
Quantum Metrology with Photoelectrons Vol. 3 *Analysis methodologies*

Paul Hockett

Nov 22, 2022

CONTENTS

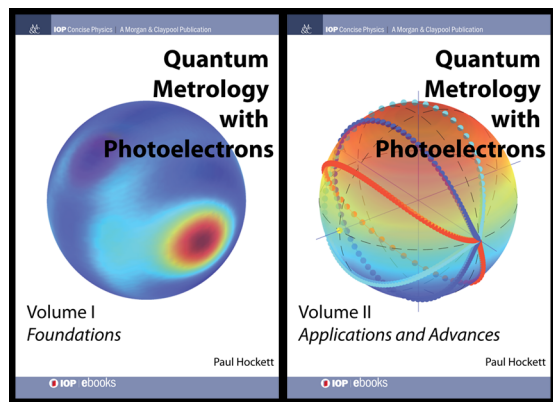
I	Frontmatter	5
1	Overview	7
1.1	General overview	7
1.2	Provisional contents	7
II	Theory & software	9
2	Introduction	11
2.1	Topical introduction	11
2.2	Context & aims for Vol. 3	11
3	Quantum metrology software platform/ecosystem overview	13
3.1	Analysis components	13
3.2	Additional tools	15
3.3	Docker deployments	15
3.4	General discussion	15
4	Theory	17
4.1	Observables: photoelectron flux in the LF and MF	17
4.2	Photoionization dynamics	20
III	Backmatter	23
5	Bibliography	25

Quantum Metrology with Photoelectrons Volume 3: *Analysis methodologies*, an open source executable book. This repository contains the source documents (mainly Jupyter Notebooks in Python) and notes for the book, as of Jan 2022 writing is in progress, and the [current HTML build can be found online](#). The book is due to be finished in 2023, and will be published by IOP Press - see below for more details.

Series abstract

Photoionization is an interferometric process, in which multiple paths can contribute to the final continuum photoelectron wavefunction. At the simplest level, interferences between different final angular momentum states are manifest in the energy and angle resolved photoelectron spectra: metrology schemes making use of these interferograms are thus phase-sensitive, and provide a powerful route to detailed understanding of photoionization. In these cases, the continuum wavefunction (and underlying scattering dynamics) can be characterised. At a more complex level, such measurements can also provide a powerful probe for other processes of interest, leading to a more general class of quantum metrology built on phase-sensitive photoelectron imaging. Since the turn of the century, the increasing availability of photoelectron imaging experiments, along with the increasing sophistication of experimental techniques, and the availability of computational resources for analysis and numerics, has allowed for significant developments in such photoelectron metrology.

About the books



- Volume I covers the core physics of photoionization, including a range of computational examples. The material is presented as both reference and tutorial, and should appeal to readers of all levels. ISBN 978-1-6817-4684-5, <http://iopscience.iop.org/book/978-1-6817-4684-5> (IOP Press, 2018)
- Volume II explores applications, and the development of quantum metrology schemes based on photoelectron measurements. The material is more technical, and will appeal more to the specialist reader. ISBN 978-1-6817-4688-3, <http://iopscience.iop.org/book/978-1-6817-4688-3> (IOP Press, 2018)

Additional online resources for Vols. I & II can be found on [OSF](#) and [Github](#).

- Volume III in the series will continue this exploration, with a focus on numerical analysis techniques, forging a closer link between experimental and theoretical results, and making the methodologies discussed directly accessible via new software. The book is due for publication by IOP due in 2023; this volume is also open-source, with a live HTML version at <https://phockett.github.io/Quantum-Metrology-with-Photoelectrons-Vol3/> and source available at <https://github.com/phockett/Quantum-Metrology-with-Photoelectrons-Vol3>.

For some additional details and motivations (including topical video), see [the ePSdata project](#).

Technical details

This repository contains:

- `doc-source`: the source documents (mainly Jupyter Notebooks in Python)
- `notes`: additional notes for the book,
- the `gh-pages` branch contains the current HTML build, also available at <https://phockett.github.io/Quantum-Metrology-with-Photoelectrons-Vol3/>

The project has been setup to use the [Jupyter Book](#) build-chain (which uses Sphinx on the back-end) to generate HTML and Latex outputs for publication from source Jupyter notebooks & markdown files.

The work *within* the book will make use of the [Photoelectron Metrology Toolkit](#) platform for working with experimental & theoretical data.



Running code examples

Each Jupyter notebook (*.ipynb) can be treated as a stand-alone computational document. These can be run/used/modified independently with an appropriately setup python environment (details to follow).

Building the book

The full book can also be built from source:

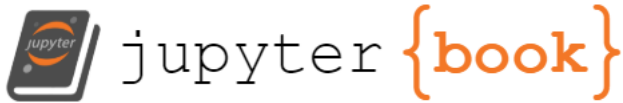
1. Clone this repository
2. Run `pip install -r requirements.txt` (it is recommended you do this within a virtual environment)
3. (Optional) Edit the books source files located in the `doc-source/` directory
4. Run `jupyter-book clean doc-source/` to remove any existing builds
5. For an HTML build:
 - Run `jupyter-book build doc-source/`
 - A fully-rendered HTML version of the book will be built in `doc-source/_build/html/`.
6. For a LaTeX & PDF build:
 - Run `jupyter-book build doc-source/ --builder pdflatex`
 - A fully-rendered HTML version of the book will be built in `doc-source/_build/latex/`.

See <https://jupyterbook.org/basics/building/index.html> for more information.

Credits

This project is created using the open source [Jupyter Book project](#) and the [executablebooks/cookiecutter-jupyter-book template](#).

To add: build env & main software packages (see automation for this...)



Part I

Frontmatter

OVERVIEW

1.1 General overview

Vol. 3. will focus on analysis techniques for quantum metrology with photoelectrons, including:

- Interpreting experimental data.
- Extraction/reconstruction/determination of quantum mechanical properties (matrix elements, wavefunctions, density matrices) from experimental data.
- Comparison of experimental and theoretical data.
- New analysis methodologies & techniques.
- Introduction to newly-developed software platform (see below).

1.2 Provisional contents

1.2.1 Part 1: theory & software

General review & update of the topic, including recent theory developments.

1. Introduction
 - a. Topic overview.
 - b. Context of vol. 3 (following vols. 1 & 2).
 - c. Aims: Vol. 3 in the series will continue the exploration of quantum metrology with photoelectrons, with a focus on numerical analysis techniques, forging a closer link between experimental and theoretical results, and making the methodologies discussed directly accessible via a new software platform/ecosystem.
2. Quantum metrology software platform/ecosystem overview
 - a. Introduction to python packages for simulation, data analysis, and open-data.
 - b. Photoelectron metrology toolkit (PEMtk) package/platform for experimental data processing & analysis. (See pemtk.readthedocs.io.)
 - c. ePSproc package for theory & simulation. (See epsproc.readthedocs.io.)
 - d. ePSdata platform for data/results library (see [ePSdata motivations](#)).
3. General method development: geometric tensor treatment of photoionization, fitting & matrix-inversion techniques
 - a. Theory development overview - tensor methods (e.g. [ePSproc tensor methods](#))

b. Direct molecular frame reconstruction via matrix-inversion methods (see Gregory, Margaret, Paul Hockett, Albert Stolow, and Varun Makhija. “Towards Molecular Frame Photoelectron Angular Distributions in Polyatomic Molecules from Lab Frame Coherent Rotational Wavepacket Evolution.” *Journal of Physics B: Atomic, Molecular and Optical Physics* 54, no. 14 (July 2021): 145601. DOI: 10.1088/1361-6455/ac135f.)

4. Numerical implementation & analysis platform tools

- a. Tensor methods implementation in ePSproc/PEMtk.
- b. Information content analysis (inc. basis-set exploration, e.g. *PEMtk fitting demo*), see also vol. 2, sect. 12.1.
- c. Density matrix analysis. (e.g. *ePSproc density matrix method dev notes*)
- d. Generalised bootstrapping implementation in PEMtk (see vol. 2, sects. 11.3 & 12.3)

1.2.2 Part 2: numerical examples

Open-source worked examples using the new software platform.

1. Quantum metrology example: generalised bootstrapping for a homonuclear diatomic scattering system (N₂)*
 - a. Experimental data overview & simulation.
 - b. Matrix element extraction (bootstrap protocol, see vol. 2, sects. 11.3 & 12.3) & statistical analysis.
 - c. Direct molecular frame reconstruction via matrix-inversion methods.
 - d. Comparison of methods.
 - e. Information content/quantum information analysis. (See vol. 2, sect. 12.1.)
2. Quantum metrology example: generalised bootstrapping for a heteronuclear scattering system (CO)*
 - a. Experimental data overview & simulation.
 - b. Matrix element extraction (bootstrap protocol, see vol. 2, sects. 11.3 & 12.3) & statistical analysis.
 - c. Direct molecular frame reconstruction via matrix-inversion methods.
 - d. Comparison of methods.
 - e. Information content/quantum information analysis. (See vol. 2, sect. 12.1.)
3. Quantum metrology example: generalised bootstrapping and matrix-inversion methods for a complex/general asymmetric top scattering system (C₂H₄ (ethylene))*
 - a. Experimental data overview & simulation.
 - b. Matrix element extraction (bootstrap protocol, see vol. 2, sects. 11.3 & 12.3) & statistical analysis.
 - c. Direct molecular frame reconstruction via matrix-inversion methods.
 - d. Comparison of methods.
 - e. Information content/quantum information analysis.
4. Future directions & outlook
5. Summary & conclusions

* Exact choice of “simple” and “complex” systems may change, but should include a homonuclear diatomic and/or heteronuclear diatomic, and symmetric and asymmetric top polyatomic systems. May also include an atomic example.

Part II

Theory & software

INTRODUCTION

The overall aim of Vol. 3 is to expand, explore, and illustrate, quantum metrology with photoelectrons: specifically, the application of new python-based tools to tackle problems in matrix element retrieval. The book itself is written as a set of Jupyter Notebooks, hence all the material herein is available directly to readers, and can be run locally to further explore the topic, or adapt the methodology to new problems

Whilst this volume aims to provide a self-contained text, and computational examples which may be used without extensive background knowledge, a brief introduction to the core physics and some recent extensions is presented herein. The unfamiliar reader is referred to Volume 1 of the series for a more detailed presentation, and gateway to the literature []. Following the topical introduction, the remainder of Part I introduces the main computational and software tools, recent theory developments, and concludes with a general overview for approaching matrix element retrieval numerically.

Part II details the application of these tools to a few specific cases, starting with a (relatively) simple homonuclear diatomic example, then escalating to a polyatomic asymmetric top case.

2.1 Topical introduction

2.2 Context & aims for Vol. 3

As noted previously, Vol. 3 is somewhat distinct from the previous volumes in the series; although involving computational elements, Vols. 1 & 2 [] are more traditional publications. The material presented in this volume aims to continue the exploration of quantum metrology with photoelectrons, with a focus on numerical analysis techniques, forging a closer link between experimental and theoretical results, and making the methodologies discussed directly accessible via a new software platform/ecosystem. In order to fulfil this aim, Vol. 3 is a computational/computable document, with code directly available to readers. Each chapter or section is composed of a Jupyter Notebook (`.ipynb`), each of which can be modified and used independently.

To facilitate code transparency and reuse, the book is available via a Github repository, [Quantum Metrology Vol. 3](#). An HTML version is also available, which includes interactive figures. A full introduction to the relevant tool-chain, including installation instructions, can be found in [Chapter 3: Quantum metrology software platform/ecosystem overview](#).

QUANTUM METROLOGY SOFTWARE PLATFORM/ECOSYSTEM OVERVIEW

STUB

In recent years, a unified Python codebase/ecosystem/platform has been in development to tackle various aspects of photoionization problems, including *ab initio* computations and experimental data handling, and (generalised) matrix element retrieval methods. The eponymous *Quantum Metrology with Photoelectrons* platform is introduced here, and is used for the analysis herein. The main aim of the platform is to provide a unifying data platform, and analysis routines, for photoelectron metrology, including new methods and tools, as well as a unifying bridge between these and existing tools. [Fig. 3.1](#) provides a general overview of some of the main tools and tasks/layers.

As of late 2022, the new parts of the platform - primarily the [Photoelectron Metrology Toolkit](#) [] library - implement general data handling (although not a full experimental analysis toolchain), matrix element handling and retrieval, which will be the main topic of this volume. In the future, it is hoped that the platform will be extended to other theoretical and experimental methods, including full experimental data handling.

3.1 Analysis components

The two main components of the platform for analysis tasks, as used herein, are:

- The [Photoelectron Metrology Toolkit](#) [] (PEMtk) codebase aims to provide various general data handling routines for photoionization problems. At the time of writing, simulation of observables and fitting routines are implemented, along with some basic utility functions. Much of this is detailed herein, and more technical details and ongoing documentation can be found in the [PEMtk documentation](#) [].
- The [ePSproc](#) codebase [] aims to provide methods for post-processing with *ab initio* radial dipole matrix elements from [ePolyScat \(ePS\)](#) [], or equivalent matrix elements from other sources (dedicated support for R-matrix results from the [RMT suite](#) [] is in development). The core functionality includes the computation of AF and MF observables. Manual computation without known matrix elements is also possible, e.g. for investigating limiting cases, or data analysis and fitting - hence these routines also provide the backend functionality for PEMtk fitting routines. Again more technical details can be found in the [ePSproc documentation](#) [].

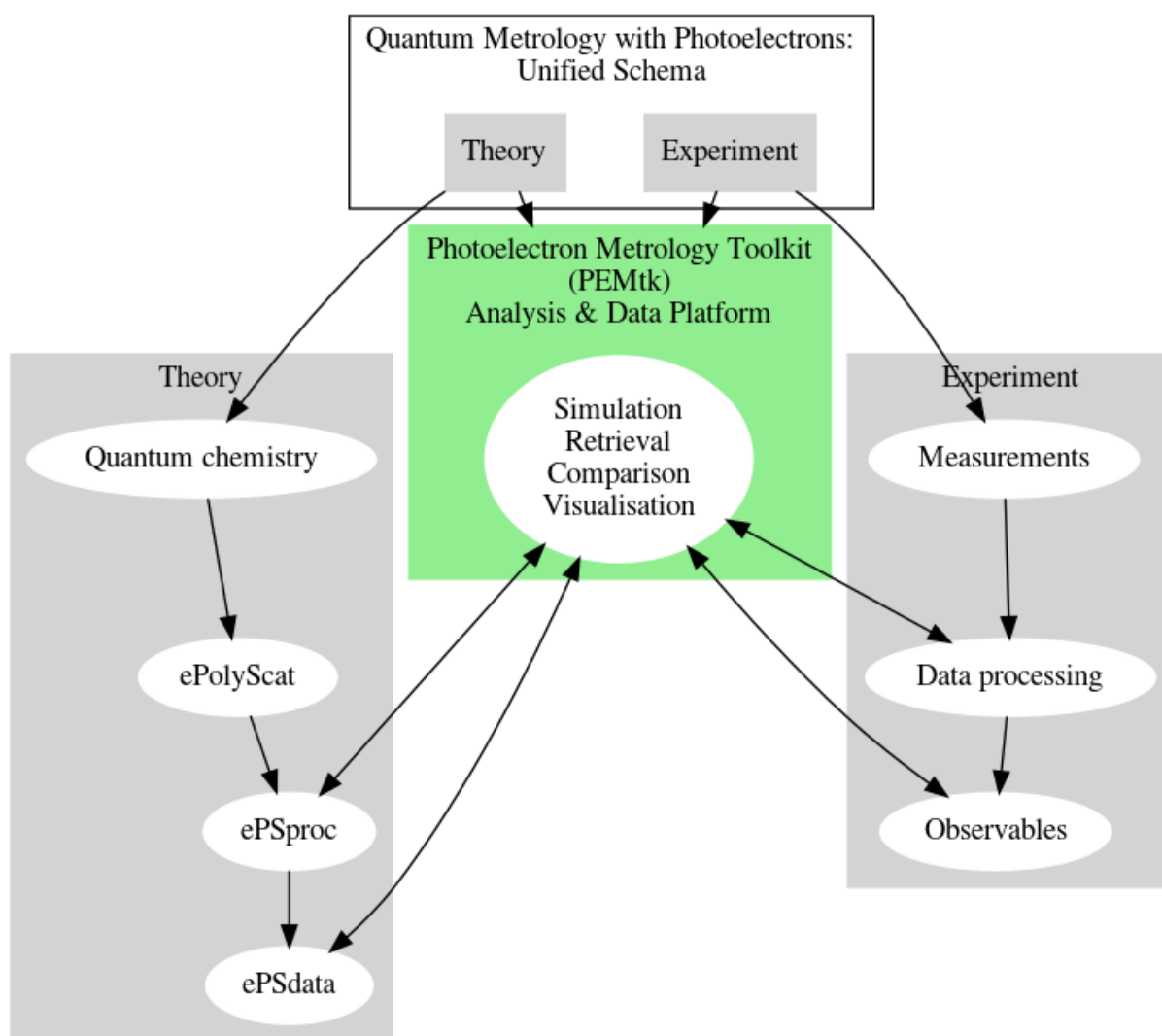


Fig. 3.1: Quantum metrology with photoelectrons ecosystem overview.

3.2 Additional tools

Other tools listed in Fig. 3.1 include:

- Quantum chemistry layer. The starting point for *ab initio* computations. For the examples herein, all computations made use of Gamess (“The General Atomic and Molecular Electronic Structure System”) [] for electronic structure computations, and inputs to ePolyScat.
- ePolyScat (ePS) [] is an open-source tool for numerical computation of electron-molecule scattering & photoionization by Lucchese & coworkers. All matrix elements used herein were obtained via ePS calculations. For more details see ePolyScat website and manual [] and Refs. [].
- ePSdata [] is an open-data/open-science collection of ePS + ePSproc results.
 - ePSdata collects ePS datasets, post-processed via ePSproc (Python) in Jupyter notebooks, for a full open-data/open-science transparent pipeline.
 - Source notebooks are available on the ePSdata [] Github project repository, and notebooks + datasets via ePSdata Zenodo []. Each notebook + dataset is given a Zenodo DOI for full traceability, and notebooks are versioned on Github.
 - Note: ePSdata may also be linked or mirrored on the existing ePolyScat Collected Results OSF project, but will effectively supercede those pages.
 - All results are released under Creative Commons Attribution-NonCommercial-ShareAlike 4.0 (CC BY-NC-SA 4.0) license, and are part of an ongoing Open Science initiative.

3.3 Docker deployments

A Docker-based distribution of various codes for tackling photoionization problems is also available from the Open Photoionization Docker Stacks [] project, which aims to make a range of these tools more accessible to interested researchers, and fully cross-platform/portable. The project currently includes Docker builds for ePS, ePSproc and PEMtk.

3.4 General discussion

Note that, at the time of writing, rotational wavepacket simulation is not yet implemented in the PEMtk suite, and these must be obtained via other codes. An initial build of the limapack suite for rotational wavepacket simulations is currently part of the Open Photoionization Docker Stacks [], but has yet to be tested.

THEORY

STUB

TODO: plotly HV wrappers, see <https://github.com/executablebooks/jupyter-book/issues/1815>

UPDATE: now in place, working in Render-debug notebook, but not here - even after clean build? Weird.

TODO: Fix plots (HV & plotly) in PDF output, currently missing here and Render-debug notebook.

17/11/22 Rendering now OK for Plotly, ongoing notes: <https://github.com/phockett/Quantum-Metrology-with-Photoelectrons-Vol3/issues/2>

4.1 Observables: photoelectron flux in the LF and MF

The observables of interest - the photoelectron flux as a function of energy, ejection angle, and time - can be written quite generally as an expansion in spherical harmonics:

$$\bar{I}(\epsilon, t, \theta, \phi) = \sum_{L=0}^{2n} \sum_{M=-L}^L \bar{\beta}_{L,M}(\epsilon, t) Y_{L,M}(\theta, \phi) \quad (4.1)$$

Here the flux in the laboratory frame (LF) or aligned frame (AF) is denoted $\bar{I}(\epsilon, t, \theta, \phi)$, with the bar signifying ensemble averaging, and the molecular frame flux by $I(\epsilon, t, \theta, \phi)$. Similarly, the expansion parameters $\bar{\beta}_{L,M}(\epsilon, t)$ include a bar for the LF/AF case. These observables are generally termed photoelectron angular distributions (PADs), often with a prefix denoting the reference frame, e.g. LFPADs, MFPADs, and the associated expansion parameters $\bar{\beta}_{L,M}(\epsilon, t)$ are generically termed “anisotropy” parameters. The polar coordinate system (θ, ϕ) is referenced to an experimentally-defined axis in the LF/AF case (usually defined by the laser polarization), and the molecular symmetry axis in the MF. Some arbitrary examples are given in Fig. 4.1, which illustrates both a range of distributions of increasing complexity, and some basic code to set $\beta_{L,M}$ parameters and visualise them; the values used as tabulated in Fig. 4.2.

```
# Plot some distributions from specified BLMs

# Set specific LM coeffs by list with setBLMs, items are [l,m,value]
from epsproc.sphCalc import setBLMs

# BLM = setBLMs([[0,0,1],[1,1,1],[2,2,1]])
# BLM = setBLMs([[0,0,1,1,1],[1,1,1,0.5,0.2],[2,2,1,1,0.2]]) # Note different index
BLM = setBLMs([[0,0,1,1,1,1],[1,1,0,0.5,0.8,1],[2,0,1,0.5,0,0],
               [4,2,0,0,0,0.5],[4,-2,0,0,0,0.5]])

# Set the backend to 'pl' for an interactive surface plot with Plotly
# NOTE PL FIG RETURN BROKEN FOR THIS CASE (epsproc v1.3.1), so run sphSumPlotX too.
dataPlot, figObj = ep.sphFromBLMPlot(BLM, facetDim='t', plotFlag = False, backend = _
    plotBackend);
```

(continues on next page)

(continued from previous page)

```

figObj = ep.sphSumPlotX(dataPlot, facetDim='t', plotFlag = False, backend = plotBackend);

# And GLUE for display later with caption
# from myst_nb import glue
# glue("padExamplePlot", figObj[0], display=False);
# Glue with Plotly wrapper.
# gluePlotly("padExamplePlot", figObj[0]) # Working in Render test notebook, but not here? Issue with subplots?

# Test in separate cell...
gluePlotly("padExamplePlot", figObj[0]) # Working in Render test notebook, but not here? Issue with subplots?

```

Fig. 4.1: Examples of angular distributions (expansions in spherical harmonics $Y_{L,M}$), for a range of cases. Note that up-down asymmetry is associated with odd- l contributions, and breaking of cylindrical symmetry with $m \neq 0$ terms.

Fig. 4.2: Values used for the plots in Fig. 4.1

In general, the spherical harmonic rank and order (L, M) of Eq. (4.1) are constrained by experimental factors in the LF/AF, and n is effectively limited by the molecular alignment (which is correlated with the photon-order for gas phase experiments, or conservation of angular momentum in the LF more generally []), but in the MF is defined by the maximum continuum angular momentum $n = l_{max}$ imparted by the scattering event [].

For basic cases these limits may be low: for instance, a simple 1-photon photoionization event ($n = 1$) from an isotropic ensemble (zero net ensemble angular momentum) defines $L_{max} = 2$; for cylindrically symmetric cases (i.e. $D_{\infty h}$ symmetry) $M = 0$ only. For MF cases, $l_{max} = 4$ is often given as a reasonable rule-of-thumb for the continuum - hence $L_{max} = 8$ - although in practice higher- l may be populated. Some realistic example cases are discussed later (**PART II**), see also ref. [] for more discussion and complex examples.

In general, these observables may also be dependent on various other parameters; in Eq. (4.1) two such parameters, (ϵ, t) , are included, as the usual variables of interest. Usually ϵ denotes the photoelectron energy, and t is used in the case of time-dependent (usually pump-probe) measurements. As discussed below (Sect. 4.2), the origin of such dependencies may be complicated but, in general, the associated photoionization matrix elements are energy-dependent, and time-dependence may also appear for a number of intrinsic or extrinsic (experimental) reasons, e.g. electronic or nuclear dynamics, rotational (alignment) dynamics, electric field dynamics etc. In many cases only one particular aspect may be of interest, so t can be used as a generic label to index changes as per Fig. 4.1.

4.1.1 Symmetrized harmonics

Symmetrized (or generalised) harmonics, which essentially provide correctly symmetrized expansions of spherical harmonics (Y_{lm}) functions for a given irreducible representation, Γ , can be defined by linear combinations of spherical harmonics (Refs. [] as below):

$$X_{hl}^{\Gamma\mu*}(\theta, \phi) = \sum_{\lambda} b_{hl\lambda}^{\Gamma\mu} Y_{l,\lambda}(\theta, \phi) \quad (4.1)$$

where:

- Γ is an irreducible representation,
- (l, λ) define the usual spherical harmonic indicies (rank, order)
- $b_{hl\lambda}^{\Gamma\mu}$ are symmetrization coefficients,

- index μ allows for indexing of degenerate components,
- h indexes cases where multiple components are required with all other quantum numbers identical.

The exact form of these coefficients will depend on the point-group of the system, see, e.g. Refs. []. Numerical routines for the generation of symmetrized harmonics are implemented in PEMtk: point-groups, character table generation and symmetrization (computing $b_{hl\lambda}^{\Gamma\mu}$ parameters) is handled by libmsym []; additional handling also makes use of pySHtools []. A brief example is given below, see the PEMtk documentation [] for more details.

```
# Import class
from pemtk.sym.symHarm import symHarm

# Compute hamronics for Td, lmax=4
sym = 'Td'
lmax=4

symObj = symHarm(sym,lmax)

# Character tables can be displayed
symObj.printCharacterTable()

# Glue items for later
glue("symHarmPG2", f"${sym}$", display=False)
glue("symHarmPG", sym, display=False)
glue("symHarmLmax", lmax, display=False)

# The full set of expansion parameters can be tabulated
# pd.set_option('display.max_rows', 1)
symObj.displayXlm() # Display values (note this defaults to REAL harmonics)
# symObj.displayXlm(YlmType='comp') # Display values for COMPLEX harmonic expansion.

# To plot using ePSproc/PEMtk class, these values can be converted to ePSproc BLM_
# data type...

# Run conversion - the default is to set the coeffs to the 'BLM' data type
symObj.toePSproc()

# Set to new key in data class
data.data['symHarm'] = {}

for dataType in ['BLM']: #['matE','BLM']:
    data.data['symHarm'][dataType] = symObj.coeffs[dataType]['b (comp)'] # Select_
    # expansion in complex harmonics
    data.data['symHarm'][dataType].attrs = symObj.coeffs[dataType].attrs

# Plot full harmonics expansions, plots by symmetry
# Note 'squeeze=True' to force drop of singleton dims may be required.
# data.padPlot(keys='symHarm',dataType='BLM', facetDims = ['Cont'], squeeze = True,
# backend=plotBackend)

data.padPlot(keys='symHarm',dataType='BLM', facetDims = ['Cont'], squeeze = True,
backend=plotBackend, plotFlag=False, returnFlag=True) # Working
figObj = data.data['symHarm']['plots']['BLM']['polar'][0]

# And GLUE for display later with caption
# from myst_nb import glue
# glue("padExamplePlot2", figObj, display=False);
```

(continues on next page)

(continued from previous page)

```
gluePlotly("symHarmPADs", figObj)
```

Fig. 4.3: Examples of angular distributions from expansions in symmetrized harmonics $X_{hl}^{\Gamma\mu*}(\theta, \phi)$, for all irreducible representations in symmetry ($l_{max} =$)

4.2 Photoionization dynamics

The core physics of photoionization has been covered extensively in the literature, and only a very brief overview is provided here with sufficient detail to introduce the metrology/reconstruction/retrieval problem; the reader is referred to Vol. 1 [] (and refs. therein) for further details and general discussion.

Photoionization can be described by the coupling of an initial state of the system to a particular final state (photoion(s) plus free photoelectron(s)), coupled by an electric field/photon. Very generically, this can be written as a matrix element $\langle \Psi_i | \hat{\Gamma}(\mathbf{E}) | \Psi_f \rangle$, where $\hat{\Gamma}(\mathbf{E})$ defines the light-matter coupling operator (depending on the electric field \mathbf{E}), and Ψ_i, Ψ_f the total wavefunctions of the initial and final states respectively.

There are many flavours of this fundamental light-matter interaction, depending on system and coupling. For metrology, the focus is currently on the simplest case of single-photon absorption, in the weak field (or perturbative), dipolar regime, resulting in a single photoelectron. (For more discussion of various approximations in photoionization, see Refs. []). In this case the core physics is well defined, and tractable (albeit non-trivial), via the separation of matrix elements into radial (energy) and angular-momentum (geometric) terms pertaining to couplings between various elements of the problem; the retrieval of such matrix elements is a well-defined problem, making use of analytic terms in combination with fitting methodologies as explored herein. Again, more extensive background and discussion can be found in *Quantum Metrology* Vol. 1 [], and references therein. [Add some more refs here?]

The basic case also provides a strong foundation for extension into more complex light-matter interactions, in particular cases with shaped laser-fields (i.e. a time-dependent coupling $\hat{\Gamma}(\mathbf{E}, \mathbf{t})$) and multi-photon processes (which require multiple matrix elements). Note, however, that non-perturbative (strong field) light-matter interactions are, typically, not amenable to description in a separable picture in this manner. In such cases the laser field, molecular and continuum properties are strongly coupled, and are typically treated numerically in a fully time-dependent manner (although some separation of terms may work in some cases).

Underlying the photoelectron observables is the photoelectron continuum state $|\mathbf{k}\rangle$, prepared via photoionization. The photoelectron momentum vector is denoted generally by $\mathbf{k} = k\hat{\mathbf{k}}$, in the MF. The ionization matrix elements associated with this transition provide the set of quantum amplitudes completely defining the final continuum scattering state,

$$|\Psi_f\rangle = \sum \int |\Psi_+; \mathbf{k}\rangle \langle \Psi_+; \mathbf{k} | \Psi_f \rangle d\mathbf{k}, \quad (4.2)$$

where the sum is over states of the molecular ion $|\Psi_+\rangle$. The number of ionic states accessed depends on the nature of the ionizing pulse and interaction. For the dipolar case,

$$\hat{\Gamma}(\mathbf{E}) = \hat{\mu} \cdot \mathbf{E}$$

Hence,

$$\langle \Psi_+; \mathbf{k} | \Psi_f \rangle = \langle \Psi_+; \mathbf{k} | \hat{\mu} \cdot \mathbf{E} | \Psi_i \rangle \quad (4.3)$$

Where the notation implies a perturbative photoionization event from an initial state i to a particular ion plus electron state following absorption of a photon $h\nu$, $|\Psi_i\rangle + h\nu \rightarrow |\Psi_+; \mathbf{k}\rangle$, and $\hat{\mu} \cdot \mathbf{E}$ is the usual dipole interaction term [], which includes a sum over all electrons s defined in position space as \mathbf{r}_s :

$$\hat{\mu} = -e \sum_s \mathbf{r}_s \quad (4.4)$$

The position space photoelectron wavefunction is typically expressed in the “partial wave” basis, expanded as (asymptotic) continuum eigenstates of orbital angular momentum, with angular momentum components (l, m) (note lower case notation for the partial wave components, distinct from upper-case for the similar terms (L, M) in the observables),

$$\Psi_{\mathbf{k}}(r) \equiv \langle r | \mathbf{k} \rangle = \sum_{lm} Y_{lm}(\hat{\mathbf{k}}) \psi_{lm}(r, k) \quad (4.5)$$

where r are MF electronic coordinates and $Y_{lm}(\hat{\mathbf{k}})$ are the spherical harmonics.

Similarly, the ionization dipole matrix elements can be separated generally into radial (energy-dependent or ‘dynamical’ terms) and geometric (angular momentum) parts (this separation is essentially the Wigner-Eckart Theorem, see Ref. [] for general discussion), and written generally as (using notation similar to []):

$$\langle \Psi_+; \mathbf{k} | \hat{\mu} \cdot \mathbf{E} | \Psi_i \rangle = \sum_{lm} \gamma_{l,m} \mathbf{r}_{k,l,m} \quad (4.6)$$

Provided that the geometric part of the matrix elements $\gamma_{l,m}$ - which includes the geometric rotations into the LF arising from the dot product in Eq. (4.6) and other angular-momentum coupling terms - are known, knowledge of the so-called radial (or reduced) dipole matrix elements, at a given k thus equates to a full description of the system dynamics (and, hence, the observables).

For the simplest treatment, the radial matrix element can be approximated as a 1-electron integral involving the initial electronic state (orbital), and final continuum photoelectron wavefunction:

$$\mathbf{r}_{k,l,m} = \int \psi_{lm}(r, k) r \Psi_i(r) dr \quad (4.7)$$

As noted above, the geometric terms $\gamma_{l,m}$ are analytical functions which can be computed for a given case - minimally requiring knowledge of the molecular symmetry and polarization geometry, although other factors may also play a role (see Sect. 4.2 for details).

The photoelectron angular distribution (PAD) at a given (ϵ, t) can then be determined by the squared projection of $|\Psi_f\rangle$ onto a specific state $|\Psi_+; \mathbf{k}\rangle$,

and therefore the amplitudes in Eq. (4.6) also determine the observable anisotropy parameters $\beta_{L,M}(\epsilon, t)$ (Eqn. (4.1)). (Note that the photoelectron energy ϵ and momentum k are used somewhat interchangeably herein, with the former usually preferred in reference to observables.) Note, also, that in the treatment above there is no time-dependence incorporated in the notation; however, a time-dependent treatment readily follows, and may be incorporated either as explicit time-dependent modulations in the expansion of the wavefunctions for a given case, or implicitly in the radial matrix elements. Examples of the former include, e.g. a rotational or vibrational wavepacket, or a time-dependent laser field. The rotational wavepacket case is discussed herein (see Sect. 4.2). The radial matrix elements are a sensitive function of molecular geometry and electronic configuration in general, hence may be considered to be responsive to molecular dynamics, although they are formally time-independent in a Born-Oppenheimer basis - for further general discussion and examples see Ref. []; discussions of more complex cases with electronic and nuclear dynamics can be found in Refs. [].

Typically, for reconstruction experiments, a given measurement will be selected to simplify this as much as possible by, e.g., populating only a single ionic state (or states for which the corresponding observables are experimentally energetically-resolvable), and with a bandwidth $d\mathbf{k}$ which is small enough such that the matrix elements can be assumed constant. Importantly, the angle-resolved observables are sensitive to the magnitudes and (relative) phases of these matrix elements, and can be considered as angular interferograms

Part III

Backmatter

BIBLIOGRAPHY