimisation that excludes the new global minimum from the search space.

(c) Map of the function $\chi^2(\rho_0;n)$ computed for the same sample data. The colour scaling shows the log-scale value of the $\chi^2$ function, so that the minima above and below $n = \Xi$ are obvious, despite the difference in scale. The density minimum persists as a local minimum beyond $n = \Xi$ until $n = 14$, although the minimum begins to be shifted to lower values of $\rho_0$. The optimum number of iterations can be seen as a local maximum in the value of $\rho_0$ along the minimum in $\chi^2(\rho_0;n)$.
\[ N_{ab} = \int_{r_1}^{r_2} 4\pi \rho_0 c_\beta g_{a\beta}(r) r^2 dr, \tag{21} \]

where \( c_\beta \) is the atomic concentration of species \( \beta \) in the sample and \( g_{a\beta}(r) \) is the partial pair distribution function for the \( a-\beta \) pair. For a monatomic system this is equivalent to simply integrating over the RDF(\( r \)). LiquidDiffract provides a toolbox to extract the first coordination number, \( N_1 \), by computing this integral across the first peak in RDF(\( r \)). Three alternative methods are implemented based on commonly used methods in the literature (Waseda 1980).

The first method is based on the assumption that the quantity \( r g(r) \) is symmetrical for a coordination shell about its average position. This is computed by integrating over RDF(\( r \)) from the leading edge of the first peak in \( r g(r) \) (equivalently, \( T(r) \), Eq. 27), \( r'_0 \), to its maximum \( r'_\text{max} \), and doubling the result:

\[ \bar{N}_A = 2 \int_{r'_0}^{r'_\text{max}} 4\pi \rho_0 \left[r g(r)\right]_{\text{sym}} dr. \tag{22} \]

The second method is based on the assumption that the coordination shell is instead symmetrical about a radius which defines the maximum in the \( r^2 g(r) \) curve. This is computed in a similar way, integrating over RDF(\( r \)) from the leading edge of the first peak in \( r^2 g(r) \) (equivalently, the RDF(\( r \)), \( r_0 \)), to its maximum \( r_\text{max} \) and doubling:

\[ \bar{N}_B = 2 \int_{r_0}^{r_\text{max}} 4\pi \rho_0 \left[r^2 g(r)\right]_{\text{sym}} dr. \tag{23} \]

Since the first peak is not truly symmetrical, \( \bar{N}_A \) and \( \bar{N}_B \) can be considered estimates of the lower bound on the average coordination number. Typically, it also the case that \( \bar{N}_A < \bar{N}_B \). This consideration of asymmetry leads to the third
method, which involves integrating $RDF(r)$ from the leading edge of the first peak $r_0$ to first minimum following it, $r_{\text{min}}$:

$$\bar{N}_c = \int_{r_0}^{r_{\text{min}}} 4\pi r^2 g(r) dr.$$  \hfill (24)

An auto-refine button is provided to quickly find the positions of $r_0$, $r_{\text{max}}$, $r_{\text{max}}$, and $r_{\text{min}}$ using a root-finding and minimisation procedure. The user can also set the limits manually via a spinbox or by dragging the limits on the data plot. In particular, careful setting of $r_{\text{min}}$ may be required as its apparent value can fluctuate due to errors in $RDF(r)$, and with changes in $p-T$ conditions (e.g., Yagafarov et al. 2012).

An example of coordination number extraction for a monatomic sample using the Integration Toolbox is illustrated in Fig. 3d.

For polyatomic samples it is more useful to fit the $RDF(r)$ with a number of peaks so that the individual contributions from different $\alpha-\beta$ pairs can be investigated. LiquidDiffract provides a curve-fitting toolbox to do this. For each individual peak attributed to an $\alpha-\beta$ pair the coordination number can be calculated by (Petkov 2012):

$$\bar{N}_{a\beta} = \frac{c_{\beta}}{W_{a\beta}^{\text{x-ray}}} \times \text{Respective RDF Peak Area},$$  \hfill (25)

where $c_\beta$ is the atomic proportion of $\beta$ in the composition, and $W_{a\beta}^{\text{x-ray}}$ is the x-ray weighting factor for the $\alpha-\beta$ pair, given by (Faber and Ziman 1965):

$$W_{a\beta}^{\text{x-ray}} = \frac{c_\alpha c_\beta K_\alpha K_\beta (2 - \delta_{a\beta})}{(\sum \alpha c_\alpha K_\alpha)^2},$$  \hfill (26)

where $c_\alpha$ is the atomic fraction of species $\alpha$, $\delta$ is the Kro- nacker delta ($\delta_{a\beta} = 0$ if $\alpha \neq \beta$ and $\delta_{a\beta} = 1$ if $\alpha = \beta$), and $K_\alpha$ is the effective atomic number of species $\alpha$ given by the Warren–Krutter–Morningstar approximation specifically at $Q = 0$ (an option is provided in the Additional Preferences dialog to use the data $Q$-range instead; Eq. 43; Warren et al. 1936).

It is generally preferable to fit the $T(r)$ instead of $RDF(r)$ as the peaks are symmetrically broadened and gaussian in shape (Waseda 1980), with:

$$T(r) = \frac{RDF(r)}{r}.$$  \hfill (27)

The gaussian peak function is defined as:

$$\text{Gauss}(r) = \frac{A}{\sigma \sqrt{2\pi}} \exp \left[ \frac{-(r - \mu)^2}{2\sigma^2} \right]$$  \hfill (28)

with $\mu$ the peak centre, $\sigma$ describing the width of the peak, and $A$ the area under the curve. In many cases prior knowledge of the coordination number or average bond length for an $\alpha-\beta$ pair may be available so it is more useful to redefine the peak function with $N_{a\beta}$ as a fitting parameter.

Combining Eq. 28 with Eq. 25 and the definition of $T(r)$ from Eq. 27 we can find:

$$T(r) = \sum \limits_{a\beta} \left[ \frac{\bar{N}_{a\beta} W_{a\beta}^{\text{x-ray}}}{c_p \sigma_{a\beta} \sqrt{2\pi}} \exp \left( \frac{-(r - r_{a\beta})^2}{2\sigma_{a\beta}^2} \right) \right] \times \left[ 1 + \text{erf} \left( \frac{r - r_{a\beta}}{\sigma_{a\beta}} \right) \right]$$  \hfill (29)

with the peak centre, $\mu$, equivalent to the average $\alpha-\beta$ bond length, $r_{a\beta}$, and $\sigma_{a\beta}$ a measure of the distribution of interatomic distances arising from thermal and static structural disorder.

In LiquidDiffract the user can add an arbitrary number of peaks and fix or refine any combination of the parameters $(\bar{N}_{a\beta}, r_{a\beta}$ and $\sigma_{a\beta})$ for each peak individually or simultaneously. The data range to fit can also be selected by the user, and residuals are plotted alongside the fits.

In some cases the peaks in the radial distribution function may have significant asymmetry. Sukhomlinov and Müser (2017, 2020) noted that for some samples and conditions the interatomic distances may form a skewed distribution, and that a skew-normal distribution provides a closer fit to the $RDF(r)$ than a pure gaussian. The skew-normal distribution is a pure gaussian multiplied by a skewness coefficient defined as:

$$1 + \text{erf} \left( \frac{\xi - \mu}{\sigma \sqrt{2}} \right)$$  \hfill (30)

where $\xi$ is an additional fitting parameter describing the skew. By analogy with Eq. 29 we can then instead fit:

$$T(r) = \sum \limits_{a\beta} \left[ \frac{\bar{N}_{a\beta} W_{a\beta}^{\text{x-ray}}}{c_p \sigma_{a\beta} \sqrt{2\pi}} \exp \left( \frac{-(r - r_{a\beta})^2}{2\sigma_{a\beta}^2} \right) \right] \times \left[ 1 + \text{erf} \left( \frac{\xi_{a\beta} - r_{a\beta}}{\sigma_{a\beta} \sqrt{2}} \right) \right]$$  \hfill (31)

When $\xi_{a\beta} = 0$ Eq. 31 is identical to Eq. 29. LiquidDiffract provides the option to individually toggle this additional skewness parameter for any of the peaks in the fit.

It is important to note that naively fitting multiple Gaussians to the $RDF(r)$ or $T(r)$ can sometimes lead to unphysical results (Cristiglio et al. 2009; Petkov 2012). Peaks must be correctly assigned to specific pair correlations, which requires additional structural information. Strongly overlapping peaks may also result in non-unique solutions to the fit, and additional structural constraints from theory, modelling, or complementary experimental data (such as neutron diffraction with isotope substitution or extended x-ray absorption fine structure spectroscopy) may be required to...
fix one or more values in the fit and obtain sensible estimates of the remaining structural parameters (Fig. 4b; Cristiglio et al. 2009). As $N_{ab}$ and $r_{ab}$ are both fitting parameters in the Curve-fitting Toolbox, and any combination of parameters can be fixed or refined, this process is simple in LiquidDiffRACT. Nevertheless, structural parameters of multiple correlations can still often be extracted directly from partially overlapping peaks without prior knowledge of any parameters (Fig. 4a; Petkov 2012). An example of using the Curve-fitting Toolbox to extract structural parameters from the $T(r)$ functions of polyatomic samples is shown in Fig. 4.

Further development

The features and capabilities of LiquidDiffRACT described here relate to version 1.1.8 of the software. Planned features include the option to use neutron scattering data, improvements to the Curve-fitting Toolbox such as convolution with modification functions to account for FT artifacts in the RDF($r$), support for saving data options set in the GUI to a configuration file, and better support for working with multiple data sets simultaneously. LiquidDiffRACT also assumes there is no intramolecular contribution to the $D(r)$, but this is only strictly true for some compositions (e.g. monatomic metallic liquids; Eggert et al. 2002). Support for including intramolecular forces may be added in future releases however. The current version of LiquidDiffRACT does not implement simultaneous refinement of the background scaling factor, $b$ along with density, which is a method commonly employed in the literature (Eggert et al. 2002). LiquidDiffRACT was primarily designed for analysis of data collected in high-pressure experiments. In these cases additional care in the background scaling is often required, and naive optimisation of $b$ along with density can lead to unphysical results. Nevertheless, this is a planned feature for future releases of LiquidDiffRACT, as is support for more complex background scaling such as $Q$-dependent background corrections. We also welcome feature requests or bug reports via the github issue tracker.

**Fig. 4** Example usage of the Curve-fitting Toolbox to extract structural information in LiquidDiffRACT. Gaussian fits (red dashed lines) to the refined $T(r)$ (grey lines) of two samples are shown: a CaSiO$_3$ glass and b MgSiO$_3$ liquid. The blue and green lines show the individual gaussians that contribute to the summed models. Data was obtained from *in situ* synchrotron X-ray diffraction with aerodynamic levitation and CO$_2$ laser-heating at beamline ID11 of the European Synchrotron Radiation Facility (ESRF), Grenoble, France. a The $T(r)$ of CaSiO$_3$ glass at ambient conditions was calculated by LiquidDiffRACT with $\rho_0 = 0.0749$ atoms/Å$^3$ (2.89 g/cm$^3$). The correlation function was refined using the iterative procedure (with $Q_{\text{min}} = 1.25$, $Q_{\text{max}} = 26$, $r_{\text{min}} = 1.25$ and $n_{\text{iter}} = 1000$) and a Lorch function was applied prior to the final Fourier transform. The first two peaks in the $T(r)$ can be attributed to the nearest-neighbour Si–O and Ca–O correlations respectively, with a high-$r$ contribution from O–O correlations overlapping the second peak (Waseda and Toguri 1977; Taniguchi et al. 1997; Benmore et al. 2010). As the Si–O and Ca–O correlations do not overlap significantly, both peaks could be fitted simultaneously and all parameters allowed to vary in the fit. This yielded parameters for the Si–O correlation of $N_{\text{Si–O}} = 4.00 \pm 0.01$, $r_{\text{Si–O}} = 1.624 \pm 0.001$ and $\sigma_{\text{Si–O}} = 0.110 \pm 0.001$ ($\xi_{\text{Si–O}} = 0$), and parameters for the Ca–O correlation of $N_{\text{Ca–O}} = 6.06 \pm 0.04$, $r_{\text{Ca–O}} = 2.367 \pm 0.001$ and $\sigma_{\text{Ca–O}} = 0.157 \pm 0.001$ ($\xi_{\text{Ca–O}} = 0$). The obtained structural parameters are consistent with previously reported values derived from both simulations (Mead and Mountjoy 2006; Lan et al. 2017b) and experimental studies (Yin et al. 1986; Eckersley et al. 1988; Gaskell et al. 1991; Taniguchi et al. 1997; Benmore et al. 2010; Skinner et al. 2012; Salmon et al. 2019). b The $T(r)$ of MgSiO$_3$ liquid at ambient pressure was calculated by LiquidDiffRACT at $\rho_0 = 0.07816$ atoms/Å$^3$ (2.606 g/cm$^3$), with $\rho_0$ extracted using the density refinement feature. The refined density is consistent with previously reported estimates (Nomura et al. 2017; Kim et al. 2019). The correlation function was refined using the iterative procedure (with $Q_{\text{min}} = 0.3$, $Q_{\text{max}} = 24$, $r_{\text{min}} = 1.34$ and $n_{\text{iter}} = 1000$) and a Lorch function was applied prior to the final Fourier transform. The first two peaks in the $T(r)$ are attributed to the nearest-neighbour Si–O and Mg–O correlations respectively (Waseda and Toguri 1977; Wilding et al. 2008). The Si–O coordination number was fixed to 4.0 in the fit (Wilding et al. 2008) due to the overlapping nature of the two peaks. A simultaneous fit of the remaining structural parameters yielded $r_{\text{Si–O}} = 1.651 \pm 0.001$, $\sigma_{\text{Si–O}} = 0.132 \pm 0.001$ ($\xi_{\text{Si–O}} = 0$), $N_{\text{Mg–O}} = 5.08 \pm 0.06$, $r_{\text{Mg–O}} = 2.114 \pm 0.003$ and $\sigma_{\text{Mg–O}} = 0.232 \pm 0.003$ ($\xi_{\text{Mg–O}} = 0$). These structural parameters are also consistent with previously reported values derived from both simulations (Lan et al. 2017a; Kim et al. 2019) and experimental studies (Funamori et al. 2004; Wilding et al. 2008).
Appendix A: Total structure factor calculations

LiquidDiffract implements both the Faber-Ziman (Faber and Ziman 1965) and Ashcroft-Langreth (Ashcroft and Langreth 1967) formalisms of the total molecular structure factor, $S(Q)$. In both cases the $S(Q)$ is converted to normalised (atomic) units following the method of Krogh-Moe (1956) and Norman (1957). In both formalisms the scattering intensity from the sample, $I_{\text{Sample}}(Q)$, is first extracted from the measured intensity by subtracting a background pattern with a scaling factor applied as in Eq. 4.

Faber-Ziman formalism

The Faber-Ziman total structure factor is defined as:

$$S_{\text{FZ}}(Q) = \left( I_{\text{Coh}}(Q) - \langle f^2 \rangle \right) / \langle f \rangle^2.$$  

(32)

In the above equation $I_{\text{Coh}}(Q)$ is the coherent scattering from the sample, defined as:

$$I_{\text{Coh}}(Q) = a_{\text{FZ}} I_{\text{Sample}}(Q) - \frac{1}{N} \sum_p f_{\text{incoh}}(Q),$$  

(33)

where $a_{\text{FZ}}$ is the Krogh-Moe-Norman normalisation factor that converts the signal into atomic units, $N$ is the total number of atoms in the compositional unit, and $\sum_p f_{\text{incoh}}(Q)$ is the sum of the $Q$-dependent incoherent (Compton) scattering signals from each atom $p$ in the compositional unit. In LiquidDiffract $I_{\text{incoh}}(Q)$ is interpolated from the analytical tabulations of Hubbell et al. (1975). The normalisation factor $a_{\text{FZ}}$ is defined as:

$$a_{\text{FZ}} = -2\pi^2 \rho_0 + \frac{\int_0^{Q_{\text{max}}} \sum_p f_{\text{incoh}}(Q) + \langle f^2 \rangle}{\int_0^{Q_{\text{max}}} \langle f \rangle^2} \frac{Q^2 dQ}{\langle f \rangle^2},$$  

(34)

where $Q_{\text{max}}$ is the maximum scattering momentum and $\rho_0$ is the average atomic number density of the sample. The $Q$-dependent functions $\langle f^2 \rangle$ and $\langle f \rangle^2$ are the average scattering functions:

$$\langle f^2 \rangle = \frac{1}{N} \sum_p (f_p(Q))^2$$  

(35)

and

$$\langle f \rangle^2 = \frac{1}{N^2} \sum_p \sum_q f_p(Q) f_q(Q).$$  

(36)

where $f_p(Q)$ are the atomic form factors for each atom $p$ in the compositional unit. The atomic form factors are taken from the analytic tabulation published in the International Tables of Crystallography (Brown et al. 2006). For a monatomic sample $\langle f^2 \rangle = \langle f \rangle^2$.

Ashcroft-Langreth formalism

The Ashcroft-Langreth total structure factor is defined as:

$$S_{\text{AL}}(Q) = \frac{f_{\text{Coh}}(Q)}{NZ_{\text{tot}}^2 f_s^2(Q)},$$  

(37)

where $N$ is the number of atoms in the compositional unit, $Z_{\text{tot}}$ is the total atomic number of the compositional unit (i.e. $\sum_p Z_p$), and $f_s(Q)$ is an effective electronic form factor. The coherent scattering from the sample $I_{\text{Coh}}(Q)$ is here defined as:

$$I_{\text{Coh}}(Q) = N \left[ a_{\text{AL}} I_{\text{Sample}}(Q) - \sum_p f_{\text{incoh}}(Q) \right].$$  

(38)

The incoherent scattering signals are calculated in the same way as Eq. 33, but here the normalisation factor $a_{\text{AL}}$ is defined as:

$$a_{\text{AL}} = Z_{\text{tot}}^2 \frac{-2\pi^2 \rho_0 + \int_0^{Q_{\text{max}}} [J(Q) + S_\infty] Q^2 dQ}{\int_0^{Q_{\text{max}}} f_{\text{Sample}}(Q) Q^2 dQ}.$$  

(39)

The effective electronic form factor for the sample, $f_s(Q)$, is defined as:

$$f_s(Q) = \frac{\sum_p f_p(Q)}{Z_{\text{tot}}^2}.$$  

(40)

The function $J(Q)$ is then defined by:

$$J(Q) = \frac{\sum_p f_{\text{incoh}}(Q)}{Z_{\text{tot}}^2 f_s^2(Q)}$$  

(41)

and the value $S_\infty$ by:

$$S_\infty = \frac{\sum_p K_p^2}{Z_{\text{tot}}^2},$$  

(42)

where $K_p$ is the Warren–Krutter–Morningstar approximation for the effective atomic number of each atom (Warren et al. 1936). This is calculated by:
\[ K_p = \left\{ \frac{f_0(Q)}{f_0(Q)} \right\}_Q \tag{43} \]

where the angled brackets denote an average over the \( Q \)-range of the data.

**Re-scaling of functions in the Ashcroft-Langreth formalism**

The Faber-Ziman \( S(Q) \) is normalised to unity at infinite \( Q \) (i.e., \( \lim_{Q \to \infty} S_{\text{FZ}}(Q) = 1 \)), but for the Ashcroft-Langreth formalism \( \lim_{Q \to \infty} S_{\text{AL}}(Q) = S_\infty \). For monatomic samples the value \( S_\infty = 1 \), but for multi-component (polyatomic) samples \( 0 < S_\infty < 1 \). This means that the \( g(r) \) approaches \( 1 - S_\infty \) rather than \( 0 \) at \( r = 0 \), and the \( D(r) \) and \( \text{RDF}(r) \) are similarly scaled. An option is provided in the Additional Preferences dialog to either plot the Ashcroft-Langreth \( S(Q) \) and \( g(r) \) as calculated, or re-scaled to match the conventional scaling of the Faber-Ziman functions. The re-scaling of the \( \text{RDF}(r) \) is always performed to allow the same fitting routines to be used in the structural information tab.

The re-scaled functions are calculated by:

\[ S'(Q) = \frac{S_{\text{AL}}(Q)}{S_\infty} \tag{44} \]

\[ g'(r) = \frac{g_{\text{AL}}(r) - 1}{S_\infty} + 1 \tag{45} \]

\[ \text{RDF}'(r) = \frac{\text{RDF}_{\text{AL}}(r) - 4\pi \rho_0 r^2}{S_\infty} + 4\pi \rho_0 r^2. \tag{46} \]

**Acknowledgements** We would like to thank two anonymous reviewers, whose careful comments helped improve and clarify this manuscript. We are grateful to Louis Hennet (CEMHTI, Orléans) and Daniel Neuvillé (IPGP, Paris) for their assistance in collecting the synchrotron X-ray diffraction measurements of liquid MgSiO\(_3\) and CaSiO\(_3\), glass (proposal No. HD517) used to generate Fig. 4. We would also like to thank Oliver Lord for his support and useful discussions during the completion of this work.

**Funding** BJH would like to acknowledge support in the form of a studentship awarded as part of a Royal Society Grant (No. RG160631). JD acknowledges funding from the EPSRC (standard grant EP/V001736/1).

**Declarations**

**Conflict of interest** The authors have no conflicts of interest to declare.

**Code availability** The LiquidDiffrac source code is available from a public github repository (https://github.com/bjheinen/LiquidDiffrac), and releases are distributed via PyPi (https://pypi.org/project/LiquidDiffrac/). LiquidDiffrac v1.1.8 can be directly accessed at https://doi.org/10.5281/zenodo.5796187.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

**References**

Ashcroft NW, Langreth DC (1967) Structure of binary liquid mixtures. i. Phys Rev 156(3):685

Benmore CJ, Weber JKR, Wilding MC, Du J, Parise JB (2010) Temperature-dependent structural heterogeneity in calcium silicate liquids. Phys Rev B 82(22):224202

Bernandes CES (2021) STRFACT: Determining x-ray total scattering factors from molecular dynamics simulation data. https://doi.org/10.5281/zenodo.4762008

Blackman RB, Tukey JW (1958) The measurement of power spectra from the point of view of communications engineering - part i. Bell Syst Tech J 37(1):185–282

Briggs R, Gorman MG, Zhang S, McGonegle D, Coleman AL, Coppari F, Morales-Silva MA, Smith RF, Wicks JK, Bolme CA, Gleason AE, Cunningham E, Lee HJ, Nagler B, McMahon MI, Eggert JH, Fratanduono DE (2019) Coordination changes in liquid tin under shock compression determined using in situ femtosecond x-ray diffraction. Appl Phys Lett 115(26):264101

Brown PJ, Fox AG, Maslen EN, O’Keefe MA, Willis BTM (2006) Intensity of diffracted intensities. In: Prince E (ed) International Tables for Crystallography Volume C: Mathematical, physical and chemical tables, Springer, Dordrecht, pp 554–595. https://doi.org/10.1017/9780521694991.900

Byrd RH, Lu P, Nocedal J, Zhu C (1995) A limited memory algorithm for bound constrained optimization. SIAM J Sci Comput 16(5):1190–1208

Cristiglione V, Cuello GJ, Piarristeguy AA, Pradel A (2009) The coordination number calculation from total structure factor measurements. J Non-Cryst Solids 355(37–42):1811–1814

Dimitroulis C, Raptis T, Raptis V (2015) Polycla - a tool for the calculation of molecular radial distribution functions based on molecular dynamics trajectories. Comput Phys Commun 197:220–226

Drewitt JWE (2021) Liquid structure under extreme conditions: high-pressure x-ray diffraction studies. J Phys Condensed Matter In press

Drewitt JWE, Sanloup C, Bytchkov A, Brassamin S, Hennon L (2013) Structure of \( \text{Fe}_{x}\text{Ca}_{(1-x)}\)\(_{\text{SiO}_3}\) liquids and glasses from high-energy x-ray diffraction: Implications for the structure of natural basaltic magmas. Phys Rev B 87:224201. https://doi.org/10.1103/PhysRevB.87.224201

Drewitt JWE, Turci F, Heinen BJ, Macleod SG, Qin F, Kleppe AK, Lord OT (2020) Structural ordering in liquid gallium under extreme conditions. Phys Rev Lett 124(14):145501

Eckersley MC, Gaskell PH, Barnes AC, Chieux P (1988) Structural ordering in a calcium silicate glass. Nature 335(6190):525–527

Eggert JH, Weck G, Loubeyre P, Mezouar M (2002) Quantitative structure factor and density measurements of high-pressure fluids in diamond anvil cells by x-ray diffraction: Argon and water. Phys Rev B 65(17):174105
Cournapeau D, Wieser E, Taylor J, Berg S, Smith JR, Kern R, Picus M, Hoyer S, van Kerkwijk MH, Brett M, Haldane A, del Rio IF, Wiebe M, Peterson P, Gérard-Marchant P, Sheppard K, Reddy T, Weckesser W, Abbasi H, Gohlke C, Oliphant TE (2020) Array programming with NumPy. Nature 585(7825):357–362. https://doi.org/10.1038/s41586-020-2649-2

Hubbell JH, Veigele WJ, Barnes AC, Chieux P (1991) Medium-Faber TE, Ziman JM (1965) A theory of the electrical properties of liquid metals: ii. the resistivity of binary alloys. Phil Mag 11(109):153–173

Funamori N, Yamamoto S, Yagi T, Kikegawa T (2004) Exploratory studies of silicate melt structure at high pressures and temperatures by in situ x-ray diffraction. J Geophys Res Solid Earth 109(B3)

Gaskell PH, Eckersley MC, Barnes AC, Chieux P (1991) Medium-range order in the cation distribution of a calcium silicate glass. Nature 350(6320):675–677

Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, Kern R, Picus M, Hoyer S, van Kerkwijk MH, Brett M, Haldane A, del Rio IF, Wiebe M, Peterson P, Gérard-Marchant P, Sheppard K, Reddy T, Weckesser W, Abbasi H, Gohlke C, Oliphant TE (2020) Array programming with NumPy. Nature 585(7825):357–362. https://doi.org/10.1038/s41586-020-2649-2

Hubbell JH, Veigele WJ, Briggs EA, Brown RT, Cromer DT, Howerton RJ (1975) Atomic form factors, incoherent scattering functions, and photon scattering cross sections. J Phys Chem Ref Data 4(3):471–538

Juhás P, Davis T, Farrow CL, Billinge SJ (2013) Pdfgetx3: a rapid and highly automatizable program for processing powder diffraction data into total scattering pair distribution functions. J Appl Crystallogr 46(2):560–566

Kaplow R, Strong SL, Averbach BL, Kaplow R, Strong SL, Averbach BL (1965) Radial density functions of liquid metals: II. the resistivity of binary alloys. Phil Mag 11(109):153–173

Katagiri K, Ozaki N, Ohmura S, Albertazzi B, Hironaka Y, Inubushi Y, Ishida K, Koenig M, Miyanishi K, Nakamura H et al (2021) Liquid structure of talc-um under internal negative pressure. Phys Rev Lett 126(17):175503

Keen DA (2001) A comparison of various commonly used correlation functions for describing total scattering. J Appl Crystallogr 34(2):172–177

Kim YH, Yi YS, Kim HI, Chow P, Xiao Y, Shen G, Lee SK (2019) Structural transitions in mgso3 glasses and melts at the cermantle boundary observed via inelastic x-ray scattering. Geophys Res Lett 46(23):13756–13764

Kitaigorodskiy A, Chomet S (1967) Elements of order in disorder. In: Chomet S (ed) Order and disorder in the world of mantle. Versuchsanstalt für Luft- und Raumfahrt (DFVLR) - Institut für Dynamik der Flugsysteme, Köln, Germany

Krog-Moe J (1956) A method for converting experimental x-ray diffraction intensities to an absolute scale. Acta Crystallogr A 9(11):951–953

Lan MT, Duong TT, Huy NV, Van Hong N (2017a) Network structure of si2o and mgso3 in amorphous and liquid states. Materials Research Express 4(3):035202

Lan MT, Duong TT, Itatka T, Van Hong N (2017b) Computer simulation of casio3 glass under compression: Correlation between si–si pair radial distribution function and intermediate range order structure. Mater Res Express 4(6):065201

Lanczos C (1966) Discourse on Fourier Series. Oliver & Boyd, Edinburgh

Le Roux S, Petkov V (2010) Isaacs-interactive structure analysis of amorphous and crystalline systems. J Appl Crystallogr 43(1):181–185

Li Z, Scheraga HA (1987) Monte carlo-minimization approach to the multiple-minima problem in protein folding. Proc Natl Acad Sci 84(19):6611–6615

Lorch E (1969) Neutron diffraction by germania, silica and radiatiion-damaged silica glasses. J Phys C: Solid State Phys 2(2):229–237

Mead RN, Mountjoy G (2006) A molecular dynamics study of the atomic structure of (cao)x(sio2)1–x glasses. J Phys Chem B 110(29):14273–14278

Michaud-Agrawal N, Denning EJ, Woolf TB, Beckstein O (2011) Mdanalysis: a toolkit for the analysis of molecular dynamics simulations. J Comput Chem 32(10):2319–2327

Newville M, Stensitzki T, Allen DB, Ingargiola A (2014) Lmint: Non-linear least-square minimization and curve-fitting for python, https://doi.org/10.5281/zenodo.11813

Nomura R, Zou Y, Iriune T (2017) Melting phase relations in the mgso3-casio3 system at 24 gpa. Prog Earth Planet Sci 4(1):1–11

Norman N (1957) The fourier transform method for normalizing intensities. Acta Crystallogr A 10(5):370–373

Petkov V (1989) Rad, a program for analysis of x-ray diffraction data from amorphous materials for personal computers. J Appl Crystallogr 22(4):387–389

Petkov V (2012) Pair distribution functions analysis. In: Characterization of Materials, Wiley, pp 1361–1372. https://doi.org/10.1002/0471266965.com159

Powell MJ (1998) Direct search algorithms for optimization calculations. Acta Numer 7:287–336

Powell MJD (1994) A direct search optimization method that models the objective and constraint functions by linear interpolation. In: Gomez S, Hennar JP (eds) Advances in Optimization and Numerical Analysis, Springer, Netherlands, pp 51–67. https://doi.org/10.1007/978-94-015-8330-5_4

Powell MJD (2007) A view of algorithms for optimization without derivatives. Math Today Bull Inst Math Appl 43(5):170–174

Prescher C (2017) Glassure: An api and gui program for analyzing angular dispersive total x-ray diffraction data. https://doi.org/10.5281/zenodo.88086

Rodriguez I, Valladares RM, Valladares A, Hinojosa-Romero D, Santiago U, Valladares AA (2021) Correlation: An analysis tool for liquids and for amorphous solids. J Open Source Softw 6(65):2976. https://doi.org/10.21105/joss.02976

Salmon PS, Moody GS, Ishii Y, Pizzy JK, Polidori A, Salanne M, Zedler A, Buscemi M, Fischer HE, Bull CL, Klotz S, Weber R, Benmore CJ, MacLeod SG (2019) Pressure induced structural transformations in amorphous mgso3 and casio3. J Non-Crystal Solids: X 3

Soper A (2017) Gudrun - routines for reducing total scattering data. https://www.isis.stfc.ac.uk/Pages/Gudrun.aspx

Sukhomlinov SV, Müser MH (2017) Determination of accurate, mean bond lengths from radial distribution functions. J Chem Phys 146(2):024506

Sukhomlinov SV, Müser MH (2020) Erratum. J Chem Phys 152(4):049902

Taniguchi T, Okuno M, Matsumoto T (1997) X-ray diffraction and exafs studies of silicate glasses containing mg, ca and ba atoms. J Non-Cryst Solids 211(1–2):56–63

Van Der Spoel D, Lindahl E, Hess B, Groenhof G, Mark AE, Berendsen HJ (2005) Gromacs: fast, flexible, and free. J Comput Chem 26(13):1424–1425

Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, Burovski E, Peterson P, Weckesser W, Bright J, van der Walt SJ, Brett M, Wilson J, Millman KJ, Mayorov N, Nelson ARJ, Jones E, Kern R, Larson E, Carey CJ, Polat I, Feng Y, Moore EW,
VanderPlas J, Laxalde D, Perktold J, Cimrman R, Henriksen I, Quintero EA, Harris CR, Archibald AM, Ribeiro AH, Pedregosa F, van Mulbregt P, SciPy 10 Contributors (2020) SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nat Methods 17:261–272. https://doi.org/10.1038/s41592-019-0686-2

Wales DJ, Doye JPK (1997) Global optimization by basin-hopping and the lowest energy structures of lennard-jones clusters containing up to 110 atoms. J Phys Chem A 101(28):5111–5116

Warren BE, Krutter H, Morningstar O (1936) Fourier analysis of x-ray patterns of vitreous SiO2 and B2O2. J Am Ceram Soc 19(1–12):202–206

Waseda Y (1980) The structure of non crystalline materials: liquids and amorphous solids. McGraw-Hill, New York

Waseda Y, Suzuki K (1972) Structure factor and atomic distribution in liquid metals by x-ray diffraction. Phys Status Solid (b) 49(1):339–347

Waseda Y, Toguri J (1977) The structure of molten binary silicate systems CaO-SiO2 and MgO-SiO2. Metall Trans B 8(4):563–568

Wilding MC, Benmore CJ, Weber JR (2008) In situ diffraction studies of magnesium silicate liquids. J Mater Sci 43(14):4707–4713

Yagafarov OF, Katayama Y, Brazhkin VV, Lyapin AG, Saitoh H (2012) Energy dispersive x-ray diffraction and reverse monte carlo structural study of liquid gallium under pressure. Phys Rev B 86(17):174103

Yin CD, Okuno M, Morikawa H, Marumo F, Yamanaka T (1986) Structural analysis of casio3 glass by x-ray diffraction and raman spectroscopy. J Non-Cryst Solids 80(1–3):167–174

Yu T, Prescher C, Ryu YJ, Shi F, Greenberg E, Prakapenka V, Eng P, Stubbs J, Kono Y, Shen G, Watson H, Rivers ML, Sutton SR, Wang Y (2019) A paris-edinburgh cell for high-pressure and high-temperature structure studies on silicate liquids using monochromatic synchrotron radiation. Minerals 9(11):715

Zhu C, Byrd RH, Lu P, Nocedal J (1997) Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. ACM Trans Math Softw (TOEMS) 23(4):550–560

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.