JOURNAL OF SYNCHROTRON RADIATION

ISSN 1600-5775

Received 11 June 2025
Accepted 29 September 2025

Edited by M. Guizar-Sicairos, Paul Scherrer Institute, Switzerland, and EPFL, Switzerland

Keywords: X-ray imaging; phase retrieval; computed tomography; phase contrast; holography.

HoToPy: a toolbox for X-ray holo-tomography in Python

Jens Lucht,[a]* Paul Meyer,[a] Leon Merten Lohse[b,a] and Tim Salditt[a]*

[a]Georg-August-Universität Göttingen, Institut für Röntgenphysik, Friedrich-Hund-Platz 1, 37077 Göttingen, Germany, and [b]Universität Hamburg, The Hamburg Centre for Ultrafast Imaging, Luruper Chaussee 149, 22761 Hamburg, Germany. *Correspondence e-mail: jens.lucht@uni-goettingen.de, tsaldit@gwdg.de

We present a Python toolbox for holographic and tomographic X-ray imaging. It comprises a collection of phase retrieval algorithms for the deeply holographic and direct contrast imaging regimes, including non-linear approaches and extended choices of regularization, constraint sets and optimizers, all implemented with a unified and intuitive interface. Moreover, it features auxiliary functions for (tomographic) alignment, image processing and simulation of imaging experiments. The capability of the toolbox is illustrated by an example of a catalytic nanoparticle, imaged in the deeply holographic regime at the 'GINIX' instrument of the P10 beamline at the PETRA III storage ring (DESY, Hamburg, Germany). Due to its modular design, the toolbox can be used for algorithmic development and benchmarking in a lean and flexible manner, or be interfaced and integrated in the reconstruction pipeline of other synchrotron or X-ray free-electron laser instruments for phase imaging based on propagation.

1. Introduction

The ability of X-ray radiation to penetrate matter is key to its use as a non-destructive probe for the inner structure of objects, materials and tissues, by way of computed tomography (CT). Penetration relies on weak attenuation and therefore has always been a limitation as much as an enabling property. Contrast vanishes for soft tissues and low-Z materials at scales in the micrometer range and below, when attenuation becomes insufficient. The more recent exploitation of phase contrast based on (partially) coherent beam propagation has helped to overcome these limits, offering sufficient contrast even to unstained soft biological tissue, soft matter materials and/or nanoscale structures in solution. Different experimental techniques exist to transform phase shifts imparted by a sample into measurable intensity patterns. One example is propagation based imaging (PBI), where intensity patterns emerge through self-interference after sufficient (optics-free) free space propagation, as illustrated in Fig. 1. In ptychography this is additionally combined with lateral sample shifts (Rodenburg & Faulkner, 2004; Robisch *et al.*, 2015; Pfeiffer, 2018). Other techniques probe the phase by additional optical elements such as grating-based (Momose *et al.*, 2009; Pfeiffer *et al.*, 2013) or speckle imaging (Zanette *et al.*, 2014). For a detailed introduction to X-ray phase contrast imaging we refer to the following references: Paganin (2006); Salditt *et al.* (2020); Quenot *et al.* (2022). Hence, X-ray phase contrast imaging and computed tomography (XPCT) is a unique tool for a wide range of applications. Recent cutting edge examples are diverse, including nanoimaging of neuronal tissue for connectomics (Livingstone *et al.*, 2025; Azevedo *et al.*, 2024), morphological transitions of nanoparticles in solution (Grote

OPEN ACCESS
Published under a CC BY 4.0 licence

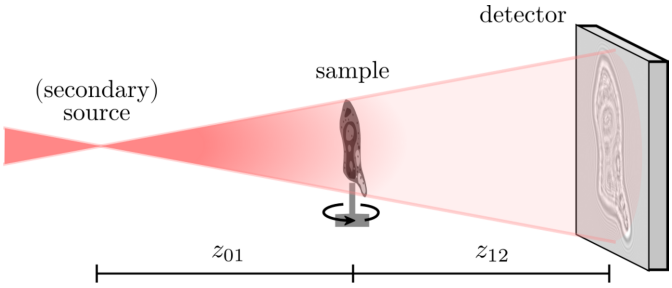

Figure 1
Schematic for PBI in cone-beam geometry. A diverging X-ray beam illuminates a sample with weak absorption, which is placed at a defocus distance z_{01}. The sample imprints phase shifts onto the X-ray wavefront, which render into a measurable self-interference pattern on a detector, placed at distance z_{12} downstream of the sample, after sufficient free-space propagation. For tomography the sample is rotated and imaged at multiple angles.

et al., 2022; Veselý *et al.*, 2021), or ultrafast imaging of hydrodynamics at X-ray free-electron lasers (XFELs) for cavitation (Hoeppe *et al.*, 2024) and fusion confinement research (Montgomery, 2023). In all cases, efficient and high-quality phase retrieval is a key element in phase contrast imaging, in particular in the high resolution full-field variant of holographic tomography (holo-tomography) (Cloetens *et al.*, 1999).

Emerging capabilities of fourth-generation synchrotrons (Chushkin & Zontone, 2025; Li *et al.*, 2022) and XFEL sources, of larger and faster detectors (Correa *et al.*, 2024; Donath *et al.*, 2023; Hatsui & Graafsma, 2015), as well as of optics and instruments, enable higher spatial resolution, dose efficiency, larger fields of view and shorter acquisition times (Vijayakumar *et al.*, 2024; Spiecker *et al.*, 2023; Astolfo *et al.*, 2017). At the other end of the image chain, new paradigms in image analysis are fueled by machine learning, which requires large image libraries, for example as training data (Bellens *et al.*, 2024; Flenner *et al.*, 2022; Hendriksen *et al.*, 2020). To meet both ends, reconstruction software and computing pipelines must keep up with the boost in efficiency and throughput, while at the same time achieving image quality beyond the standard linearized phase retrieval approaches.

A variety of toolboxes already exist for X-ray holography, tomography and ptychography, reflecting the growing demand for advanced reconstruction software. A few examples are *PyPhase* (Langer *et al.*, 2021), *CIL* (Jørgensen *et al.*, 2021), *TomoPy* (Gürsoy *et al.*, 2014), *HoloTomoToolbox* (Lohse *et al.*, 2020), *Holotomocupy* (Nikitin, 2023), *TOFU/UFO* (Faragó *et al.*, 2022; Vogelgesang *et al.*, 2016), *PyHST* (Mirone *et al.*, 2014), *PtyLab* (Loetgering *et al.*, 2023), *PyNX* (Favre-Nicolin *et al.*, 2020) and *PtyPy* (Enders & Thibault, 2016). Each of these has its own strengths and capabilities. With *HoToPy*, we contribute implementations of recent advances in phase retrieval (Faragó *et al.*, 2024; Huhn *et al.*, 2022), iterative tomographic alignment techniques and image processing tools, that to our knowledge are not yet available in other frameworks. We pair this with high level implementations of smooth and non-smooth optimization. Emerging from *HoloTomoToolbox* (Lohse *et al.*, 2020) for MATLAB, *HoToPy* not only serves as a drop-in replacement but extends it with recent methods in a more flexible and modular framework. By leveraging the Python ecosystem including the 'PyTorch' (Ansel *et al.*, 2024) library for GPU acceleration and automatic differentiation, *HoToPy* seamlessly integrates with other existing frameworks simplifying integration into existing analysis workflows and pipelines. It offers a broad spectrum of functionality for the entire range of data analysis, from image processing, phase retrieval and alignment to tomographic reconstruction, but also the simulation of experiments. The source code is licensed under GNU General Public License and publicly available, ref. HoToPy (2025); up-to-date installation instructions, detailed documentation and examples are provided therein.

Currently, *HoToPy* is rapidly evolving. In this manuscript, we showcase its current status and demonstrate its capabilities on experimental data taken on the *Göttingen instrument for nano-imaging with X-rays* (GINIX) operated by our research group at the P10 beamline of the PETRA III storage ring at DESY in Hamburg, Germany.

2. *HoToPy* toolbox

The *HoToPy* package is a Python toolbox for holographic and tomographic X-ray imaging. Its primary use is the reconstruction of propagation based X-ray phase contrast tomography—*holo-tomography*—data recorded at synchrotron or laboratory X-ray sources, but it can also be used for visible light or electron holographic imaging.

HoToPy is implemented using the machine learning library 'PyTorch' (Ansel *et al.*, 2024), which provides strong GPU acceleration and flexible automatic differentiation. High-performance tomographic primitives are provided through an interface to the *ASTRA* toolbox (Palenstijn *et al.*, 2013; van Aarle *et al.*, 2015; van Aarle *et al.*, 2016). *HoToPy* can either be used as a software library in reconstruction pipelines or it can be used interactively by the user, thanks to its sane defaults and intuitive interface. Furthermore, real experimental datasets for testing and development of novel algorithms are included.

2.1. Numerical concepts

HoToPy's flexibility, extensibility and high numerical performance resides on two core principles: the use of *state-of-the-art numerical optimization algorithms* and *automatic differentiation (AD)*, whose concepts are briefly introduced in the following. In order to ease adoption of these concepts we provide reusable optimization algorithms through the hotopy.optimize submodule.

State-of-the-art numerical optimization underpins *HoToPy*'s phase retrieval algorithms and enables fast and robust reconstructions with high numerical efficiency. We provide algorithms for smooth and non-smooth optimization, including a robust *proximal gradient method* (PGM) with backtracking line search and adaptive step sizes described by

Table 1
Overview of available phase retrieval methods for the holographic as well as the direct contrast regime.

Name	Class	Reference
Holographic regime		
Contrast transfer function (CTF)	`CTF`	Cloetens *et al.* (1999)
— constrained CTF	`CTF`	Huhn *et al.* (2022)
Intensity transfer function (ICT)	`ICT`	Faragó *et al.* (2024)
Nonlinear Tikhonov	`Tikhonov`	Huhn *et al.* (2022)
— (smoothed) TV regularization	`TikhonovTV`	Lucht *et al.* (unpublished)
Alternating projections (AP)	`AP`	Hagemann *et al.* (2018)
Direct contrast regime		
Paganin	`Paganin`	Paganin *et al.* (2002)
Generalized Paganin	`GeneralizedPaganin`	Paganin *et al.* (2020)
Bronnikov-aided correction	`BronnikovAidedCorrection`	De Witte *et al.* (2009)
Modified Bronnikov	`ModifiedBronnikov`	Groso *et al.* (2006)

Goldstein *et al.* (2014*a*), an accelerated *alternating direction method of multipliers* (ADMM) (Goldstein *et al.*, 2014*b*) and *fast iterative shrinkage-thresholding algorithm* (FISTA) (Beck & Teboulle, 2009). In particular, the ADMM algorithm is used for the constrained variant of the contrast transfer function (CTF) and intensity transfer function (ICT) phase retrieval algorithms and the PGM with automatic differentiation for Tikhonov and TikhonovTV, as described by Huhn *et al.* (2022).

Automatic differentiation (AD) is popular for training of (deep) neural networks (Paszke *et al.*, 2017; Baydin *et al.*, 2018), where it is used to dynamically compute the gradient of a scalar-valued objective or loss function. Thus, neither finite-difference approximation nor analytical derivation and explicit implementation of gradients is required. In general, this allows fast prototyping of new algorithms or adaptation of existing ones without the burden of manually deriving and updating explicit gradients with an insignificant overhead in performance.

In *HoToPy*, AD is automatically used within the gradient-based optimization algorithms, *e.g.* PGM and FISTA, and thus in the phase retrieval algorithms based on them, such as the Tikhonov algorithm.

Using AD provides a high degree of flexibility. For example, this allows extension or modification of the Tikhonov algorithm with *any* (smooth) regularization, such as the smoothed L^1-norm for a total variation (TV) regularization (Hansen *et al.*, 2021) used in TikhonovTV. Furthermore, *any* modification of the forward model can easily be combined with any other constraint or regularization.

2.2. Phase retrieval

Since the phase of an X-ray wavefront cannot be measured with current technology, a central step in holo-tomography is the computational reconstruction of quantitative phase and attenuation images from recorded near-field diffraction patterns, or (inline) *holograms*. This so-called 'phase problem' poses a nonlinear, ill-posed inverse problem. To solve this inverse problem, several algorithms have been developed incorporating different assumptions and priors, such as assuming short propagation distance, a single-material object (Paganin *et al.*, 2002) or an optically weak object (Cloetens *et al.*, 1999). Generally, in PBI two regimes can be distinguished characterized by a dimensionless quantity, the *Fresnel number*. It is defined by $\mathcal{F} = \sigma^2/z\lambda$, with reference length scale (*e.g.* sample diameter, structure size or pixel size) σ, propagation distance z between sample and detector, and wavelength λ. One regime is the so-called *direct contrast* or edge-enhancement regime, where $\mathcal{F} \cong 1$; the other is the *holographic regime*, where $\mathcal{F} \ll 1$. For each regime a range of different algorithms are available in *HoToPy*, see Table 1.

In *HoToPy*, holography related methods (propagation and phase retrieval) are provided through the hotopy.holo submodule. So far, it is focused on propagation based phase contrast in the direct-contrast as well as holographic regime. All algorithms for the holographic regime support imposing priors as object constraints, *e.g.* pixel-wise non-positivity of the phase or finite supports. An overview of the available phase retrieval algorithms at the time of writing is given in Table 1, while an updated list can be found in the online documentation. All are implemented with GPU computation support. Moreover, they allow for astigmatism, so that the effective propagation distance or equivalently the Fresnel number $\mathcal{F}$ can be different in the two directions orthogonal to the optical axis. This can, for example, be caused by anisotropic magnifications in the horizontal and vertical directions, see for example the case of Bragg magnifiers (Spiecker *et al.*, 2023). For detailed documentation of the algorithms we refer to the online documentation, example notebooks and their respective literature. In addition, a number of methods for preprocessing are provided. These include the automated removal of faulty pixels as well as principal component analysis based (Nieuwenhove *et al.*, 2015) and curvature based methods for empty-beam division.

2.3. Computed tomography

The tomographic methods are organized in the hotopy.tomo submodule. Through interfaces to the *ASTRA* toolbox (Palenstijn *et al.*, 2013; van Aarle *et al.*, 2015; van Aarle *et al.*, 2016), efficient (multi-)GPU reconstruction and projection algorithms in two and three dimensions are provided, including, for example, filtered back-projection (FBP), the cone-beam algorithm of Feldkamp, Davis and Kress (FDK),

and the simultaneous iterative reconstruction technique (SIRT). For both parallel beam and cone beam geometric models, the source, sample and detector can be positioned freely for each projection image. Thus, any inexactness of the tomographic trajectory can be incorporated directly into the geometric model. This is computationally more efficient than aligning and interpolating the projection images which also degrades image quality. The toolbox contains methods based on image registration (Guizar-Sicairos *et al.*, 2008) for determining deviations from an assumed tomographic trajectory.

The center of rotation (CoR) can be found by registering the shift between either two opposing projection images, or, for scans with an angular range larger than 0° to 180°, two opposite segments of a sinogram—ideally two half rotations—can instead be registered. In the latter, the CoR estimate becomes an average over all acquisition angles, making it more robust for acquisitions with little position precision. The registration can be repeated for sinograms from different detector rows to further increase robustness and also determine small tilts of the rotation axis.

The iterative *reprojection alignment* algorithm (van Leeuwen *et al.*, 2018) in the toolbox enables reconstructing and correcting rigid sample movement between individual projections during a tomographic scan. In each iteration, the volume is reconstructed based on the current geometry estimate. (Re-)Projection images are generated from the reconstructed volume, registered against the acquired images and the geometric model for the respective projection updated according to the detected shift. In practice it is often advisable to apply pixel binning and a bandpass filter to the projection images prior to the alignment routine to accelerate the computation and improve the registration. Nonlinearities in the detector response, but also illumination, can cause stripes in the sinograms which lead to ring artifacts in the reconstructed volumes. *HoToPy* contains implementations of additive (Ketcham, 2006) and wavelet based (Münch *et al.*, 2009) ring-removal algorithms to mitigate these artifacts.

3. Reconstruction example: catalytic particle

We demonstrate the *HoToPy* toolbox with the example of an X-ray holo-tomography dataset of an isolated catalytic particle with a diameter of approximately 33 µm used for olefin polymerization. The particle morphology and fragmentation ('cracks') is of particular interest. For an in-depth description of the sample, and how the inner structure matters for the catalytic function in the application context, we refer to Werny *et al.* (2022*a*,*b*) and Veselý *et al.* (2021). The particle is attached to the interior wall of a Kapton tube. The data were recorded at the P10 beamline of the PETRA III storage ring at DESY in Hamburg with the GINIX. The GINIX was used in its cone-beam configuration to achieve geometrical magnification and effective pixel sizes down to the nanometer range. To this end, the incident beam is focused with a Kirkpatrick–Baez (KB) mirror system onto an X-ray waveguide acting as quasi-point source for holographic illumination. The waveguide (ID 4743) is a combination of two orthogonally crossed thin-film waveguides with a diameter of 58 nm and depth of 600 µm each, functioning together as a two-dimensional waveguide (Krüger *et al.*, 2010). Two tomograms at two defocus distances were acquired with 1501 projections each, covering an angular range of 180°. The source-to-sample distance z_{01} was adjusted to 13.53 mm and 16.73 mm, at constant source-to-detector distance z_{02} = 5110 mm. The photon energy $E_{\rm ph}$ = 13.8 keV was selected by a Si(111) monochromator. Images were recorded using a Gadox scintillator of 15 µm thickness fiber-coupled to an Andor Zyla sCMOS sensor with a pixel size Δ_x of 6.5 µm and 2160 × 2560 pixels. The exposure time per acquisition was 1 s. The geometry corresponds to geometric magnifications of $M = z_{02}/z_{01}$ = 378 and 305, resulting in an effective pixel sizes of 17.2 nm and 21.3 nm, respectively. The Fresnel numbers (with respect to the effective pixel size), $\mathcal{F} = \Delta_x^2/[M\lambda(z_{02} - z_{01})]$, evaluate to 2.44×10^{-4} and 3.02×10^{-4}, respectively.

3.1. Phase retrieval

First, the phase of the catalytic particle has to be reconstructed by phase retrieval. To this end, the recorded raw intensity patterns are preprocessed by dark current subtraction and divided by interpolated empty beam images, *i.e.* images taken without a sample. The empty image $\imath$ used for normalization at the tomographic angle θ with range $[0, \theta_{\rm max}]$ is interpolated by a linear combination of the (average) empty images before $\imath_{\rm pre}$ and after $\imath_{\rm post}$ the tomographic scan, $\imath(\theta) = \imath_{\rm pre}(1 - \theta/\theta_{\rm max}) + \imath_{\rm post}\theta/\theta_{\rm max}$. The resulting normalized holographic diffraction patterns are also called (inline) *holograms*. Afterwards, residual low frequency background variations are suppressed by a least curvature inpainting of the background within a compact support of the particles. Prior to phase retrieval the holograms of the second distance are magnified to an effective parallel beam geometry with effective pixel size 17.2 nm and respective Fresnel numbers 2.44×10^{-4} and 1.98×10^{-4}.

An exemplary empty-beam divided hologram is shown in Fig. 2(*a*) together with different phase reconstructions in Figs. 2(*b*) and 2(*c*). Fig. 2(*b*) shows a single step phase reconstruction using the contrast transfer function (CTF) method (Cloetens *et al.*, 1999) without the use of any constraints. Figs. 2(*c*) and 2(*d*) show a non-linear and constraint-based phase reconstruction using the Tikhonov (Huhn *et al.*, 2022) algorithm with pixel-wise non-positivity, and non-positivity combined with a compact disk-shaped support, respectively. All reconstructions were computed with the input of the recordings of the two distances, assuming a homogeneous object with beta/delta ratio β/δ = 0.035 and applying a two-level frequency regularization using the weights $\alpha_{\rm low} = 2 \times 10^{-5}$, $\alpha_{\rm high} = 3 \times 10^{-5}$ (Huhn *et al.*, 2022).

By comparing the reconstructions, we can directly observe strong background variations in Figs. 2(*b*) and 2(*c*). In the lower right corner of Fig. 2(*c*) we can observe that this also affects the phase reconstruction within the particle. Furthermore, the linear and unconstrained reconstruction in Fig. 2(*b*)

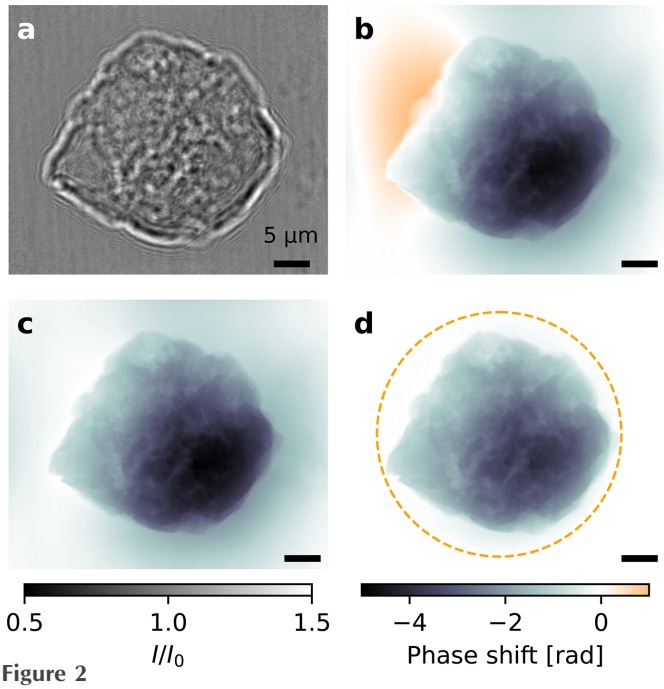

Figure 2
Hologram and phase reconstructions of a catalytic particle. (*a*) Example showing one of the two normalized holographic intensity interference patterns I/I_0 (hologram) of a catalytic particle at one tomographic angle. (*b*–*d*) Comparison of different phase retrieval methods and constraints. The reconstruction in (*b*) uses an unconstrained linear contrast transfer function (CTF). The reconstructions (*c*) and (*d*) are obtained using the HoToPy–Tikhonov algorithm. For these, a pixel-wise non-positivity constraint is used and for (*d*) additionally a finite disk-shaped support, indicated by the dashed circle. Scale bars: 5 µm. Effective pixel size: 17.2 nm. Images have 2160 × 2560 pixels.

incorrectly contains positive values; due to the convention used here a higher density sample relative to the empty beam should solely have non-positive values (negative phase shift or retarded waves). Thus, using this as prior knowledge for the reconstructions yields more faithful reconstructions [Fig. 2(*c*)]. Finally, if additionally combined with a disk-shaped support, low frequency background variations are effectively suppressed [Fig. 2(*d*)].

The corresponding code snippets used for the phase reconstructions are given in Appendix *A*. A comparison of computation times of two phase retrieval algorithms with different sets of constraints is given in Table 2 in Appendix *B*.

3.2. Tomography

A three dimensional model of the sample is created from the phase projections by means of tomographic reconstruction. A wavelet-based filter (Münch *et al.*, 2009) is applied to reduce stripes in the sinograms. The corresponding reduction of ring artifacts in the reconstructed slice can be seen in Fig. 3(*a*). The direct tomographic reconstruction of the sinograms [Fig. 3(*b*)] suffers from artifacts caused by inexactness of the acquisition trajectory, mainly due to sample movement. With a geometric model which includes the sample movement [Fig. 3(*c*)], the quality of the reconstruction improves drastically and fragmentation cracks can be traced [Fig. 3(*d*)]. To extract the sample movement from the acquired projection

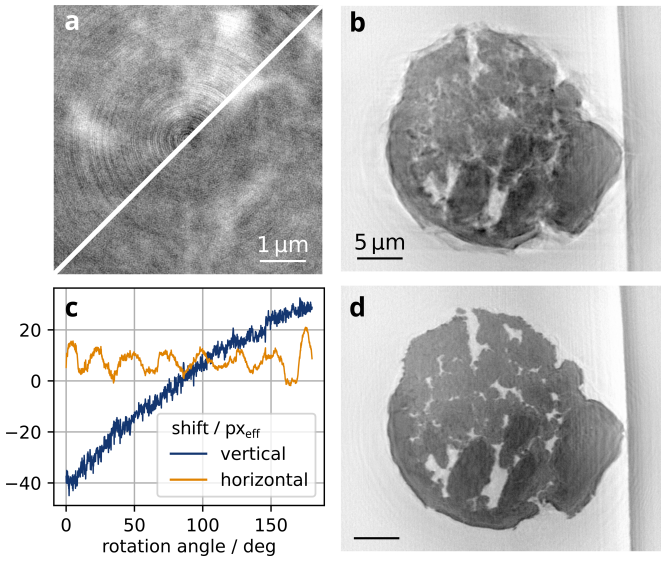

Figure 3
Tomographic reconstruction by the FDK algorithm. (*a*) Zoom into the center of a reconstructed horizontal slice. Concentric ring artifacts (top left) are mitigated (bottom right) after applying wavelet-based ring removal ($l = 4, \sigma = 1$) to the sinogram. (*b*) Vertical slice through a volume reconstruction assuming an idealized acquisition trajectory. (*c*) Shifts of the projection images estimated by registration of opposite projections and reprojection registration (8 × 8 pixel binning, high-pass filter with $\sigma = 5$, 100 iterations). (*d*) Virtual slice through the volume reconstruction after applying the shift correction. The effective voxel size is 17.2 nm.

images, the CoR and drift were first estimated by registration of opposite projections. The reprojection alignment algorithm was then applied to a reduced dataset which was obtained by 8 × 8 pixel binning and high-pass filtering with a Gaussian filter kernel (σ = 40/8). The influence of the Kapton tube's strong edge was reduced by a directional Fourier filter applied to images with observation angles close to the edge surface. The displayed shifts were obtained after 100 iterations, taking about 9.3 min on a machine with 24 CPU cores [Intel(R) Xeon(R) w7-3455] and an NVIDIA RTX 6000 Ada Generation GPU. The FDK reconstruction of the full volume takes about 2 min. The code snippets for the tomography reconstruction are given in Appendix *A*.

4. Conclusion

We have presented *HoToPy*, an open source toolbox for holographic and tomographic X-ray imaging in Python. It provides a collection of phase retrieval algorithms for the direct contrast and in-line holographic regime, suitable for propagation-based phase imaging at synchrotron and laboratory µCT instruments alike. Tomographic reconstruction and alignment methods are also included. It is easily extensible without sacrificing speed by leveraging automatic differentiation and GPU computation, facilitating rapid testing of novel algorithms. Compared with our current 'workhorse' – the *HoloTomoToolbox* for MATLAB – *HoToPy* not only integrates into the extensive ecosystem of scientific computing in Python but also features iterative tomographic reprojection alignment and new phase retrieval algorithms such as ICT

(Faragó *et al.*, 2024) and TikhonovTV, extending the approach of Huhn *et al.* (2022) to TV regularization. We have showcased the toolbox by reconstructing the volume from a deeply holographic and severely misaligned dataset of a catalytic particle.

The toolbox can serve at least four different purposes. First, it can be used for testing and development of algorithms, operated stand-alone on pre-recorded or simulated data, for example using the supplied phantoms. Second, in the same manner, it can serve classroom teaching and visualization. Third, it can be operated with a script-based analysis pipeline for a specific instrument such as the 'GINIX' instrument. In this case, wrapper scripts handling the instrument specific metadata and data formats, or default reconstruction parameters, can easily be written and supplied to the user. Fourth, algorithmic implementations and functions of interest can be integrated into entirely different reconstruction pipelines tailored to specific instruments and requirements. For example, integration and interoperability with other phase and tomographic reconstruction platforms such as CIL (Jørgensen *et al.*, 2021) should be possible with little effort. Finally, beyond its primary use as a tool for X-ray phase imaging, we envision it to be useful also for inline optical or electron holography. To this end, the generic formulation in units of pixel size, Fresnel number or real and imaginary part of the refractive index is particularly helpful.

APPENDIX *A*
Code examples from *HoToPy*

For the complete reconstruction script, full documentation and further examples, refer to HoToPy (2025). The given code examples in this manuscript are based on *HoToPy* version v0.21.

A source code snippet for phase reconstruction of a catalytic particle with *HoToPy* follow in Fig. 4.

A source code snippet for the tomographic reconstruction and reprojection alignment to correct the acquisition trajectory follows in Fig. 5.

APPENDIX *B*
Computations times for phase retrieval

We performed a comparison of the computation times for the phase retrieval listed in Table 2. Here we compared the CTF and HoToPy–Tikhonov algorithms with and without enforcing pixel-wise non-positive phase. Additionally, for the Tikhonov a disk-shaped object support was also tested. The single-angle reconstructions were performed on the dual distance holograms shown in Fig. 2 (2160 × 2560 pixels), which corresponds to a single angle of the tomogram. All computations were done with single precision floats. We used default settings in *HoToPy* with homogeneous object assumption with a beta/delta ratio $\beta/\delta = 0.0035$ and two-level frequency regularization $\alpha_{\rm low} = 2 \times 10^{-5}$, $\alpha_{\rm high} = 3 \times 10^{-5}$. The iterations were stopped with the default stopping conditions. For the single tomographic angle reconstruction the iterations needed to meet these condition are given. The phase reconstruction for the full dataset were performed sequentially for each tomographic angle on the GPU.

```python
from hotopy.holo import CTF, ICT, Tikhonov, Constraints

# [...] data loading and preprocessing

# phase retrieval parameters
shape = holograms.shape[-2:]
betadelta = 0.0035  # single material object
alpha = (2e-5, 3e-5)  # frequency regularization parameters
device = "cuda:0"  # torch.device string

# setup of phase retrieval algorithms
ctf = CTF(shape, fresnel_numbers, betadelta=betadelta, alpha=alpha, device=device)
tikhonov = Tikhonov(shape, fresnel_numbers, betadelta=betadelta, alpha=alpha,
        device=device)
ict = ICT(shape, fresnel_numbers, betadelta=betadelta, alpha=alpha, device=device)

# pixel-wise non-positivity constraint
c_nonpos = Constraints(phase_max=0.0)

# Figure b: linear CTF without constraints
rec_ctf = ctf(holograms).cpu()

# not shown: alternatively ICT without constraints
rec_ict = ict(holograms).cpu()

# not shown: CTF with non-positivity constraint
rec_ctf_nonpos = ctf(holograms, constraints=c_nonpos).cpu()

# Figure c: nonlinear reconstruction with non-positivity constraint
rec_tikhonv = tikhonov(holograms, constraints=c_nonpos).cpu()

# Figure d: same as above but additionally with support mask
# generate mask, where ball is non-zero
from hotopy.image import ball
radius = 980  # pixel
center = (shape[0]/2-0.5, shape[1]/2-50)
mask = ball(shape, radius, center=center) > 0  # binary mask

c_nonpos_support = c_nonpos * Constraints(support=mask)  # concatenation is supported
rec_tikhonov_support = tikhonov(holograms, constraints=c_nonpos_support).cpu()
```

Figure 4
Source code snippet for phase reconstruction of a catalytic particle with *HoToPy*.

```python
from hotopy import tomo, image
import numpy as np

# [...] load ‘projections‘ and define directional filter ‘filter_vertical_edge‘
#       which reduces artifacts caused by the strong kapton edge

# ringremoval
projections = tomo.ringremove(projections, type="wavelet", level=4, sigma=1)

# pixel binning
binning = 8
proj_small = image.AveragePool2d(binning)(projections)

# image filter
proj_small = filter_vertical_edge(proj_small)
sigma_low, sigma_high = 0, 40 / binning
proj_small = image.GaussianBandpass.apply(proj_small, sigma_low, sigma_high, pad="auto",
        ndim=2)
proj_small = proj_small[:,20:-20, 20:-20]  # crop borders

# CoR/drift estimation and reprojection alignment
shift, _ = image.register_images(proj_small[0], np.fliplr(proj_small[-1]).copy())
initial_shifts = np.c_[shift[1] / 2 * np.ones(len(angles)),             # horizontal
                       shift[0] / 2 * np.linspace(1, -1, len(angles))]  # vertical
z01, z02, px = 13.53e-3, 5110e-3, 6.5e-6  # in m
t = tomo.setup(proj_small.shape[-2:], angles, cone=(z01, z02, px * binning),
        sino_pad="corner")
rep_align = tomo.ReprojectionAlignment(t, proj_small)
rep_shifts = rep_align(initial_shifts, tol=0.01, max_iter=100, upsample=100) * binning

# reconstructions
t = tomo.setup(projections.shape[-2:], angles, cone=(z01, z02, px), sino_pad="corner")
direct_reconstruction = t.reconstruct(projections)
t.apply_shift(rep_shifts)
improved_reconstruction = t.reconstruct()
```

Figure 5
Source code snippet for the tomographic reconstruction and reprojection alignment to correct the acquisition trajectory.

Table 2
Compute times of two phase retrieval algorithms implemented in *HoToPy* with different sets of constraints.

Computations were performed on a NVIDIA RTX 6000 Ada and a NVIDIA H100 GPU with the same dataset shown in Fig. 2 and the full catalytic particle dataset, respectively.

Algorithm	RTX 6000 Ada	H100
Single tomographic angle, two distances		
CTF, no constraints (single-step method)	4.1 ms	3.9 ms
CTF, phase $\phi \leq 0$ (24 iterations)	38.5 ms	30.2 ms
Tikhonov, no constraints (25 iterations)	245.8 ms	217.9 ms
Tikhonov, phase $\phi \leq 0$ (25 iterations)	284.5 ms	233.8 ms
Tikhonov, phase $\phi \leq 0$ and disk-shaped support (30 iterations)	439.5 ms	317.6 ms
Full dataset of catalytic particle		
CTF, no constraints	9 s	6 s
CTF, phase $\phi \leq 0$	72 s	47 s
Tikhonov, no constraints	6.1 min	4.4 min
Tikhonov, phase $\phi \leq 0$	7.4 min	5.1 min
Tikhonov, phase $\phi \leq 0$ and disk-shaped support	11.8 min	7.3 min

Clearly, the linear CTF algorithm without constraints is the fastest in the comparison, since it is a single-step algorithm without iterations. When constraints are set, like the pixel-wise phase non-positivity used here, ADMM iterations are applied which takes approximately one order of magnitude longer than the single-step method. The HoToPy–Tikhonov algorithm takes again one order of magnitude longer to converge without constraints. If pixel-wise phase non-positivity is added, convergence is reached in a comparable time. This time is approximately doubled if phase non-positivity is combined with a disk-shaped support.

This benchmark gives an impression of the performance of *HoToPy*'s phase retrieval using a real dataset on standard hardware. When considering these results, one should keep in mind that the performance also strongly depends on the dataset, parameters and constraints.

Acknowledgements

The authors would like to thank Simon Huhn for inspiring discussions and his help analyzing the dataset, as well as Jan Goeman and Markus Osterhoff for their kind support of our in-house computing infrastructure. Moreover, we thank Maximilian J. Werny, Florian Meirer and Bert Weckhuysen for providing the catalytic particle. We acknowledge Deutsches Elektronen-Synchrotron (DESY) (Hamburg, Germany), a member of the Helmholtz Association HGF, for the provision of experimental facilities. Parts of this research were carried out at PETRA III and we would like to thank Michael Sprung and Fabian Westermeier for assistance in using the P10 beamline. Beam time was allocated for proposal II-20211052. This research was supported in part through the Maxwell computational resources operated at DESY. The authors are members of the Max Planck School of Photonics. Open access funding enabled and organized by Projekt DEAL.

Conflict of interest

The authors declare no conflicts of interest.

Data availability

The source code is openly available under the GPLv3 license at HoToPy (2025) and the data used for the examples can be obtained from Lucht *et al.* (2025).

Funding information

We acknowledge partial funding by Max Planck School of Photonics as well as Deutsche Forschungsgemeinschaft (DFG) (432680300 SFB 1456), and the German Minstry of Research and Technology for grant Holo-Tomogaphy (05K22MG1) and RECOX (05K25MG1) within the ErUM-Pro funding line.

References

Ansel, J., Yang, E., He, H., Gimelshein, N., Jain, A., Voznesensky, M., Bao, B., Bell, P., Berard, D., Burovski, E., Chauhan, G., Chourdia, A., Constable, W., Desmaison, A., DeVito, Z., Ellison, E., Feng, W., Gong, J., Gschwind, M., Hirsh, B., Huang, S., Kalambarkar, K., Kirsch, L., Lazos, M., Lezcano, M., Liang, Y., Liang, J., Lu, Y., Luk, C. K., Maher, B., Pan, Y., Puhrsch, C., Reso, M., Saroufim, M., Siraichi, M. Y., Suk, H., Zhang, S., Suo, M., Tillet, P., Zhao, X., Wang, E., Zhou, K., Zou, R., Wang, X., Mathews, A., Wen, W., Chanan, G., Wu, P. & Chintala, S. (2024). *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS2024)*, Vol. 2, 27 April–1 May 2024, San Diego, USA, pp. 929–947.

Astolfo, A., Endrizzi, M., Vittoria, F. A., Diemoz, P. C., Price, B., Haig, I. & Olivo, A. (2017). *Sci. Rep.* **7**, 2187.

Azevedo, A., Lesser, E., Phelps, J. S., Mark, B., Elabbady, L., Kuroda, S., Sustar, A., Moussa, A., Khandelwal, A., Dallmann, C. J., Agrawal, S., Lee, S. J., Pratt, B., Cook, A., Skutt-Kakaria, K., Gerhard, S., Lu, R., Kemnitz, N., Lee, K., Halageri, A., Castro, M., Ih, D., Gager, J., Tammam, M., Dorkenwald, S., Collman, F., Schneider-Mizell, C., Brittain, D., Jordan, C. S., Dickinson, M., Pacureanu, A., Seung, H. S., Macrina, T., Lee, W. A. & Tuthill, J. C. (2024). *Nature* **631**, 360–368.

Baydin, A. G., Pearlmutter, B. A., Radul, A. A. & Siskind, J. M. (2018). *J. Mach. Learn. Res.* **18**, 5595–5637.

Beck, A. & Teboulle, M. (2009). *2009 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 19–24 April 2009, Taipei, Taiwan, pp. 693–696.

Bellens, S., Guerrero, P., Vandewalle, P. & Dewulf, W. (2024). *CIRP J. Manuf. Sci. Technol.* **51**, 324–341.

Chushkin, Y. & Zontone, F. (2025). *IUCrJ* **12**, 280–287.

Cloetens, P., Ludwig, W., Baruchel, J., Van Dyck, D., Van Landuyt, J., Guigay, J. P. & Schlenker, M. (1999). *Appl. Phys. Lett.* **75**, 2912–2914.

Correa, J., Ignatenko, A., Pennicard, D., Lange, S., Fridman, S., Karl, S., Lohse, L., Senfftleben, B., Sergeev, I., Velten, S., Prajapat, D., Bocklage, L., Bromberger, H., Samartsev, A., Chumakov, A., Rüffer, R., von Zanthier, J., Röhlsberger, R. & Graafsma, H. (2024). *J. Synchrotron Rad.* **31**, 1209–1216.

De Witte, Y., Boone, M., Vlassenbroeck, J., Dierick, M. & Van Hoorebeke, L. (2009). *J. Opt. Soc. Am. A* **26**, 890–894.

Donath, T., Šišak Jung, D., Burian, M., Radicci, V., Zambon, P., Fitch, A. N., Dejoie, C., Zhang, B., Ruat, M., Hanfland, M., Kewish, C. M., van Riessen, G. A., Naumenko, D., Amenitsch, H., Bourenkov, G., Bricogne, G., Chari, A. & Schulze-Briese, C. (2023). *J. Synchrotron Rad.* **30**, 723–738.

Enders, B. & Thibault, P. (2016). *Proc. R. Soc. London A* **472**, 20160640.

Faragó, T., Gasilov, S., Emslie, I., Zuber, M., Helfen, L., Vogelgesang, M. & Baumbach, T. (2022). *J. Synchrotron Rad.* **29**, 916–927.

Faragó, T., Spiecker, R., Hurst, M., Zuber, M., Cecilia, A. & Baumbach, T. (2024). *Opt. Lett.* **49**, 5159.

Favre-Nicolin, V., Girard, G., Leake, S., Carnis, J., Chushkin, Y., Kieffer, J., Paleo, P. & Richard, M.-I. (2020). *J. Appl. Cryst.* **53**, 1404–1413.

Flenner, S., Bruns, S., Longo, E., Parnell, A. J., Stockhausen, K. E., Müller, M. & Greving, I. (2022). *J. Synchrotron Rad.* **29**, 230–238.

Goldstein, T., O'Donoghue, B., Setzer, S. & Baraniuk, R. (2014*b*). *SIAM J. Imaging Sci.* **7**, 1588–1623.

Goldstein, T., Studer, C. & Baraniuk, R. (2014*a*). *arXiv*:1411.3406.

Groso, A., Abela, R. & Stampanoni, M. (2006). *Opt. Express* **14**, 8103–8110.

Grote, L., Seyrich, M., Döhrmann, R., Harouna-Mayer, S. Y., Mancini, F., Kaziukenas, E., Fernandez-Cuesta, I. A., Zito, C., Vasylieva, O., Wittwer, F., Odstrčzil, M., Mogos, N., Landmann, M., Schroer, C. G. & Koziej, D. (2022). *Nat. Commun.* **13**, 4971.

Guizar-Sicairos, M., Thurman, S. T. & Fienup, J. R. (2008). *Opt. Lett.* **33**, 156–158.

Gürsoy, D., De Carlo, F., Xiao, X. & Jacobsen, C. (2014). *J. Synchrotron Rad.* **21**, 1188–1193.

Hagemann, J., Töpperwien, M. & Salditt, T. (2018). *Appl. Phys. Lett.* **113**, 041109.

Hansen, P. C., Jørgensen, J. & Lionheart, W. R. (2021). *Computed tomography: algorithms, insight, and just enough theory*. SIAM.

Hatsui, T. & Graafsma, H. (2015). *IUCrJ* **2**, 371–383.

Hendriksen, A. A., Pelt, D. M. & Batenburg, K. J. (2020). *IEEE Trans. Computat. Imaging* **6**, 1320–1335.

Hoeppe, H. P., Osterhoff, M., Aghel Maleki, A., Rosselló, J. M., Vassholz, M., Hagemann, J., Engler, T., Schwarz, D., Rodriguez-Fernandez, A., Boesenberg, U., Möller, J., Shayduk, R., Hallmann, J., Madsen, A., Mettin, R. & Salditt, T. (2024). *New J. Phys.* **26**, 033002.

HoToPy (2025). *HoToPy*, https://gitlab.gwdg.de/irp/hotopy.

Huhn, S., Lohse, L. M., Lucht, J. & Salditt, T. (2022). *Opt. Express* **30**, 32871–32886.

Jørgensen, J. S., Ametova, E., Burca, G., Fardell, G., Papoutsellis, E., Pasca, E., Thielemans, K., Turner, M., Warr, R., Lionheart, W. R. B. & Withers, P. J. (2021). *Philos. Trans. R. Soc. A.* **379**, 20200192.

Ketcham, R. (2006). *Proc. SPIE* **6318**, 63180O.

Krüger, S. P., Giewekemeyer, K., Kalbfleisch, S., Bartels, M., Neubauer, H. & Salditt, T. (2010). *Opt. Express* **18**, 13492.

Langer, M., Zhang, Y., Figueirinhas, D., Forien, J.-B., Mom, K., Mouton, C., Mokso, R. & Villanueva-Perez, P. (2021). *J. Synchrotron Rad.* **28**, 1261–1266.

Li, P., Allain, M., Grünewald, T. A., Rommel, M., Campos, A., Carbone, D. & Chamard, V. (2022). *Light Sci. Applic.* **11**, 73.

Livingstone, J., Bosch, C., Kuan, A. T., Benoït, L., Busca, P., Martin, T., Mazri, M. F., Xiang, W., Allen Lee, W.-C., Schaefer, A. T., Cloetens, P. & Pacureanu, A. (2025). *Biomed. Opt. Expr.* **16**, 2047–2060.

Loetgering, L., Du, M., Boonzajer Flaes, D., Aidukas, T., Wechsler, F., Penagos Molina, D. S., Rose, M., Pelekanidis, A., Eschen, W., Hess, J., Wilhein, T., Heintzmann, R., Rothhardt, J. & Witte, S. (2023). *Opt. Express* **31**, 13763–13797.

Lohse, L. M., Robisch, A.-L., Töpperwien, M., Maretzke, S., Krenkel, M., Hagemann, J. & Salditt, T. (2020). *J. Synchrotron Rad.* **27**, 852–859.

Lucht, J., Meyer, P. & Salditt, T. (2025). *X-ray phase contrast nanotomography of an olefin catalytic particle*, https://doi.org/10.25625/CQ1EKY.

Mirone, A., Brun, E., Gouillart, E., Tafforeau, P. & Kieffer, J. (2014). *Nucl. Instrum. Methods Phys. Res. B* **324**, 41–48.

Momose, A., Yashiro, W., Kuwabara, H. & Kawabata, K. (2009). *Jpn. J. Appl. Phys.* **48**, 076512.

Montgomery, D. S. (2023). *Rev. Sci. Instrum.* **94**, 021103.

Münch, B., Trtik, P., Marone, F. & Stampanoni, M. (2009). *Opt. Express* **17**, 8567–8591.
Nikitin, V. (2023). *J. Synchrotron Rad.* **30**, 179–191.
Paganin, D. (2006). *Coherent X-ray Optics*. Oxford University Press.
Paganin, D., Mayo, S. C., Gureyev, T. E., Miller, P. R. & Wilkins, S. W. (2002). *J. Microsc.* **206**, 33–40.
Paganin, D. M., Favre-Nicolin, V., Mirone, A., Rack, A., Villanova, J., Olbinado, M. P., Fernandez, V., da Silva, J. C. & Pelliccia, D. (2020). *J. Opt.* **22**, 115607.
Palenstijn, W. J., Batenburg, K. J. & Sijbers, J. (2013). *Proceedings of the 13th International Conference on Computational and Mathematical Methods in Science and Engineering (CMMSE)*, 24–27 June 2013, Almería, Spain, pp. 1139–1145.
Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. & Lerer, A. (2017). *Autodiff Workshop* at the *31st Conference on Neural Information Processing Systems (NIPS2017)*, Long Beach, CA, USA (https://openreview.net/references/pdf?id=SJTr5QMRZ).
Pfeiffer, F. (2018). *Nat. Photon.* **12**, 9–17.
Pfeiffer, F., Herzen, J., Willner, M., Chabior, M., Auweter, S., Reiser, M. & Bamberg, F. (2013). *Z. für Med. Phys.* **23**, 176–185.
Quenot, L., Bohic, S. & Brun, E. (2022). *Appl. Sci.* **12**, 9539.
Robisch, A.-L., Kröger, K., Rack, A. & Salditt, T. (2015). *New J. Phys.* **17**, 073033.
Rodenburg, J. M. & Faulkner, H. M. (2004). *Appl. Phys. Lett.* **85**, 4795–4797.
Salditt, T., Egner, A. & Luke, D. R. (2020). *Nanoscale photonic imaging*. Springer Nature.
Spiecker, R., Pfeiffer, P., Biswal, A., Shcherbinin, M., Spiecker, M., Hessdorfer, H., Hurst, M., Zharov, Y., Bellucci, V., Faragó, T., Zuber, M., Herz, A., Cecilia, A., Czyzycki, M., Dias, C. S. B., Novikov, D., Krogmann, L., Hamann, E., van de Kamp, T. & Baumbach, T. (2023). *Optica*, **10**, 1633.
van Aarle, W., Palenstijn, W. J., Cant, J., Janssens, E., Bleichrodt, F., Dabravolski, A., De Beenhouwer, J., Joost Batenburg, K. & Sijbers, J. (2016). *Opt. Express* **24**, 25129.
van Aarle, W., Palenstijn, W. J., De Beenhouwer, J., Altantzis, T., Bals, S., Batenburg, K. J. & Sijbers, J. (2015). *Ultramicroscopy* **157**, 35–47.
van Leeuwen, T., Maretzke, S. & Batenburg, K. J. (2018). *Inverse Probl.* **34**, 024004.
Van Nieuwenhove, V., De Beenhouwer, J., De Carlo, F., Mancini, L., Marone, F. & Sijbers, J. (2015). *Opt. Express* **23**, 27975.
Veselý, M., Valadian, R., Lohse, L. M., Toepperwien, M., Spiers, K., Garrevoet, J., Vogt, E. T. C., Salditt, T., Weckhuysen, B. M. & Meirer, F. (2021). *ChemCatChem* **13**, 2494–2507.
Vijayakumar, J., Dejea, H., Mirone, A., Muzelle, C., Meyer, J., Jarnias, C., Dollman, K., Zabler, S., Paolasini, L., Bellier, A., *et al.* (2024). *Synchrotron Radiat. News* **37**(5), 16–25.
Vogelgesang, M., Farago, T., Morgeneyer, T. F., Helfen, L., dos Santos Rolo, T., Myagotin, A. & Baumbach, T. (2016). *J. Synchrotron Rad.* **23**, 1254–1263.
Werny, M. J., Müller, D., Hendriksen, C., Chan, R., Friederichs, N. H., Fella, C., Meirer, F. & Weckhuysen, B. M. (2022*a*). *ChemCatChem* **14**, e202200067.
Werny, M. J., Siebers, K. B., Friederichs, N. H., Hendriksen, C., Meirer, F. & Weckhuysen, B. M. (2022*b*). *J. Am. Chem. Soc.* **144**, 21287–21294.
Zanette, I., Zhou, T., Burvall, A., Lundström, U., Larsson, D. H., Zdora, M., Thibault, P., Pfeiffer, F. & Hertz, H. M. (2014). *Phys. Rev. Lett.* **112**, 253903.