

PyQENS

A python toolbox for fitting QENS Data

Christian Beck^{*1,2}

¹Institut für Angewandte Physik, Universität Tübingen, Auf der
Morgenstelle 10, 72076 Tübingen, Germany

²Institut Max von Laue - Paul Langevin, 71 avenue des Martyrs,
38042 Grenoble, France

May 27, 2020

^{*}ORCID: 0000-0001-7214-3447; mail: christian.beck@uni-tuebingen.de

Contents

1	Introduction	3
2	Software Requirements	3
3	A general comment on fit complexity	3
4	TODO	4
5	The main script	5
5.1	The Protocol	5
6	Experiment Class	6
7	Sample Class	7
8	Data Class	8
9	The Fit Class	10
9.1	The Resolution function	10
9.2	The Fit levels for QENS spectra	10
9.3	The different implemented fit models	10
9.4	Solvent handling	12
9.4.1	Solvent treatment	12
9.4.2	Solvent rescaling	12
9.4.3	Solvent modeling	13
10	Plot and saving options	14
	Appendix	15
A	Overview of flags to control script	16
B	Index	17
C	Glossary	17
D	Literature	17

1 Introduction

This document is a manual for the python routines called **PyQENS**. The routines are used for analyzing quasielastic neutron scattering (QENS) spectra of samples containing diffusive processes such as protein samples (powders, solutions), as well as colloidal suspensions, etc.

The software is written in such a way, that it can treat spectra from different instruments including neutron back-scattering (NBS) as well as time of flight (TOF) measurements. It can also treat fixed window scan (FWS) with several analysis methods or can include these measurements in the analysis of the full QENS spectra to improve the fits.

2 Software Requirements

The **PyQENS** routines are written for Python 3. Some parts of the routines are based on Mantid [1]. To have full functionality of the routines, it is necessary, that the Python version matches the one of Mantid. It should also be possible to run the scripts in the Mantid-workbench [To be tested!!](#). The main part of the scripts is however thought as standalone and can be run without a Mantid installations.

3 A general comment on fit complexity

The routines of **PyQENS** use fits in two different ways. The first type of fits use an excess of fit parameters to obtain a description of the data set which matches as good as possible. This can be used in the case of the resolution function (see Section 9.1) [5] or might also be applied in the case of the solvent. These parameters might have big uncertainties and are normally not linked directly to physical meaningful parameters. However, since the whole parameter set is fixed to these constants in later fits, an accurate description can result in improved fit results at the end.

4 TODO

savepath manual binning

5 The main script

5.1 The Protocol

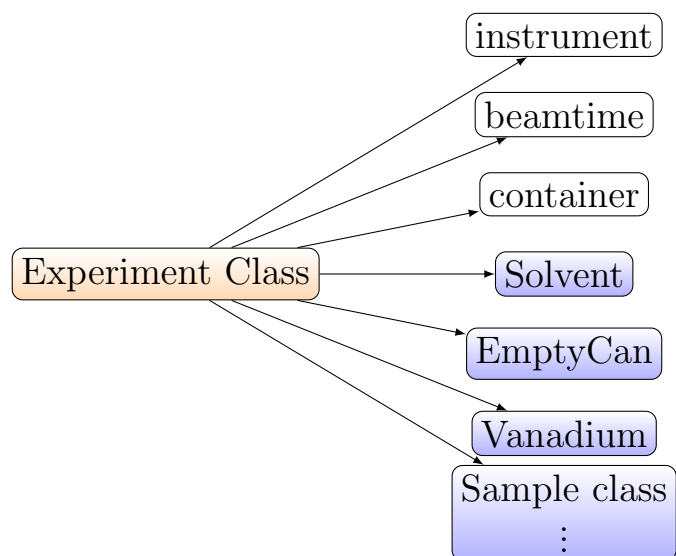


Figure 1: Graphical overview of the structure of the Experiment class

6 Experiment Class

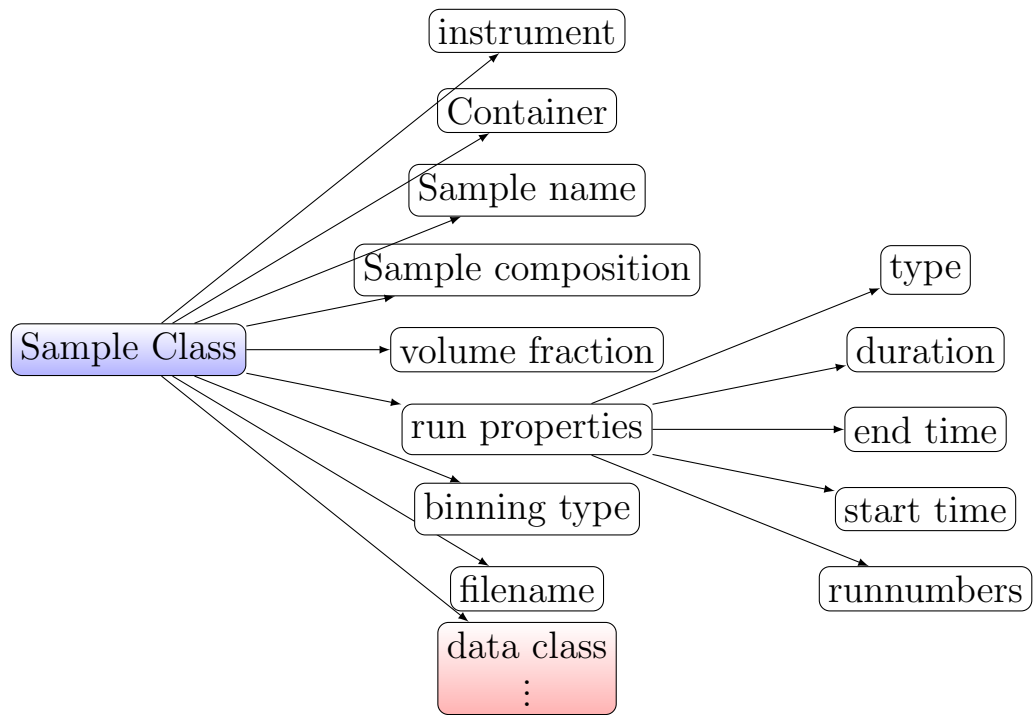


Figure 2: Graphical overview of the structure of the sample class

7 Sample Class

8 Data Class

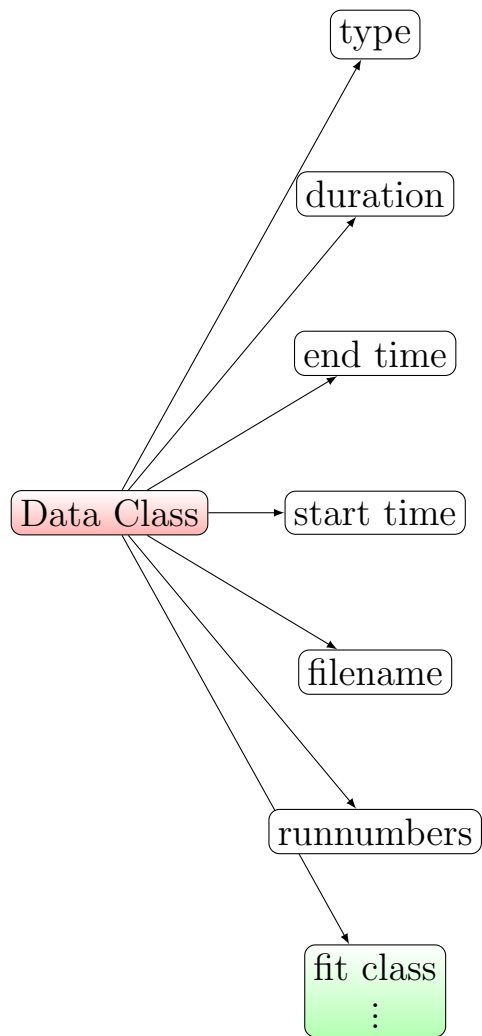


Figure 3: Graphical overview of the structure of the data class

9 The Fit Class

9.1 The Resolution function

Normally, one is interested in describing the resolution function of the instrument as accurate as possible. Since in most of the cases, an accurate physical description describing all of the features of the instrumental resolution function is missing, an adapted parametrization is often chosen. See also Section 3. In the routines of **PyQENS**, the resolution function can be fitted with a combination of

- Gaussian functions (`VanadiumData.NumberOfGaussians=i`),
- Lorentzian functions (`VanadiumData.NumberOfLorentzians=j`) as well as
- Voigt functions (`VanadiumData.NumberOfVoigts=k`)

with `i,j,k` being integers. The widths and scalings are set to be positive values. The center is free and can shift. By doing so, also asymmetric resolution functions can be treated [5]. It can also include flat or sloped backgrounds (`VanadiumData.bkg="slope/flat/none"`).

9.2 The Fit levels for QENS spectra

The routines of **PyQENS** are able to treat different types of fits. What type of fits are performed depends on the data as well as on the fit model. For QENS data, the routines recognize on the given fit model, what fit-level should be applied.

1. As a first step, fits performed for each q individually are recommended (Fitlevel q).
2. As a second step, fits including the q dependence can be performed. These fits, fitting each sample individually (Fitlevel S), can include q -independent parameters such as *e.g.* fraction of immobile particles in the sample [3].
3. As a last step, fits can be performed, which take several QENS spectra of one Experiment (Fitlevel E) into account. By doing so, fits determining parameters, which are independent from the control parameters (*e.g.* time, pressure, temperature, ...) can be determined. In addition, this fit approach can be used to analyzed kinetically changing samples which were investigated by FWS.

9.3 The different implemented fit models

A set of different fit models is implemented for the analysis of the QENS spectra. Here, the flags to use them as well as the model functions and main references are reported. An overview of different applications can be found *e.g.* in [7]. In case, other fit models are desired, they can be written manually in

the file `fitfunctions.py` by respecting the corresponding syntax or the function `addfitfunction.py` to be written! can be use to add the fit model via an graphical interface. If a data set is fitted several times, the results are save in a list of Fit Classes. To perform fits, the function `Fit.QENS` is called with the data set as well as with the corresponding `fit name`:

1. `Vanadium`: This should be used to fit the resolution function of the instrument. The fit function is assembled based on the parameters set as described in Section 9.1.
2. `D2O`: This fit used the full width at half maximum (FWHM) `check!!!` of `IN6check!!!` measurements to fix the width of the solvent (D_2O). It is fitted for each q independently to determine the absolute values. See also Point 2 in Section 9.4.3.
3. `SolventOneLorentz`:
4. `SolventTwoLorentz`:
5. `twolorentz_free_q`: This model fits the data on each q independently:

$$S(q, \omega) = A_0(q) \mathcal{L}_\gamma(\omega) + (1 - A_0(q)) \mathcal{L}_{\gamma+\Gamma}(\omega) \quad (1)$$

with A_0 being the elastic incoherent structure factor (EISF). In a second step, the q -dependence of the EISF [4] as well as the one from γ and Γ [2] are fitted with the following functions:

$$A_0 = p + (1 - p) \exp\left(-\frac{q^2 a^2}{5}\right) \cdot \left[s \left| \frac{3j_1(qR)}{qR} \right|^2 + (1 - s) \frac{1 + 2j_0(qa_m)}{3} \right] \quad (2)$$

$$\gamma = Dq^2 \quad (3)$$

$$\Gamma = \frac{D_{int} q^2}{1 + D_{int} \tau} \quad (4)$$

6. `brownian_diffusion_internal_free`: This model uses Equation 1 implying the q dependence of of Equation 3. The EISF as well as Γ is kept as q -dependent parameter. The fits are performed as sample dependent fits.
7. `jump_diffision_internal_jump`: This model uses Equation 1 implying the q dependence of of Equation 4 both for the internal as well as for the global diffusion. The EISF is kept as q -dependent parameter. The fits are performed as sample dependent fits.
8. `immobile_jump_diffusion_internal_jump`: This model describes the scattering function assuming a certain fraction of immobile scatterers:

$$S(q, \omega) = A_c \cdot D(q) + (1 - A_c)(A_0(q) \mathcal{L}_\gamma(\omega)) + A_c(1 - A_0(q)) \mathcal{L}_\Gamma(\omega) + (1 - A_c)(1 - A_0(q)) \mathcal{L}_{\gamma+\Gamma}(\omega) \quad (5)$$

The internal as well as the global diffusion implies the q dependence of Equation 4. The EISF is kept as q -dependent parameter. The fits are performed as sample dependent fits. [3]

9. [brownian_diffusion_internal_jump](#): This model uses Equation 1 implying the q dependence of of Equation 3 and Equation 4. The EISF is kept as q -dependent parameter. The fits are performed as sample dependent fits.
10. [brownian_diffusion_internal_switching](#): This model fits the data on the Fitlevel S. The global diffusion is described by a Brownian diffusion with $\gamma = D \cdot q^2$. The internal diffusion is described by Grimaldo *et.al.* [4]. The theoretical framework is given by Roosen-Runge *et al.* [8]. It is described by two linked Lorentzian functions:

$$\Gamma_{1,2} = D_{1,2} \cdot q^2 \quad (6)$$

$$\Lambda = \sqrt{\left(\Gamma_1 - \Gamma_2 + \frac{1}{\tau_1} - \frac{1}{\tau_2}\right)^2 + \frac{4}{\tau_1 \tau_2}} \quad (7)$$

$$\lambda_{1,2} = \frac{\Gamma_1 + \Gamma_2 + \frac{1}{\tau_1} + \frac{1}{\tau_2} \pm \Lambda}{2} \quad (8)$$

$$\alpha = \frac{\Lambda}{\tau_1 + \tau_2} \left((\tau_1 + \tau_2) \lambda_1 - \tau_1 \Gamma_2 - \frac{\tau_1}{\tau_2} - \tau_2 \Gamma_1 - \frac{\tau_2}{\tau_1} - 2 \right) \quad (9)$$

$$S_{int}(q, \omega) = \alpha \mathcal{L}_{\lambda_1}(\omega) + (1 - \alpha) \mathcal{L}_{\lambda_2}(\omega) \quad (10)$$

9.4 Solvent handling

9.4.1 Solvent treatment

Two different methods exist to treat the solvent contribution in QENS spectra:

1. **Subtraction of the solvent contribution:** The solvent contribution can be subtracted from each QENS spectra. In this case, the remaining signal is given by the scattering function of the dissolved particles. To subtract the solvent contribution, set the flag `solventtreatment='subtractsolvent'`.
2. **modeling the solvent contribution:** Next to subtracting the solvent contribution, it can be taken into account in the fit model. To do so, the the flag has to be set to `solventtreatment='modelsolvent'`

9.4.2 Solvent rescaling

In both cases mentioned above in section 9.4.1, the solvent contribution measured from the pure solvent has to be rescaled accordingly to the volume fraction:

1. If the volume fraction is known, it can be fixed. for a certain list of proteins (`todo`) the volume fraction is calculated based on the concentration. If this is not desired, the volume fraction can be specified in the Protocol (see

Section 5.1) as optional parameter. This method is enabled by choosing `solventscaling='fix'`.

2. If the volume fraction is not known, this parameter can be set as fit parameter in the fit. It is recommended to set this parameter constant in q , *i.e.* use at least Fitlevel S or Fitlevel E . However, it is possible to use these three options:

- `solventscaling='freeq'`
- `solventscaling='freeS'` (recommended)
- `solventscaling='freeE'` (recommended)

For a more detailed description of the fit levels, see Section 9.2.

To be written

9.4.3 Solvent modeling

For both cases mentioned in Section 9.4.1, subtraction of the solvent or its modeling, there are several methods, how the solvent data is treated. These are listed in the following:

1. **Use experimental data:** In this option, a measured data set is used to treat the solvent in the spectra. To use this option, the flag `solventmodel='direct'` has to be chosen.
Warning: In combination with `solventtreatment='modelsolvent'` the fit results might look fuzzy, if the solvent was measured not with good statistics.
2. **D2O treatment:** In this method, the water contribution is fitted with one Lorentzian function whose temperature and q -dependent width is fixed based on TOF data as described by Grimaldo *et al.* [6]. This option is used with `solventmodel='D2Ofixed'`.
3. **Model solvent contribution with *one* Lorentzian function:** This option uses one free Lorentzian function (free width and free scaling) to model the contribution of the solvent. To use this model, set `solventmodel='onelorentz'`.
4. **Model solvent contribution with *two* Lorentzian functions:** This option uses one free Lorentzian function (free widths and free scaling parameters) to model the contribution of the solvent. To use this model, set `solventmodel='twolorentz'`.

Option 4 should only be used if there is a significant contribution of the solvent visible in the investigated energy transfer window. broad energy resolutions or limited energy transfers might hide contributions and might therefore allow to use simplified models. See also Section 3.

10 Plot and saving options

There are several plot and saving options. The different options can be combined if they are handed in as a list.

- `plotresults='fits'`: This option displays the fits. Fits of QENS are displayed with $\hbar\omega$ as abscissa, while FWS are displayed with q as abscissa. The ordinate in both cases is in logarithmic scale. In case FWS and QENS are combined, they are displayed similar to the pure QENS spectra. the FWS are plotted also in the plots.
- `plotresults='results'`: This option displays only the fit results but not the Fits itself.
- `plotresults='all'`: This option combines both options of `plotresults='fits'` and `plotresults='results'`.
- `plotresults='silent'`: These option generates the same figures as `plotresults='all'` but they are not displayed but only saved.
- `plotresults='none'`: This option does not create any figures.
- `plotresults='writeascii'`: This option saves all fit results as ascii, which might be usefull, if they should be treated later with other programs.

Next to plotting the fit results as explained above, a summary can be generated by activating the option `writesummary='all/fit/results/none'`. The options `writesummary='all/fit/results'` overwrites the option `plotresults='none'` to `plotresults='all/fit/results'`.

Files are saved as default as ".eps" files. To change this can be saved by setting the flag `pictureextension` from `pictureextension='eps'` to `pictureextension='png'`

Appendix

A Overview of flags to control script

B Index

D ₂ O, 11	Fit, 11
Fitlevel, 13	Solvent, 11, 12
E, 10 , 13	modeling, 13
q, 10 , 11, 13	rescaling, 12
S, 10 , 11–13	treatment, 12
Fitmodel, 11	
Resolution, 10	Vanadium, <i>see</i> Resolution, 11

C Glossary

EISF elastic incoherent structure factor. 11, 12

FWHM full width at half maximum. 11

FWS fixed window scan. 3, 10, 14

NBS neutron back-scattering. 3

QENS quasielastic neutron scattering. 3, 10, 14

TOF time of flight. 3, 13

D Literature

References

- [1] O. Arnold, J.C. Bilheux, J.M. Borreguero, A. Buts, S.I. Campbell, L. Chapon, M. Doucet, N. Draper, R. Ferraz Leal, M.A. Gigg, V.E. Lynch, A. Markvardsen, D.J. Mikkelsen, R.L. Mikkelsen, R. Miller, K. Palmen,

- P. Parker, G. Passos, T.G. Perring, P.F. Peterson, S. Ren, M.A. Reuter, A.T. Savici, J.W. Taylor, R.J. Taylor, R. Tolchenov, W. Zhou, and J. Zikovsky. Mantid—data analysis and visualization package for neutron scattering and μ SR experiments. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 764:156–166, November 2014.
- [2] Christian Beck, Marco Grimaldo, Felix Roosen-Runge, Michal K. Braun, Fajun Zhang, Frank Schreiber, and Tilo Seydel. Nanosecond tracer diffusion as a probe of the solution structure and molecular mobility of protein assemblies: The case of ovalbumin. *J. Phys. Chem. B*, 122(35):8343–8350, 2018.
 - [3] Christian Beck, Marco Grimaldo, Felix Roosen-Runge, Ralph Maier, Olga Matsarskaia, Michal Braun, Benedikt Sohmen, Orsolya Czakkel, Ralf Schweins, Fajun Zhang, Tilo Seydel, and Frank Schreiber. Following protein dynamics in real time during crystallization. *Cryst. Growth Des.*, 19(12):7036–7045, 2019.
 - [4] Marco Grimaldo, Felix Roosen-Runge, Marcus Hennig, Fabio Zanini, Fajun Zhang, Niina Jalarvo, Michaela Zamponi, Frank Schreiber, and Tilo Seydel. Hierarchical molecular dynamics of bovine serum albumin in concentrated aqueous solution below and above thermal denaturation. *Phys. Chem. Chem. Phys.*, 17(6):4645–4655, 2015.
 - [5] Marco Grimaldo, Felix Roosen-Runge, Marcus Hennig, Fabio Zanini, Fajun Zhang, Michaela Zamponi, Niina Jalarvo, Frank Schreiber, and Tilo Seydel. Salt-induced universal slowing down of the short-time self-diffusion of a globular protein in aqueous solution. *J. Phys. Chem. Lett.*, 6(13):2577–2582, 2015.
 - [6] Marco Grimaldo, Felix Roosen-Runge, Niina Jalarvo, Michaela Zamponi, Fabio Zanini, Marcus Hennig, Fajun Zhang, Frank Schreiber, Tilo Seydel, B. Frick, M. M. Koza, M. Boehm, and H. Mutka. High-resolution neutron spectroscopy on protein solution samples. *EPJ Web of Conferences*, 83:02005, 2015.
 - [7] Marco Grimaldo, Felix Roosen-Runge, Fajun Zhang, Frank Schreiber, and Tilo Seydel. Dynamics of proteins in solution. *Quart. Rev. Biophys.*, 52, 2019.
 - [8] Felix Roosen-Runge, Dominique J. Bicout, and Jean-Louis Barrat. Analytical correlation functions for motion through diffusivity landscapes. *The Journal of Chemical Physics*, 144(20):204109, 2016.