SANDIA REPORT
SAND2024-14381R
Printed October 2024

Sandia National Laboratories

Ultra-High-Speed X-ray Tomography: Bridging the Gaps to See the Unknown

Benjamin R. Halls, Roderick R. La Foy, John C. Miers and Peggy J. Christenson, Alan Kastengren

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico
87185 and Livermore,
California 94550

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology & Engineering Solutions of Sandia, LLC.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@osti.gov
Online ordering: http://www.osti.gov/scitech

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5301 Shawnee Rd
Alexandria, VA 22312

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.gov
Online order: https://classic.ntis.gov/help/order-methods/

ABSTRACT

Rapid, time-varying, three-dimensional physics underpin numerous engineering challenges. Often, these physics occur within opaque environments, internal to a component, severely limiting applicable diagnostics. Development of novel diagnostics is necessary to understand and predict transient three-dimensional (3D) phenomena within opaque environments. This report highlights progress in four key areas leading to advancements in high-speed X-ray radiography and tomography. The first area is enabling MHz-rate imaging of energetics at the Advanced Photon Source at Argonne National Laboratory. The second is modeling a high-flux, rotating-anode X-ray source to understand the heat loads on the anode. The third effort was to develop a novel reconstruction algorithm that is validated by ground experimental tomography data and synthetic tomography data. The fourth is the development of a novel approach to two-color X-ray imaging.

ACKNOWLEDGEMENTS

This project was collaboration across several groups at Sandia, Universities and Laboratories. The team would like to thank Enrico C. Quintana, Mark E. Savage, John C. Miers, Matt Jordan, Edward S. Jimenez, and Kyle P. Lynch.

CONTENTS

ACRONYMS AND TERMS

Acronym/Term	Definition
2D Resolution	Spatial resolution in the two-dimensional radiographs
3D Resolution	Spatial resolution in the three-dimensional volumes
4D Resolution	Spatial resolution in the three-dimensional volumes and time
ART	Algebraic Reconstruction Technique
ASTRA	ASTRA Toolbox is a MATLAB and Python toolbox for 2D and 3D tomography
Beam Hardening	Spectral dependencies of the imaging system that favor higher energy X-rays. Includes the preferential attenuation of softer X-rays by material such that the transmitted X-ray spectrum is shifted towards the higher energies, and scintillator emission is a linearly proportional to X-ray photon energy.
Beam Softening	Spectral dependencies of the imaging system that favor lower energy X-rays. The preferential photoelectric absorption of softer X-rays by the scintillator such that the spectrum is shifted towards the lower energies.
CAD	Computer-Aided Design
Calibration (extrinsic)	The relation of different imaging systems including the rotations and translations with respect to the world coordinate system. The world coordinate system is a singular system where all objects and components can be referenced.
Calibration (intrinsic)	Relation of the image pixel coordinates to the coordinates in a reference frame. The intrinsic camera parameters are the focal length, geometric magnification, radial distortion, perspective distortion, and image center.
CDF	Cumulative Distribution Function
CNR	Contrast to Noise Ratio, measure of attenuated intensity vs fluctuations in intensity
CTF	Contrast Transfer Function, measure of image contrast vs random temporal fluctuations of pixel or voxel intensity
Dark Field	Image intensity that is not a result of the intended X-ray source. Camera image when X-ray source is off.
EM	Electro-Magnetic
FBP	Filtered Back Projection
FISTA	Fast Iterative Shrinkage-Thresholding Algorithm
Flat Field	Image intensity that is only a result of the intended X-ray source. Camera image when X-ray source is on and the FOV is empty. Commonly called “bright field” or “background”.
FOV	Field of View, region captured by the X-ray beams
GAGG	Gadolinium Aluminum Gallium Garnet (Gd3Al2Ga3O12)
Geometric Blur	Loss of spatial resolution due to the size of the X-ray focal spot, also known as penumbra.
Geometric Magnification	Magnification of the X-ray image due to the cone beam of the source
Hard X-rays	Higher energy X-rays, a relative term

Acronym/Term	Definition
Hot Pixel	Dynamic camera pixel saturation due to interaction with X-ray photon (zinger).
Image Parallax	Displacement in the position of an object image from multiple points of view
ITS	Integrated Tiger Series
KI	Potassium Iodide
LoS	Line of Sight
MC	Monte Carlo
MART	Multiplicative Algebraic Reconstruction Technique
MLEM	Maximum-Likelihood Expectation-Maximization
MLOS	Multiplicative Line-of-Sight
NeRF	Neural Radiance Field
NIRT	Neural Iterative Reconstruction Technique
NIST	National Institute of Standards and Technology
NRMSE	Normalized Root Mean Square Error
Penumbra	The partially shaded outer region of the radiograph due to the finite focal spot size
PETN	Pentaerythritol tetranitrate
Phantom	Known object used to determine reconstruction error
PIC	Particle-in-Cell
Primitives	Simplest geometric shape that the system can store
SART	Simultaneous Algebraic Reconstruction Technique
SCL	Space Charge Limit
SDD	Source-to-Detector Distance
Sinogram	Integral transform
SIRT	Simultaneous Iterative Reconstruction Technique
SLA	Stereolithography
SNR	Signal to Noise Ratio, measure of intensity vs random temporal fluctuations of pixel or voxel intensity
SOD	Source-to-Object Distance
Soft X-rays	Lower energy X-rays, a relative term
STL	Standard Triangle Language
TLD	Thermoluminescent Dosimeter
TV	Total Variation
Umbra	The fully shaded outer region of the radiograph
Voxel	Volume element, in relation to picture element or pixel
World Coordinates	A coordinate system that is common to all imaging systems

1. INTRODUCTION

X-ray diagnostics are well-established and ubiquitous across many Sandia. Current hardware (developed for medical and industrial applications) however, is inadequate to resolve 3D dynamics due to microsecond temporal scales and/or sub-millimeter spatial scales of the applications [1–3]. For transient events, X-ray tomography is restricted by the current lack of well-suited, high-speed X-ray sources and reconstruction algorithms specifically tailored for temporally resolved but view-limited data. Limitations of current state-of-the-art sources include laboratory-scale, laser-driven sources that are too large or too weak [4–6]; pulsed sources that are limited to only a few frames, therefore temporal dynamics are coarsely resolved [7–9]; large-scale synchrotrons that are by no means field transportable limiting their use to many applications [10–16]; liquid-metal-jet anodes produce a high fluence but lack the total flux required for highly dynamic events [17,18]; and current quasi-continuous wave (quasi-CW) X-ray sources, derived from medical computed tomography (CT) technologies, do not meet the high-flux requirements needed for short-duration events [1–3]. In addition, currently available reconstruction algorithms fail to accurate reconstruct complex scenes with image blur [19–34].

High-powered, quasi-continuous (quasi-CW) X-ray sources [1–3] are the ideal choice for kHz-rate cinematography because of their flux. However, they suffer from limitations: 1) their output was designed for medical and commercial applications, whose dynamics are typically much slower than many applications; 2) the spatial resolution is restricted by the ~1 mm focal spot size; and 3) their physical size limits the number of sources an experiment can utilize to ~5, introducing challenges to tomographic reconstruction accuracy. Although the X-ray flux is capable of imaging fragment motion, temporal blur is present. The spatial resolution limits the system's use for smaller devices. The 2D path-averaged image challenges the interpretation of the 3D scene.

Current state-of-the-art X-ray sources include laser-driven sources [4–6], flash or pulsed sources [7–9], large-scale synchrotrons [10–16], and current quasi-continuous wave (quasi-CW) X-ray sources [1–3]. While synchrotron facilities and laser-driven X-ray sources can produce very high-quality radiographs, due to their size tomography measurements are impractical or limited to time- or phase-averaged events [13]. CW sources using static anodes [10] or liquid-metal-jet anodes [17,18] do not provide sufficient flux for high-speed X-ray imaging of components centimeters in size. The single-shot operation of flash sources limits their use to only a few dynamic frames [7–9].

State-of-the-art synchrotron imaging includes the use of two X-ray generation methods: bending magnets and insertion devices. Bending magnets (BM) offer a larger field of view with lower available flux [14] compared to insertion devices (ID) which act as several bending magnets resulting in an increase in flux and reduction in field of view [11]. The NNSA ID beamline at the Advanced Photon Source (APS) at Argonne National Laboratory has yielded many studies relevant to mission [13,16]. However, to our knowledge other BM beamlines which could image larger scenes have been under-utilized by NNSA researchers.

The state-of-the-art in CT reconstruction falls into two categories: dense CT and limited-view CT. Traditional dense CT systems provide only time- or phase-averaged data; these measurements require 100s to 1000s of views and are amenable to volume reconstruction using back-projection methods [19–22]. These methods, in brief, project the information from the images back through the volume requiring a large number of views to achieve convergence and high spatial resolution. High-speed tomography is inherently limited by the number of possible viewing angles based on equipment limitations (size and cost). Mathematically, these limitations yield an underdetermined reconstruction problem, meaning that there are multiple 3D density fields that can fully explain any finite set of two-dimensional X-ray radiographs. Iterative reconstruction techniques have been developed for limited-

view measurements [23–27, 30–37]. For basic iterative methods, the main basis is to generate a volume that has projections that match the images along the same lines of sight, resulting in significant errors [30]. The final volume can be binarized, ensuring steep gradients, but not necessarily accurate volumes using a variety of methods [31–33]. Therefore, blind application of current reconstruction methods could lead to false interpretation of internal component dynamics, underpinning the need for a validated reconstruction approach. The basic algorithms are available via the Livermore Tomography Toolkit (LTT) [24] and commercial software [25,26], but do not satisfy Sandia's requirements. Approaches that incorporate a priori knowledge of the scene such as using probability distribution functions have been described in the literature [34–37].

2. OVERVIEW OF PROJECT SUBDIVISIONS

This report is subdivided into six tasks. Each task is descried in an overview in this section, and the complete details on each task can be found in the six appendices.

2.1. Synchrotron Measurements

Synchrotron-based X-ray measurements become necessary when the desired spatial resolution is below the limit of high-speed X-ray sources (~100 μm) and the temporal resolution is in excess of micro-focus X-ray sources (~1 kHz). The bending magnet beamlines at the Advanced Photon Source at Argonne National Laboratory have a beam size of approximately 25 × 5 mm^2 and the X-ray pulses (in 24-bunch) mode had a frequency of ~6.5 MHz. These characteristics satisfied the requirements to image a detonator at the earliest times, from bridge wire explosion to fragment breakout. The challenge to capturing such images is the Shimadzu HPV-X2 has limited available frame rates at the relevant speeds (2, 5, and 10 Hz). Despite this limitation measurements were performed at the 7-BM beam line at the Advanced Photon Source. The experiments coupled a Shimadzu camera operating at 5 MHz with the beam frequency at 6.5 MHz. This combination did pose challenges but also represented an experiment with the camera operating at 10 MHz and an X-ray pulse frequency of ~13 MHz. The specifics of the experiment are discussed in Appendix A.

2.2. X-ray Source Model

State-of-the-art quasi-CW X-ray sources are composed of the high-voltage generator and a single X-ray head and therefore provided a method to collect many images from a single focal spot location. However, they run for a relatively long (500 ms) duration at maximum power; It is hypothesized that the full potential of these sources has not been met with respect to ultra-high-speed applications that may last only 1 millisecond or less. To address this, several models were developed to analyze the potential for the sources to be over driven. The phenomena limiting the X-ray output are the space-charge limit of the electron beam, heat in cathode, and heating in the anode.

Computer models were developed to probe the possible operational space of the X-ray output and time pulses from a state-of-the-art quasi-CW commercial X-ray source. Such X-ray sources are composed of a high-voltage generator, a vacuum X-ray tube and the housing containing the X-ray tube. The intent of the modeling effort was to determine if such a commercial device could be adapted to function at higher voltage, with shorter time pulse, and with increased X-ray output. The phenomena limiting X-ray output in such devices are the space-charge limit of the electron beam, heat in cathode, and heating in the anode. The computer models utilized to determine the limiting factors were a self-consistent electromagnetic particle-in-cell (PIC) code that was coupled to electron/photon radiation transport simulation tools. The geometry modeled was based on readily available geometric depictions of commercially available X-ray devices such as what can be found in product operating manuals and, thus, was generic, but expected to be accurate enough to determine whether the X-ray device might be capable of functioning at increased voltage and electron current. Thus, the objectives of the study were to determine the maximum electron currents that could be transported in the tube diode and characterize the X-ray output with a goal of achieving reasonable agreement with experimental X-ray characteristics.

2.3. Neural Network Reconstruction Algorithm

The neural-implicit reconstruction technique (NIRT) uses a coordinate neural network to represent the solution, which enables the use of machine learning tools (i.e., programming environments,

optimization techniques, and hardware) to estimate the unknown density field. These algorithms can incorporate sophisticated forms of prior information, including symmetries, conservation principles, and transport physics to enhance the accuracy of the reconstructions. Moreover, scan data from previous measurements of the test article, or similar objects, can be included in a statistical manner, and a design specification can be used to anchor the solution. By including this information into the reconstruction procedure, the NIRT is advancing the state of the art in X-ray tomography, especially for time-resolved measurements, increasing the accuracy, spatial resolution, and temporal resolution of reconstructions. Our team has developed a suite of NIRT algorithms that enable faster and more robust time-resolved X-ray tomography measurements.

2.4. Benchmark Data: Experimental

A series of benchmark X-ray experiments was implemented to validate the accuracy and uncertainty of the reconstructions of high-speed object mass and/or shape dynamics. Mass conservation experiments utilized a droplet generator to quantify the amount of fluid entering the field of view, and a counter-propagating air jet was used to disrupt the droplets and generate highly dynamic events. Dynamic events with solid materials were generated using precise 3D phantoms, printed via stereolithography (SLA), with a minimum feature size of 150 μm. Reconstructions can then be verified with computer-aided design (CAD) models at the precision of the printing process. Finally, to evaluate the effects of X-ray imaging dynamic range on the reconstructions, objects with higher aspect ratios and a wider range of optical densities were evaluated, including ringed objects with hollow centers. Solid objects were imaged during falling and rotation to introduce temporal dynamics and help investigate methods for improving reconstruction accuracy, i.e., by using a 4D approach instead of a 3D approach.

2.5. Benchmark Data: Simulated

An image simulation software package was developed that generates high-fidelity, physics-based X-ray images that closely approximate experimental results. This package is used for validating experimental X-ray setups prior to tests as well as for providing benchmark datasets with known true solutions for developing X-ray image processing algorithms and tomographic reconstruction algorithms. The simulated imaging system is divided into X-ray sources, X-ray filters, imaging objects, scintillators, and cameras. Each component is described by a list of physically relevant parameters that are passed into the simulation software along with files containing X-ray spectrums, material absorption rates, and CAD files describing the imaging objects. The output images are generated by performing Monte Carlo ray-tracing calculations simulating the propagation of the X-ray photons to the scintillators and visible light photons to the cameras. The photon spatial and energy distributions are generated using models of known physical processes and attenuation of the photons is calculated from a material absorption database. Common sources of X-ray imaging error including quantum noise, geometric blur, and finite scintillator resolution are included in the simulation. Camera noise models are then applied to the generated sensor images to produce the final output images.

2.6. Two-Color X-ray Imaging

A two-color X-ray imaging methodology was developed to identify materials within an imaging dataset from a list of known and characterized materials. The methodology requires collecting calibration data at two different X-ray energy levels to relate the image intensity and material thicknesses for a given X-ray source, energy level, imaging detector, and specific materials. This data is fit to attenuation calibration functions so that given set of intensity values, the material thicknesses of imaged objects

can be estimated. Then objects in an image dataset are identified and sets of image intensity values are extracted for each X-ray energy. These intensity values are used to estimate the most likely matching material by performing a least-squares comparison between the measured intensity data and expected intensity values given by the calibration curves. The methodology was validated using a physics-based X-ray ray-tracing simulation. Additionally, preliminary material characterization experiments were completed and compared to the simulations. The simulations showed that three different materials could be reliably differentiated. The preliminary experimental data suggests that this result will hold in real-world measurements as well. The results of this work are not in the appendix but can be found in La Foy and Halls, 2023 [38].

3. PROJECT LEGACY

3.1. Intellectual Property

Technical Advancement: XTRaHS - X-ray Tomography and Radiography at High-Speeds

3.2. Prepared Publications

R. J. Sirimanne, R. Tang, J. P. Molnar, R. R. La Foy, B. R. Halls, S. J. Grauer, "Information Content of Radiography Data and Penalty Terms in 4D X-ray Tomography," (In Preparation).

R. J. Sirimanne, J. J. James, T. R. Meyer, B. R. Halls, S. J. Grauer, "4D Neural Reconstruction and Pose Estimation from Mono-, Stereo-, and Triscopic Radiographs," (In Preparation).

B. R. Halls, J. W. James, R. R. La Foy, N. Rahman, R. T. Marinis, W. Kuhns, P. Vinyard, A. L. Kastengren, "5 Megahertz-rate, White-beam X-ray Radiography of Energetic Events," (In Preparation).

R. R. La Foy, Benjamin R. Halls, "Two Color X-ray Material Determination," (In Preparation).

R. R. La Foy, Benjamin R. Halls, "Uncertainty Quantification in Multi-spectral X-ray Material Identification," (In Preparation).

R. R. La Foy, Benjamin R. Halls, "Camera and Scintillator Based X-ray Imaging Simulation Software," (In Preparation).

J. W. James, B. R. Halls, T. R. Meyer, "Improving 3-View X-ray Tomography Reconstructions using Temporal Information of Dynamic Events," In Preparation

J. W. James, B. R. Halls, T. R. Meyer, "Mass conservation error analysis using 3-view X-ray Tomography with droplet and counter-propagating air jet," In Preparation

3.3. Presentations

P. J. Christenson, B. R. Halls, E. C. Quintana, K. L. Cartwright, K. R. Depriest, M. E. Savage, "Characterization of an Over-Driven Commercial X-ray Tube using Coupled 3D Electromagnetic Particle-In-Cell to Monte-Carlo Radiation Transport Simulations," ICOPS 2023.

R. R. La Foy, B. Halls, "Uncertainty Quantification in Multi-spectral X-ray Material Identification," Laser Diagnostics in Energy and Combustion Science, Gordon Research Conference. July 2023.

R. R. La Foy, B. Halls, "Two-Color X-ray Imaging for Material Differentiation in Multiphase Flows," ILASS-Americas 33rd Annual Conference on Liquid Atomization and Spray Systems. May 2023.

J. W. James, B. R. Halls, T. R. Meyer, "Mass conservation error analysis using 3-view X-ray Tomography with levitating droplet," In Preparation for ILASS 2025.

3.4. Career Development

This project supported the career development of students, post-docs, and early career staff.

- John Miers: Postdoc to Staff
- Roderick La Foy: Hired as Postdoc
- PI Benjamin Halls: Started project in his 3rd year at Sandia, converted from SMTS to PMTS

- Jeremy James: Purdue Student
- Ryan Sirimanne. Penn State Student

3.5. University and National Laboratory Partnerships

- Academic Alliance (AA) with Purdue University
- Subcontract with Pennsylvania State University
- Collaboration with Argonne National Laboratory via safety meetings to enable energetic research and beamtime at the Advanced Photon Source at Sector 7-BM

REFERENCES

[1] Halls, B.R., J.R. Gord, L.E. Schultz, W.C. Slowman, M.D.A. Lightfoot, S. Roy and T.R. Meyer (2018), Quantitative 10–50 kHz X-ray radiography of liquid spray distributions using a rotating-anode tube source, Int. J. Multiph. Flow, 109, doi:10.1016/j.ijmultiphaseflow.2018.07.014.

[2] Halls, B.R., N. Rahman, M.N. Slipchenko, J.W. James, A. McMaster, M.D.A. Lightfoot, J.R. Gord and T.R. Meyer (2019), 4D spatiotemporal evolution of liquid spray using kilohertz-rate X-ray computed tomography, Opt. Lett., 44(20), doi:10.1364/OL.44.005013.

[3] Jones, E., E. Quintana, P. Reu and J. Wagner (2019), X-ray Stereo Digital Image Correlation, Exp. Tech. 44, doi:10.1007/s40799-019-00339-7.

[4] Achterhold, K., M. Bech, S. Schleede, G. Potdevin, R. Ruth, R. Loewen, and F. Pfeiffer (2013), Monochromatic computed tomography with a compact laser-driven X-ray source, Sci. Rep., 3, doi:10.1038/srep01313.

[5] Rambo, P., J. Schwarz, M. Schollmeier, M. Geissel, I. Smith, M. Kimmel, C. Speas, J. Shores, D. Armstrong, J. Bellum, E. Field, D. Kletecka and J. Porter (2016), Sandia's Z-Backlighter Laser Facility, Proc. SPIE 10014, Laser-Induced Damage in Optical Materials, doi:10.1117/12.2245271.

[6] Guénot, D., K. Svendsen, J. Björklund Svensson, H. Ekerfelt, A. Persson, O. Lundh, and E. Berrocal (2020), Simultaneous laser-driven X-ray and two-photon fluorescence imaging of atomizing sprays, Optica, 7(2), doi:10.1364/OPTICA.378063.

[7] Moser, S., S. Nau, M. Salk and K. Thoma (2014), In situ flash X-ray high-speed computed tomography for the quantitative analysis of highly dynamic processes, Meas. Sci. Tech., 25, doi:10.1088/0957-0233/25/2/025009.

[8] Wagner, J.L., S.P. Kearney, S.J. Beresh, E.P. DeMauro, and B.O. Pruett (2015), Flash X-ray measurements on the shock-induced dispersal of a dense particle curtain, Exp Fluids, 56(213) doi:10.1007/s00348-015-2087-3

[9] Zellner, M.B., K. Champley, L. McMichael, H. Martz, R. Cantrell, C.E. Yonce, K.W. Dudeck, C.A. Benjamin, R.W. Borys, D.R. Schall, Allen P. Ducote III, T.J. O'Connor, T.E. Nellenbach, N.J. Sturgill, T.L. Quigg, S.T. Halsey, J.A. Benjamin, and B.P. Huntzinger (2018), Development of a multi-energy flash computed tomography diagnostic for three dimensional imaging of ballistic experiments, AIP Conf. Proc., 1979, 160032, doi:10.1063/1.5045031.

[10] Halls, B.R. (2014), X-ray radiography and fluorescence for liquid distribution and mixing measurements in impinging jet sprays, Graduate Theses and Dissertations, 14156, doi:10.31274/etd-180810-3703.

[11] Jensen, B.J., K.J. Ramos, A.J. Iverson, J. Bernier, C.A. Carlson, J.D. Yeager, K. Fezzaa and D.E. Hooks (2014), Dynamic experiment using IMPULSE at the Advanced Photon Source, Journal of Physics: Conference Series, Volume 500, doi: 10.1088/1742-6596/500/4/042001

[12] Kastengren, A. and C.F. Powell (2014), Synchrotron X-ray techniques for fluid dynamics, Exp. Fluids, 55, doi:10.1007/s00348-014-1686-8.

[13] Willey, T.M., K. Champley, R. Hodgin, L. Lauderbach, M. Bagge-Hansen, C. May, N. Sanchez, B.J. Jensen, A. Iverson, and T. van Buuren (2016), X-ray imaging and 3D reconstruction of in-flight exploding foil initiator flyers, J. Appl. Phys., 119, doi:10.1063/1.4953681.

[14] Halls, B.R., C.D. Radke, B.J. Reuter, A.L. Kastengren, J.R. Gord and T.R. Meyer (2017), High-speed, two-dimensional synchrotron white-beam X-ray radiography of spray breakup and atomization, Opt. Express, 25(2), doi:10.1364/OE.25.001605.

[15] Heindel, T.J. (2019), X-ray Imaging Techniques to Quantify Spray Characteristics in the Near Field, At. Sprays, 28(11), doi:10.1615/AtomizSpr.2019028797.

[16] Hudspeth, M., J. Olles, J. Williams, A. Mandal, S. Root, T. Vogler (2019), Strength of porous α-SiO2 in a shock loaded environment, SAND2019-10277.

[17] Yan, J., Y. Lu, G. Chen, M. Yang and Z. Gu (2018), Advances in liquid metals for biomedical applications, Chem. Soc. Rev., 47, doi:10.1039/C7CS00309A.

[18] Wansleben, M., C. Zech, C. Streeck, J. Weser, C. Genzel, B. Beckhoff, and R. Mainz (2019), Photon flux determination of a liquid-metal jet X-ray source by means of photon scattering, J. Anal. At. Spectrom., 7, doi:10.1039/C9JA00127A.

[19] Feldkamp, L.A., L.C. Davis and J.W. Kress (1984), Practical cone-beam algorithm, J. Opt. Soc. Am. A, 1, doi:10.1364/JOSAA.1.000612.

[20] Kak, A.C. and M. Slaney (1988), Computerized Tomographic Imaging, IEEE Press, New York.

[21] Jimenez, E.S. and L.J. Orr (2013), Rethinking the union of computed tomography reconstruction and GPGPU computing, Proc. 8854, Penetrating Radiation Systems and Applications XIV; 88540A, doi:10.1117/12.2029995.

[22] Jimenez, E.S., L.J. Orr and K.R. Thompson (2012), An Irregular Approach to Large-Scale Computed Tomography on Multiple Graphics Processors Improves Voxel Processing Throughput, 2012 SC Companion: High Performance Computing, Networking Storage and Analysis, doi:10.1109/SC.Companion.2012.42.

[23] Beister, M., D. Kolditz, W.A. Kalender (2012), Iterative reconstruction methods in X-ray CT, Phys. Medica, 28(2), doi:10.1016/j.ejmp.2012.01.003.

[24] Champley, K. (2016), Livermore Tomography Tools (LTT) Technical Manual, LLNL-SM-687016.

[25] Hansen, P.C. and M. Saxild-Hansen (2011), A MATLAB package of algebraic iterative reconstruction methods, J. Comput. Appl. Math., 236, doi:10.1016/j.cam.2011.09.039.

[26] Gürsoy, D., F. De Carlo, X. Xiao and C. Jacobsen (2014), Tomopy: a framework for the analysis of synchrotron tomographic data, J. Synchrotron Radiat., 21(5), doi:10.1107/S1600577514013939.

[27] Scarano, F. (2013), Tomographic PIV: principles and practice, Meas. Sci. Tech., 24(1), doi:10.1088/0957-0233/24/1/012001.

[28] Schmidlin, P., M.E. Bellemann and G. Brix (1999), Subsets and overrelaxation in iterative image reconstruction, Phys. Med. Biol., 44(5), doi:10.1088/0031-9155/44/5/321.

[29] Arienti, M. and M. Sussman (2014), An embedded level set method for sharp-interface multiphase simulations of Diesel injectors, Int. J. Multiph. Flow, 59, doi:10.1016/j.ijmultiphaseflow.2013.10.005.

[30] Estellers, Virginia & Zosso, Dominique & Lai, Rongjie & Osher, Stanley & Thiran, Jean-Philippe & Bresson, Xavier (2012), An Efficient Algorithm for Level Set Method Preserving Distance Function, IEEE transactions on image processing: a publication of the IEEE Signal Processing Society, 21, doi:10.1109/TIP.2012.2202674.

[31] Demircan-Tureyen, E. and M.E.Kamasak (2017), A discretized tomographic image reconstruction based upon total variation regularization, Biomed. Signal Proces., 38, doi:10.1016/j.bspc.2017.03.015.

[32] Hanson, K.M. and G.W. Wecksung (1983), Bayesian approach to limited-angle reconstruction in computed tomography, J. Opt. Soc. Am., 73(11), doi:10.1364/JOSA.73.001501.

[33] Grauer, S.J., P.J. Hadwin and K.J. Daun (2016), Bayesian approach to the design of chemical species tomography experiments, Appl. Opt., 55(21), doi:10.1364/AO.55.005772.

[34] Daun, K.J., S.J. Grauer, and P.J. Hadwin (2016), Chemical species tomography of turbulent flows: discrete ill-posed and rank deficient problems and the use of prior information, J Quant. Spectrosc. RA 172, doi:10.1016/j.jqsrt.2015.09.011.

[35] Minerbo, G. (1979), Maximum entropy reconstruction from cone-beam projection data, Comput. Biol. Med., 9(1), doi:10.1016/0010-4825(79)90020-9.

[36] Gull, S.F. and T.J. Newton (1986), Maximum entropy tomography, Appl. Opt., 25(1), doi:10.1364/AO.25.000156.

[37] Smith, R.T., C.K. Zoltani, G.J. Klem and M.W. Coleman (1991), Reconstruction of tomographic images from sparse data sets by a new finite element maximum entropy approach, Appl. Opt., 30(5), doi:10.1364/AO.30.000573.

[38] R. R. La Foy and B. R. Halls, "Two-Color X-ray Imaging for Material Differentiation in Multiphase Flows," ILASS-Americas 33rd Annual Conference on Liquid Atomization and Spray Systems, May 14-17, 2023.

APPENDIX A. SYNCHROTRON MEASUREMENTS

A.1. Overview

Energetic events were imaged using white-beam radiography at Sector 7-BM of the Advanced Photon Source at Argonne National Laboratory. The images were collected via a Shimadzu HPV-X2 camera running at 5 MHz that was lens-coupled to a scintillator while the X-ray pulse repetition rate was 6.5 MHz. The mismatched repetition rates were accounted for through a multi-step method that modeled the rates of the beam and the camera and the scintillator decay time to return data with an effective rate of 6.5 MHz albeit with various levels of certainty for each frame. This scheme presents one method to collect images at MHz rates with the available camera technology.

A.2. Experimental Setup

The experiments were conducted at the Sector 7 bending magnet beamline at the Advanced Photon Source at Argonne National Laboratory. The imaging setup used the unfiltered white beam in 24-bunch mode with a copper chopper wheel to limit the flux in the hutch and the heating of the 100-μm thick GAGG scintillator that had a ~90 ns decay time. The 100-ps duration X-ray pulses froze the flow in time and the visible emission from the scintillator was collected by a Shimadzu HPV-X2 camera with a pair of coupled 50 mm f/1.2 Nikon lens resulting in 1:1 imaging. Therefore, the camera chip size and the field of view were both 250 × 400 px^2 or 8 × 12.9 mm^2, each pixel represented 32.3 μm^2. The camera framerate was 5 MHz with a fixed exposure of 110 ns. The energetic setup included a polycarbonate enclosure "boombox" with X-ray transparent windows rated to the net explosive weight of each device. The X-ray transparent windows minimized the attenuation of X-rays while providing a safe operating environment. Two energetic devices were used an igniter/initiator and a detonator. The igniter contained ~20 mg pentaerythritol tetranitrate (PETN) in a steel case. The purpose-built detonator (tailored for diagnostic development) used an RP-80 header, loaded with ~200 mg PETN, and a 304 stainless steel case with a length of 11.4 mm, diameter of 7.4 mm, and thickness of 1.1 mm. A schematic of the setup is shown in Fig. A-1 that includes the boombox with X-ray transparent windows enclosing the detonator and the X-ray beam imaging the detonator on to the scintillator and the scintillator emitting visible light collected by the camera via a turning mirror and objective.

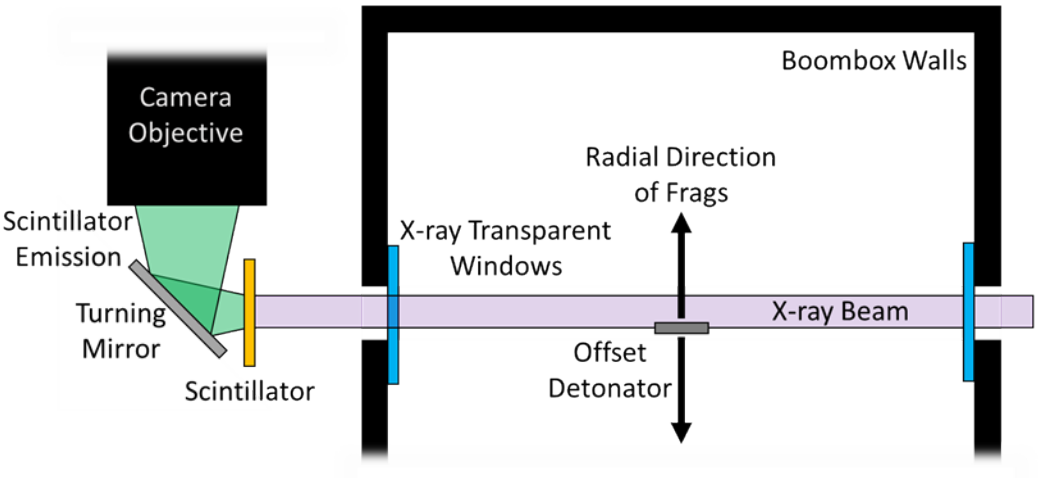

Figure A-1. The experimental setup including the boombox enclosure, the X-ray transparent windows, detonator, X-ray beam, scintillator, visible emission, turning mirror, and camera objective. Not drawn to scale.

The energetics were stored in a safe, locked location and brought out just prior to use. A low voltage firing set was used for the igniters and a high voltage firing system was used for the detonators. Both fire sets were interlocked to the hutch and to the boombox, the electrical systems were grounded, bled, and zero voltage verifications were performed to ensure there was no unintended energy discharge to the devices. Comprehensive safety documentation was in place for handling of the explosives and the firing system and was cleared with laboratory safety prior to traveling to the APS.

A.3. Image Processing

The focus of the work was to demonstrate an effective methodology to image high-speed energetic events even with rigid camera framerates that did not directly synchronize to the timing of the synchrotron. The camera was limited to specific frame rates of either 5 MHz with 110 ns exposure or 10 MHz with 50 ns exposure (and utilizing only half of the pixels, which is a function of the camera). The option to use 5 MHz frame rate allowed for better spatial resolution and lower noise (because all pixels were used). The 5 MHz frame rate couples to the synchrotron pulse rate of 6.5189 MHz in 24-bunch mode the same way running the camera at 10 MHz and imaging in 48-bunch mode. The result of the mismatched frame rates combined with the scintillator decay is the forming multiple X-ray images onto single camera exposure. This creates a pulsating background and temporal blur as seen in Figure A-2. Notable features of the images include the variation of the image intensity and the ghosting that can be readily seen at the leading edge of the detonator case and the smaller fragments at 5 μs past initial case motion.

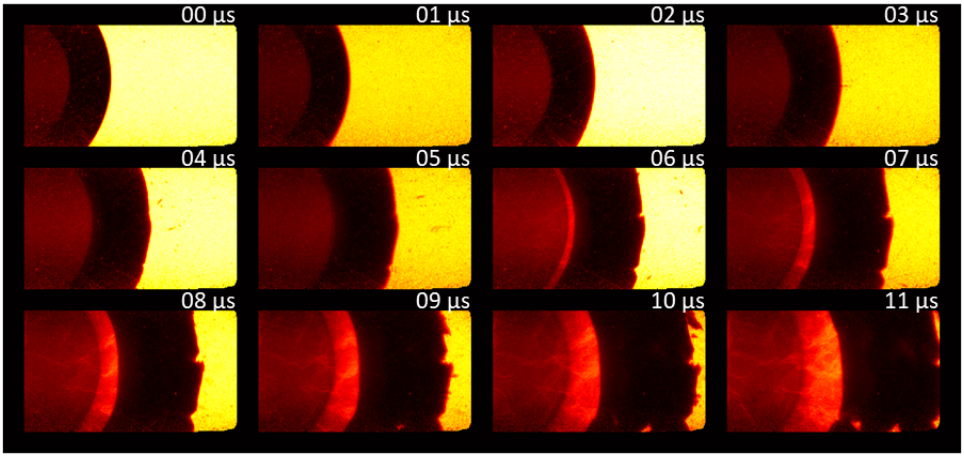

Figure A-2. Normalized images of the detonator case expansion starting from the static case.

A.4. Conclusions

Ultra-high-speed X-ray radiography was performed using a bending magnet beam line at the Advanced Photon Source to image energetic events. Due to the multiple X-ray images on each camera exposure the imaging system needed to be modeled to understand how the light was emitted and collected from the scintillator by the camera and how to temporally deconvolve the images.

APPENDIX B. X-RAY SOURCE MODEL

B.1. Overview

The results of the PIC simulations found that the tube operated well under the space-charge limit under normal maxim conditions by as much as two orders of magnitude. One caveat is that the actual geometry and thermal state of the device, especially the cathode emission area and temperature, may greatly impact that result, but achieving an increase in tube current by the factor of a few does not appear the be the limiting condition.

Modeling the radiation output such that X-ray characteristics could be compared with experimental measurements required a three-dimensional electron/photon radiation transport model to compute the electron to X-ray conversion via bremsstrahlung [1] and transport X-rays to measurement locations. The model output was compared with experimental dose measurements with very good agreement in trends in data as voltages and currents were varied, but poor agreement between measured and computed dose. This disagreement was likely due to misunderstandings between experimentalists and modelers and could be resolved with additional effort.

To determine the limitations due to anode heating, one dimensional radiation transport models were used to calculate the energy deposition by the electron beam into the anode. That deposition profile can then be used as an energy source term in a thermal diffusion, heat equation, solution.

B.2. 3D Model

The first objective of the study was to determine whether the X-ray tube, that consists of a high voltage, vacuum anode-cathode gap, or diode, where electrons are emitted from a thermionic cathode and accelerated through the applied voltage to the anode, could support an increased electron current. The fundamental physics limiting the maximum transmittable electron current density in a vacuum diode is due to the self-charge of the electrons that shields out the external electric field and thus preventing electron flow from cathode to anode, this is called the space-charge limit (SCL). [2, 3] In systems such as the electron tube modeled here Fig. A-3, the maximum admissible current is best determined using a three-dimensional, self-consistent particle-in-cell (PIC) code to simulate the electron flow. The Empire code [4] was used to determine that the X-ray tube could sustain approximately two orders of magnitude higher currents (~50 amps) than the normal operating maximum current (0.5 amps).

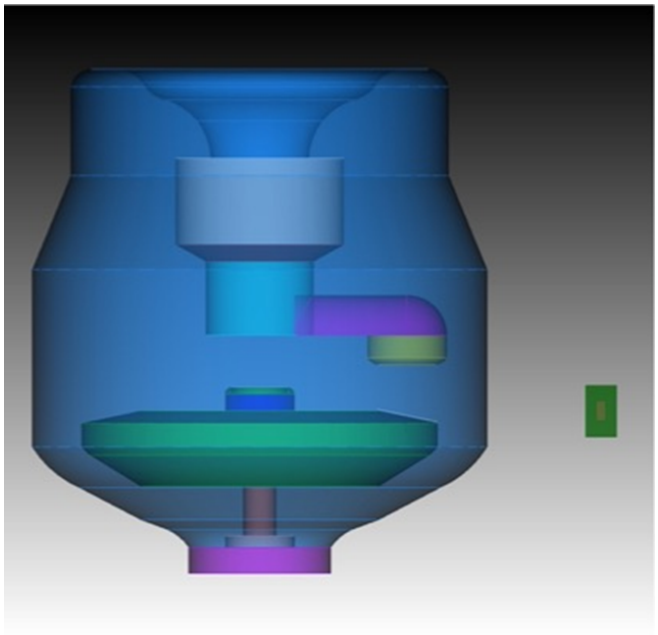

Figure A-3. As modeled geometry of the X-ray tube. The location of the TLD, dose detector is shown green outside of the Tube.

In the electro-magnetic (EM) PIC simulation, an EM wave was coupled into the X-ray tube inducing a voltage drop between the cathode and anode with the magnitude of the voltage drop and the voltage rise time depending on the coupled waveform. The diode current was injected at the cathode using either a beam injection algorithm where the beam current and thermal spread were specified, or an algorithm that injected at the SCL. Figure A-4 shows the voltage and current characteristics of the diode with either a 100 kV applied voltage, or a 150 kV applied voltage as seen in the thinner blue and magenta lines rising from zero to 1 and 1.5 respectively. The corresponding diode currents are seen as thicker lines in the figure that rise from zero at 1ns and reach steady values indicating full transmission of the injected currents. From the figure, it is easily seen that even a 500mA current injected into the 100 kV diode is fully transmitted, and, in fact, for a 150 kV diode voltage, the space charge limit is not reached until around 50 amperes Fig. A-5. As seen in Fig. A-5, the contour of electron density in the left-hand figure shows little spread due to space charge effects, whereas the beam density contour in the right-hand figure does indicate spreading of the beam at the anode when the SCL current is reached. This observation allows for the electron beam to be treated as a mono-energetic beam for the purpose of calculating an expected measured dose from diodes well below the SCL since the beam distributions are nearly monoenergetic upon impact with the anode.

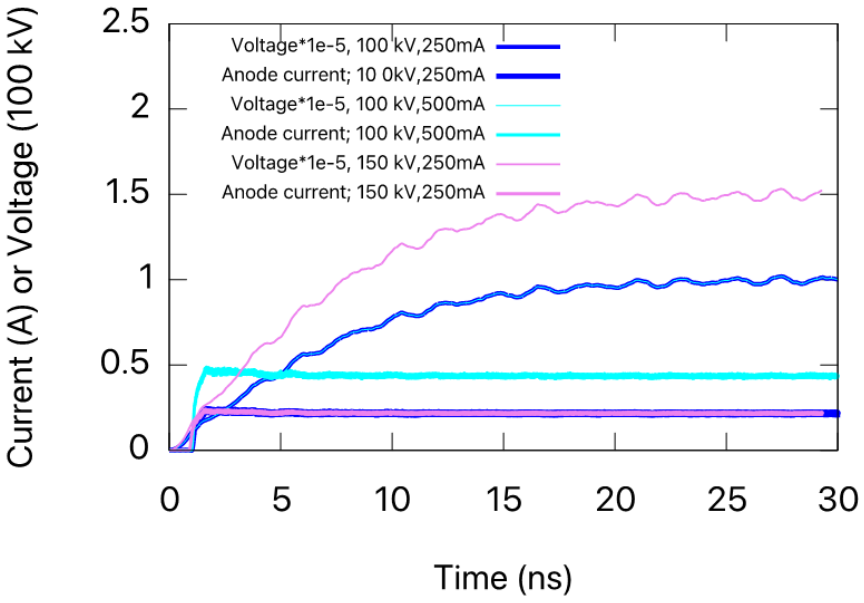

Figure A-4. PIC simulation indicates that tube operates well under the space charge limit (SCL) – Current and Voltage can be specified independently over typical operation range. The diode voltages, depicted as the thin lines rising from zero to 1 and 1.5 in units of 100 kV, reach the desired steady operation values near 30 ns in this model. The diode currents, in units of amperes, are depicted as the thicker lines that rise quickly from zero around 1 ns and reach a steady value.

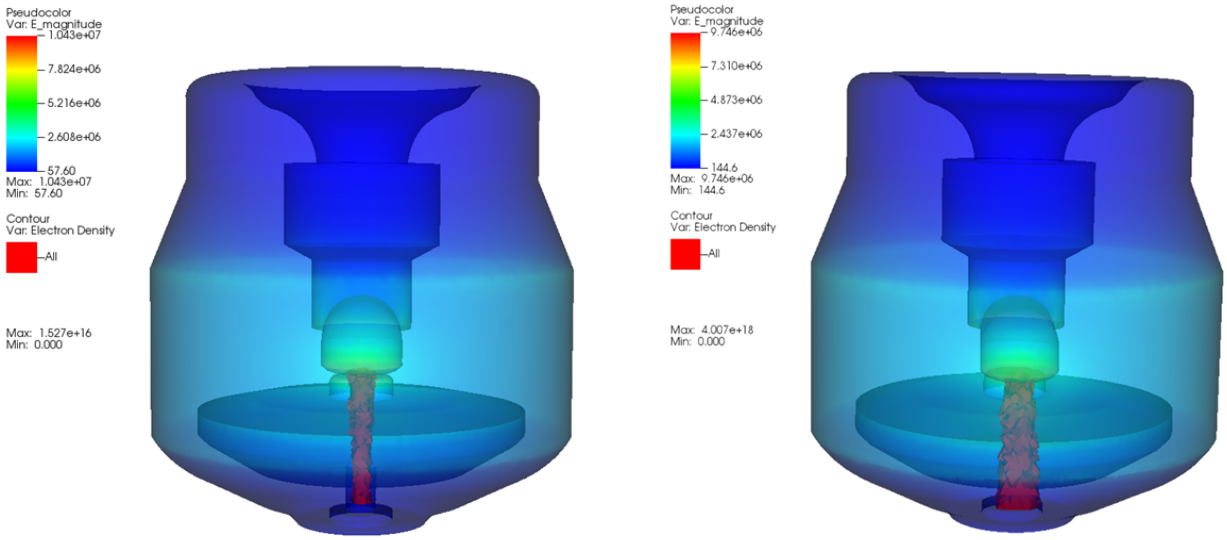

Figure A-5. PIC simulations of X-ray diodes wth a 150 kV voltage and an injection current of 0.5 amps on the left, and at the SCL current of approximately 50 amps on the right.

The second objectives were to determine anode heating and X-ray output. (Not enough information was available regarding the cathode structure to develop a model or make a valid determination regarding its ability to produce the desired current.) Electron-photon transport models of the X-ray device were developed to characterize X-ray production and output and energy deposition into the

anode which controls anode heating, with the first utilizing a three-dimensional model and the second using one-dimensional radiation transport models.

A three-dimensional Monte-Carlo (MC) model was used for characterization of the X-rays created via bremsstrahlung conversion [1] where high energy electrons slowdown in high atomic number materials producing X-rays. This model utilized the Integrated Tiger Series (ITS) code Accept [5] to model X-ray production given either a mono-energetic electron beam or an electron beam with a user-specified energy distribution. To be consistent with the PIC model, a particle boundary diagnostic was created in Empire that collected the spatial, energy and momentum distribution of each PIC macroparticle that stuck the bremsstrahlung converter Fig. A-6, and that information was used to create an electron distribution for use in the MC model. However, such an endeavor was quite time consuming and yielded only a slight variation on X-ray production compared to using a monoenergetic beam if the beam was well below the SCL. Thus, the MC model was used as a stand-alone tool for studying the effects of diode voltage on X-ray characteristics.

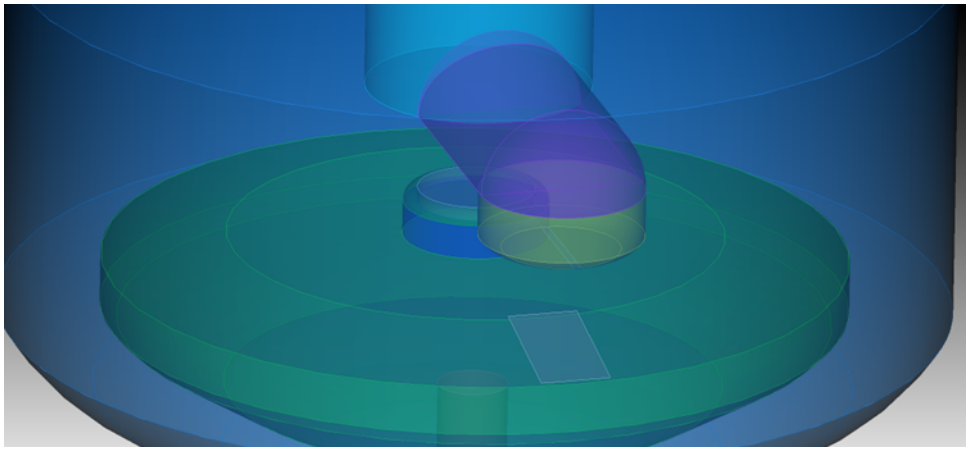

Figure A-6. The region where PIC electrons are tallied for use as input into the MC transport model is shown as a light blue outline on the green anode.

The PIC to MC model may be of interest for diodes with very high current where the electron behavior is less like an ideal mono-energetic beam. The modeling process is as follows:

1. Geometry is created (from tube drawings/specifications) in both CUBIT (for PIC simulation) and in ITS using a combinatorial-geometry scheme.
2. EM particle simulations are performed generating diode electron beam.
3. Electron information is collected in time at the anode converter (e.g. momentum, PIC weight, position).
4. The collected electron data is analyzed to produce a cumulative distribution function (cdf) in energy.
5. The electron cdf is used as a source for MC radiation production and transport.
6. Output dose (energy absorbed in modeled TLD) is the final quantity of interest.

Early experimental thermoluminescent dosimeter (TLD) measurements of the X-ray dose output from an actual device were compared with simulated TLD dose, which showed very good agreement in dose versus voltage trend Fig. A-7 but differed substantially in magnitude. Better communication between experimentalist and modelers may alleviate this issue.

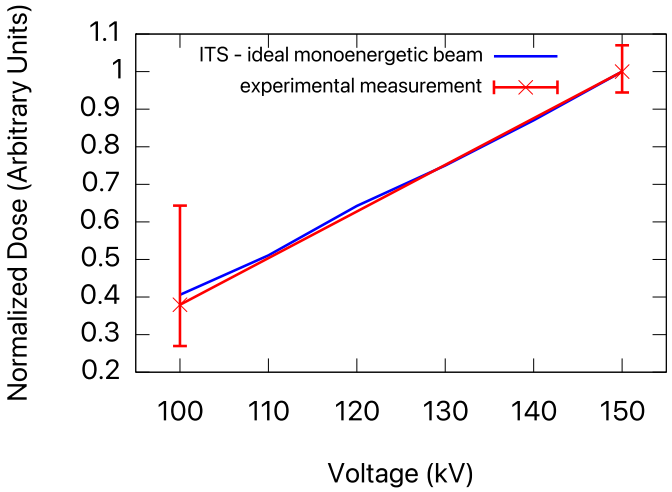

Figure A-7. The PIC and experimental data normalized to the maxima in each dataset plotted versus diode voltage. The data trend is in good agreement between measured and modeled data; however, the actual doses did differ substantially, possibly due to calibration or normalization differences.

B.3. 2D Time-Resolved Heat Transfer Model

To model heating of the tube anode, one dimensional radiation transport models were implemented to determine the energy deposition profile into the anode. One-dimensional models are used because the energy deposition requires high resolution subzoning of the anode as most of the energy is deposited within microns of the surface. A typical deposition profile is shown in Fig. A-8. As seen in the figure, most of the energy that would go into heating of the anode is deposited within 20 microns. The transport models implemented were very simple and included an electron beam of specified energy (usually mono-energetic, but a distribution is easily used) and a stack-up of the anode materials. These calculations were done both using MC methodology with the ITS Tiger code and using a one-dimensional deterministic radiation code developed at Sandia called RAPTURE [6]. The codes were in good agreement as long as the anode was sufficiently resolved. Once the energy deposition profile is calculated, it can be used as an energy source term in a thermal diffusion, heat equation, solve for the anode temperature.

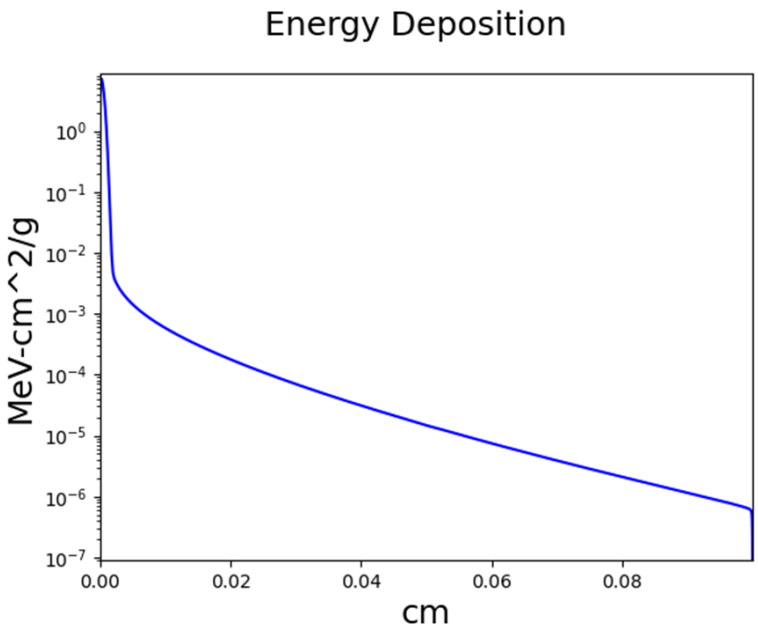

Figure A-8. Shown here is a typical energy deposition profile for a 150 kV electron beam into the anode material.

The energy deposition was then utilized in a 2D heat transfer model developed in MATLAB. The model distributed the energy deposition over a 1D slice of the 2D beam focal spot that translated in space over time to represent the rotation of the anode. The anode angle was 12 degrees, and the focal spot was 1 mm by 0.48 mm Gaussian beam resulting in a projected beam of 1 mm^2. The slice was taken from the center of the beam to avoid edge effects along the radial axis. The beam exposure was 200 μs, sufficient to achieve steady state. The anode was modeled as pure tungsten and 0.5 mm in depth to reduce computational expense while allowing the heat to dissipate to the background temperature of 300 K. The thermal properties of tungsten properties were determined from NIST databases [6, 7]. The source current was 500 mA and the potential was 150 kV. The anode diameter, at the location of the focal spot was 85 mm, and the anode rotation frequency was 167 Hz. The element size was 10 × 10 μm^2 and the time step was 0.2 μs. Radiation heat transfer from the anode to the surroundings was investigated but determined to be negligible. Figure A-9 shows the output from the model at 2, 20, and 200 μs after the initial impingement of electrons. The model reaches steady state around 100 μs, and the temperature rise at the lower edge of the domain rises ~3 K.

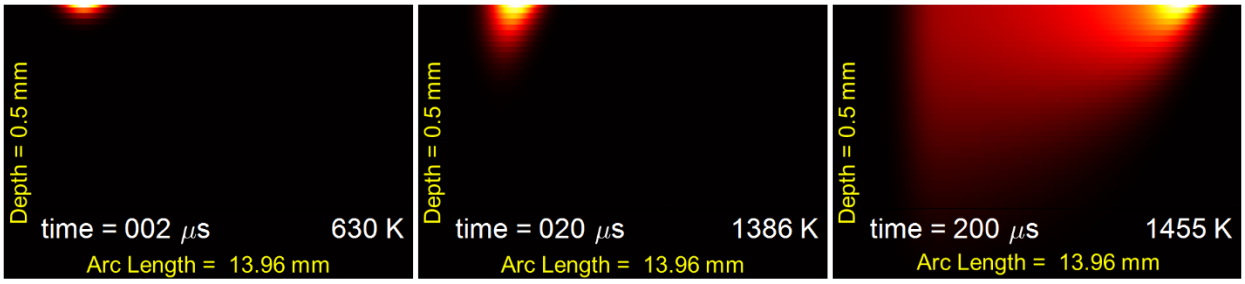

Figure A-9. Temperature map of the cross section of the anode showing the relative temperature and the max temperature of the anode at three timesteps 2 μs, 20 μs, and 200 μs.

B.4. References

[1] Emigh, C. R., "Thick Target Bremsstrahlung Theory", Technical Report, LA-4097-MS UC-34, Los Alamos Scientific Laboratory, 1970.]

[2] Child, C. D. (1911). Discharge from hot CaO. Physical Review (Series I), 32(5), 492.

[3] Langmuir, I. (1913). The effect of space charge and residual gases on thermionic currents in high vacuum. Physical Review, 2(6), 450.

[4] M Bettencourt, M. T., Brown, D. A., Cartwright, K. L., Cyr, E. C., Glusa, C. A., Lin, P. T., ... & Jarvis, S. A. (2021). EMPIRE-PIC: a performance portable unstructured particle-in-cell code. Communications in Computational Physics, 30(SAND-2021-2806J).

[5] B. C. Franke, R. P. Kenesek, T. W. LAUB, AND M. J. CRAWFORD, ITS Version 6: The Integrated TIGER Series of Coupled Electron/Photon Monte Carlo Transport Codes, Revision 5, Tech. Rep. SAND2008-3331, Sandia National Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550, 2013.

[6] D. Bruss, "RAPTURE: A 1-D Radiation Effects Analysis Tool", Technical Report, SAND2019-11323C, Sandia National Laboratories, 2019.

[7] Powell, R. W., Cho Yen Ho, and Peter Edward Liley, "Thermal conductivity of selected materials. Vol. 8. Washington, DC: US Department of Commerce," National Bureau of Standards, 1966.

[8] Chase, M. W. (1996). NIST–JANAF thermochemical tables for the bromine oxides. Journal of Physical and Chemical Reference Data, 25(4), 1069-1111.

APPENDIX C. NEURAL NETWORK RECONSTRUCTION ALGORITHM

Ryan J. Sirimanne, Rui Tang, Joseph P. Molnar, and Samuel J. Grauer

Department of Mechanical Engineering, Pennsylvania State University

C.1. Overview

Our team has developed a comprehensive software package for X-ray tomography that can be applied to both dynamic and static objects. This software provides a user-friendly framework for processing 2D X-ray radiographs of 3D and 4D scenes captured from multiple imaging devices, producing accurate reconstructions of their internal fields. The package includes a variety of reconstruction algorithms, such as analytical, algebraic, and neural network-based methods, each tailored for different types of data and application needs.

A key feature of our software is its ability to leverage advanced forms of prior information—such as known object features, similarities to other objects, or a detailed design specification—within both the neural and algebraic reconstruction methods. This approach enhances reconstruction accuracy, especially in scenarios where the measurements are spatially sparse. The core of these reconstruction techniques is a streamlined version of our comprehensive, physics-based signal model that describes the X-ray measurement process. This model can accommodate a range of ray geometries and can be discretized using voxels or wavelets or represented in continuous form using one or more coordinate neural networks. To validate the measurement models and reconstruction algorithms, our software includes a toolbox that can generate exact projections for a suite of geometric primitives. This suite offers an array of basic shapes, tools for defining their volumes and arrangement, and determines the resultant projections. Additionally, the software supports user-provided models in the form of computer-aided design (CAD) files, enabling greater flexibility for testing and validation with a representative geometry and material selection.

This document provides an overview of the theory that undergirds X-ray tomography, reconstruction algorithms, and our software's functionality. The report includes demonstrations on both synthetic and experimental data. We also discuss plans for future development and enhancements to the package.

C.2. X-ray Radiography Measurement Model

When X-rays pass through an object, a fraction of the energy is absorbed or scattered by the material, which leads to attenuation of the beam. The net effect of absorption and scattering is termed extinction, and the cumulative effect of extinction along an X-ray beam is a decrease in its intensity as the beam reaches a detector. The degree of attenuation depends on the beam geometry, path length, energy spectrum of the X-rays, and material properties of the target object, including its density and chemical composition. The effectiveness of a material in attenuating or "blocking" the X-ray beam is

quantified by its spectral extinction coefficient field, $\mu_E \propto \rho$. Note that ρ is the material density and E is the energy of X-rays:

$$E = \frac{hc}{\lambda} = hc\nu \tag{2.1}$$

where h is Planck's constant, c is the speed of light, and λ and ν are the wavelength and wavenumber of an X-ray, respectively. Hence, quantities with the subscript E, $(\cdot)_E$, are spectral quantities.

To determine the unknown extinction coefficient field, and potentially deduce the material density, multiple measurements of attenuated X-ray beams from different perspectives are reconstructed by inverting a measurement model, a.k.a. forward model. This model converts a 3D extinction coefficient field into 2D projections, termed radiographs, which are formed from a pair of reference and attenuated images recorded by a detector. Reconstruction is the inverse problem whereby a set of radiographs is used to deduce the 3D field by inverting the forward model. The accuracy of reconstructions critically depends on the fidelity of the forward X-ray model; the model must account or all relevant physical effects occurring during measurement, such as scattering, absorption, spectral beam hardening, and the detector response, to ensure reliable results.

In this section, we outline the key components of our comprehensive, physics-based signal model and the assumptions that have been made to derive its simplified forms. Various simplified models have been implemented in our suite of reconstruction algorithms.

C.2.1. Physics-Based Signal Model

An X-ray radiograph is formed from a pair of images: a reference image and an attenuated image, both recorded by the same detector. The signal measured at the ith pixel of the detector is denoted S_i, with a special designation for the reference signal, $S_{0,i}$, which is recorded in the absence of the target object, i.e., when the extinction coefficient is zero: $\mu_E = 0$. The signal at each pixel depends on the spectral intensity of the X-ray beam incident on the sensor, I_E, which is attenuated according to the extinction coefficient along the ray path,

$$\frac{\mathrm{d}I_E}{\mathrm{d}s} = -\mu_E I_E \tag{2.2}$$

where s is a progress variable along an infinitesimal ray within the X-ray beam. This attenuation relationship is encapsulated by the Beer–Lambert Law,

$$I_E = I_{0,E}\exp\left(-\int_{\mathrm{ray}}\mu_E \mathrm{d}s\right) \tag{2.3}$$

where $I_{0,\mathrm{E}}$ represents the reference intensity, and the integral is evaluated along the path of the ray. The form of Eq. (2.3) motivates the calculation of a radiograph from the reference and attenuated signals. At each pixel, we define a projection of the target object,

$$p_i \equiv \log\left(\frac{S_{0,i}}{S_i}\right) = \int_{\mathrm{ray}}\mu_E \mathrm{d}s \tag{2.4}$$

A set of projections is collected into an $m \times 1$ vector, $p = [p_1,...,p_m]^\top$, and these projections can be utilized to infer the 3D distribution of μ_E. However, the model presented in Eq. (2.4) relies on several

key assumptions that may not always hold true in practical imaging scenarios. The remainder of this section introduces a comprehensive signal model and then outlines two key simplifications. The simplified model is designed to be compatible with mainstream tomographic inversion techniques.

The signal for the ith pixel can be modeled as follows:

$$S_i = \int_0^\infty \int_{\mathcal{X}_\mathrm{i}} \eta_E I_{0,\mathrm{E}} \exp\Big(- \int_{\mathcal{R}(\mathbf{u})} \mu_E(s)\mathrm{d}s \Big)\, \mathrm{d}\mathbf{u}\, \mathrm{d}E \tag{2.5}$$

Here, $\mathcal{X}_\mathrm{i}$ is the region of the sensor corresponding to the ith pixel, $\mathbf{u} = [u,v]^\mathsf{T}$ is a vector of sensor coordinates in $\mathcal{X}_\mathrm{i}$, η_E is the detector's spectral quantum efficiency and gain, $I_{0,\mathrm{E}}$ is the spectral source term, $\mathcal{R}$ is the line of sight from the source to the sensor position $\mathbf{u}$, s is a progress variable along that line, and μ_E is the local spectral extinction field at the position x corresponding to $s \in \mathcal{R}$. Formally, this mapping is represented by

$$\mu_E(s) \equiv \mu_E\big[\mathcal{R}^{-1}(s,\mathbf{u})\big] \tag{2.6}$$

where, $\mathcal{R}^{-1}{:}(s,\mathbf{u}) \mapsto \mathbf{x}$ returns a 3D location along the line of sight from the source to the sensor position $\mathbf{u}$, and the mapping is defined such that

$$\left|\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}s}\right| = 1 \tag{2.7}$$

Equation (2.5) does not account for multiple scattering events, which can be incorporated as needed.

While the Beer–Lambert Law is included in Eq. (2.5), the presence of nonlinearities arising from spectral dependencies can cause the approximation in Eq. (2.4) to break down. To explicate the relationship between these expressions, we briefly describe how to recover Eq. (2.4) from Eq. (2.5). The first simplification involves assuming constant spectral properties, such that $\eta_E \approx \eta$, $I_{0,E} \approx I_0$, and $\mu_E \approx \mu$, resulting in a pseudo-monochromatic problem. This approximation can be justified if either the source spectrum or the detector-response spectrum is sharply peaked, minimizing variations across different energy levels. For the next simplification, we assume that the integrated extinction coefficient remains approximately constant across a pixel, leading to the following relationships:

$$S_i \approx \eta I_0 \Delta E |\mathcal{X}_i| \exp\Big(- \frac{1}{|\mathcal{X}_i|} \int_{\mathcal{X}_\mathrm{i}} \int_{\mathcal{R}(\mathbf{u})} \mu_E(s)\mathrm{d}s\, \mathrm{d}\mathbf{u} \Big) \tag{2.8}$$

and

$$p_i \approx \frac{1}{|\mathcal{X}_i|} \int_{\mathcal{V}_\mathrm{i}} \mu_E(\mathbf{x})\mathrm{d}\mathbf{x} \tag{2.9a}$$

defined such that

$$\lim_{|\mathcal{X}_i| \to 0} \frac{1}{|\mathcal{X}_i|} \int_{\mathcal{V}_\mathrm{i}} \mu_E(\mathbf{x})\mathrm{d}\mathbf{x} = \int_{\mathcal{R}(\mathbf{u}_i)} \mu_E(s)\mathrm{d}s \tag{2.9b}$$

where $\mathcal{V}_\mathrm{i}$ represents the volume of the ith X-ray beam, defined as the volume swept by $\mathcal{R}$ for $\mathbf{u} \in \mathcal{X}_i$, and $\mathbf{u}_i$ is the centroid of the ith pixel. Hence, under the assumption of an infinitesimal ray geometry,

Eq. (2.8) converges to Eq. (2.4). However, in certain scenarios—like when the detector has large pixels or is placed close to the target object—it becomes necessary to account for the full 3D beam geometry to better approximate the signal formation process and capture effects like image blur.

C.2.1.1 Ray Geometry Models

The choice of ray model can significantly influence both the accuracy and computational cost of reconstruction algorithms in computed tomography. Various geometric models are commonly employed to approximate the paths of X-ray beams through an object, each offering distinct advantages and trade-offs.

The most basic model is the infinitesimally thin ray, which approximates the X-ray beam as a line of negligible width passing through the volume. This model is efficient and widely used in analytical algorithms such as filtered back-projection (FBP) due to its simplicity. However, its lack of consideration for the beam's finite width can lead to reconstruction artifacts and inaccuracies, especially when reconstructing fine details or in the presence of image blur. An improvement over the thin-ray model is the cylinder ray model, where the X-ray beam is represented as a cylinder with a constant cross-sectional area in the probe volume. This model better approximates the finite width of real X-ray beams and can more accurately capture blur effects caused by beam divergence. However, the computational cost of the cylinder ray model is comparable to that of more advanced models, limiting its efficiency advantage.

The cone-ray model provides a further refinement by allowing the cross-sectional area of the X-ray beam to evolve along the line of sight, accommodating variations in blur and divergence throughout the field of view. This model captures changes in beam spread more accurately and provides a more realistic approximation than the cylinder ray model. However, this increased accuracy requires computational resources similar to the cylinder ray geometry, making it a balanced choice between simplicity and precision. In most cases, we adopt a cone-beam model.

For the highest fidelity, the full ray geometry model can be utilized, which is typically implemented through a Monte Carlo method. In this approach, the volume is densely sampled, and each point within the volume is projected onto the detector using a camera transform that can account for non-linear effects, such as lens distortions when imaging an X-ray scintillator. This method converges to the true underlying geometry as the number of samples increases, providing the most accurate representation of signal formation. However, this method poses a high computational cost, making it

suitable for cases where maximum accuracy is essential and computational resources are not a constraint. The specific sampling strategy and its implications are discussed in the next two sections.

C.2.2. 2.2 Discrete Fields

Most reconstruction algorithms require a parameterized extinction field to compute $\mathbf{p}$ as a function of μ. It is convenient to adopt a finite, orthonormal basis for this purpose, denoted $\phi = [\varphi_1,...,\varphi_m]$. using this basis, the continuous extinction coefficient field can be approximated as

$$\mu(\mathbf{x}) \approx \sum_{j=1}^{n} \mu_j \varphi_j(\mathbf{x}) \tag{2.10}$$

where μ_j is the coefficient corresponding to the jth basis function, φ_j. Discrete extinction coefficient fields are then represented by a vector of coefficients, $\mu = [\mu_1,...,\mu_m]$. The exact, or ideal, coefficient for each basis function is given by the inner product of the true field and the basis function,

$$\mu_{\mathrm{ex},j} = \int_{\Omega} \mu_{\mathrm{ex}}(\mathbf{x}) \varphi_j(\mathbf{x}) \mathrm{d}\mathbf{x} \tag{2.11}$$

where Ω is the probe volume and $(\cdot)_{\mathrm{ex}}$ denotes an exact quantity. Since Eq. (2.8) is linear, substituting Eq. (2.9a) into the model yields a linear tomography model,

$$\mathbf{p} = \mathbf{A}\boldsymbol{\mu} \tag{2.12}$$

where the elements of the matrix A are defined as

$$A_{i,j} = \int_{\mathcal{V}_i} \varphi_j(\mathbf{x}) \mathrm{d}\mathbf{x} \tag{2.13}$$

Algebraic algorithms for tomographic reconstruction seek the discrete field μ that best satisfies Eq. (2.10) as well as any additional equations that are used to regularize the inversion.

The enclosed package includes both voxel and wavelet bases for parameterizing the extinction coefficient field. The voxel basis is straightforward. Voxels are defined as follows:

$$\varphi_j(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in \mathcal{V}_j \\ 0, & \text{otherwise} \end{cases} \tag{2.14}$$

In this expression, $\mathcal{V}_j$ represents the cubic volume corresponding to the jth voxel. Consequently, the matrix element $A_{i,j}$ is given by the intersection of the ith ray volume and the jth voxel volume, i.e., $\mathcal{V}_i \cap \mathcal{V}_j$. For the thin-ray model, this intersection volume can be computed exactly using an off-the-

shelf ray–box collision algorithm. However, for cylinder and cone rays, one of two standard approximation methods is typically employed.

The first method involves drawing random samples from $\mathcal{V}_j$ and determining the fraction of those samples that fall within $\mathcal{V}_i$. This fraction is then multiplied by the voxel volume to estimate $A_{i,j}$. Formally:

$$A_{i,j} \equiv \mathcal{V}_i \cap \mathcal{V}_j \approx \frac{|\mathcal{V}_j|}{n_{\text{smp}}} \sum_{k=1}^{n_{\text{smp}}} \mathbb{1}_{\mathcal{V}_i}(\mathbf{x}_k) \tag{2.15}$$

where $\mathbb{1}_{\mathcal{V}_i}$ is an indicator function,

$$\mathbb{1}_{\mathcal{V}_i}(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in \mathcal{V}_i \\ 0, & \text{otherwise} \end{cases} \tag{2.16}$$

and the random samples, $\mathbf{x}_k$, are uniformly drawn from $\mathcal{V}_j$. This method approaches the true intersection volume as n_{smp} grows large, and it is fully compatible with any beam geometry and orthonormal basis.

The second method is a simpler but cruder approach that depends on the beam model and basis. Structured sampling is used within $\mathcal{V}_j$; for example, dividing each voxel into a grid of $10 \times 10 \times 10$ subvoxels and checking for intersections with Vi at the subvoxel centroids. While this method is faster than the sampling approach, it is less accurate. Since the matrix A only needs to be populated once for a given system configuration, it is generally preferable to use the first method with a large number of samples to achieve higher accuracy.

C.2.2.1 2.2.1 Wavelet Basis

Wavelets are localized oscillatory functions that can efficiently represent a wide range of signals. Many signals are sparse in the wavelet domain, making wavelets an effective tool for data compression and a strong foundation for tomographic imaging. Our codebase includes a wavelet analysis module that facilitates reconstruction with regularization directly in the wavelet domain. This section provides a brief overview of the wavelet transform as applied to X-ray tomography.

To understand the wavelet transform, consider a 1D signal, f. We discretize this to form a $n \times 1$ vector, $\mathbf{f} = [f_1,...,f_m]$.. The first level of the discrete wavelet transform decomposes f into a set of approximation coefficients, $\mathbf{a}^1 \in R^{n/2}$, and detail coefficients, $\mathbf{d}^1 \in R^{n/2}$. This can be expressed as $\mathcal{D}^j$ $:\mathbf{f} \mapsto (\mathbf{a}^1, \mathbf{d}^1)$. The approximation coefficients in $\mathbf{a}^1$ represent the low-frequency components (coarse features) of the signal, while the detail coefficients in $\mathbf{d}^1$ capture the high-frequency components (fine details). This decomposition can be applied recursively to the approximation coefficients, $\mathbf{a}^1$, resulting

in multiple levels of transformation that progressively break down the signal into coarser approximations and finer details.

At each level of the wavelet transform, a new set of approximation and detail coefficients is generated, further decomposing the signal into progressively coarser approximations and finer details:

$$\mathcal{D}^2 : \mathbf{a}^1 \mapsto (\mathbf{a}^2, \mathbf{d}^2) \tag{2.17a}$$

$$\mathcal{D}^j : \mathbf{a}^{j-1} \mapsto (\mathbf{a}^j, \mathbf{d}^j) \tag{2.17b}$$

$$\mathcal{D}^j \circ \cdots \circ \mathcal{D}^2 : \mathbf{f} \mapsto (\mathbf{a}^j, \mathbf{d}^j, \ldots, \mathbf{d}^1) \tag{2.17c}$$

where the jth-level transform, $\mathcal{D}^j$, is applied to the approximation coefficients from the $(j-1)$th-level, $\mathbf{a}^{j-1}$. The discrete wavelet transform can also be expressed in matrix form. At the first level,

$$\mathbf{H}^1 \mathbf{f} = \mathbf{a}^1 \tag{2.18a}$$

$$\mathbf{G}^1 \mathbf{f} = \mathbf{d}^1 \tag{2.18b}$$

where $\mathbf{H}^1 \in R^{n \times n/2}$ and $\mathbf{G}^1 \in R^{n \times n/2}$ are the scaling and wavelet matrices, respectively, which together form an orthonormal basis for the discrete wavelet transform,

$$[\mathbf{H}^1; \mathbf{G}^1]^\top [\mathbf{H}^1; \mathbf{G}^1] = \mathbf{I} \tag{2.19}$$

In general, $\mathbf{H}^j$ and $\mathbf{G}^j$ are matrices of size $n/2^{j-1} \times n/2^j$, corresponding to the j-th level of the wavelet transform. With these matrices, the original signal can be exactly reconstructed from its approximation and detail coefficients at each level:

$$\mathbf{f} = \mathbf{H}^{1\top} \mathbf{a}^1 + \mathbf{G}^{1\top} \mathbf{d}^1 \tag{2.20a}$$

$$\mathbf{f} = \mathbf{H}^{1\top} \left(\mathbf{H}^{2\top} \left(\mathbf{H}^{3\top} (\ldots) + \mathbf{G}^{3\top} \mathbf{d}^3 \right) + \mathbf{G}^{2\top} \mathbf{d}^2 \right) + \mathbf{G}^{1\top} \mathbf{d}^1 \tag{2.20b}$$

and so forth. The scaling and wavelet matrices are composed of 1D filters of length m, denoted as $h = [h_1, \ldots, h_m]$ for the scaling filter and $g = [g_1, \ldots, g_m]$ for the wavelet filter. The rows of $\mathbf{H}$ and $\mathbf{G}$ are populated by placing the filters h and g with zero-padding on the left and right sides of the filters. Each successive row shifts the filters to the right by two positions, effectively implementing a down sampling operation. The filters generally "wrap around" the edge of the matrix, following a periodic boundary condition, although other boundary conditions (such as zero-padding or symmetric extension) can also be used.

We use Daubechies wavelets to provide a compressed representation of the target field. The filters for this transform must satisfy several constraints, including $\sum h_i^2 = \sum g_i^2 = 1$, $\sum h_i = \sqrt{2}$, $\sum g_i = 0$. Additionally, the relationship between the scaling and wavelet filters is given by $g_i = (-1)^{i+1} h_{m-i+1}$ which allows Daubechies wavelets to be tabulated in terms of h, alone. There are multiple

orders of Daubechies wavelets, and we chose the commonly used fourth-order variant. For this version, the elements of the scaling filter are:

$$h_1 = \frac{1+\sqrt{3}}{4\sqrt{2}}, \quad h_2 = \frac{3+\sqrt{3}}{4\sqrt{2}}, \quad h_3 = \frac{3-\sqrt{3}}{4\sqrt{2}}, \quad \text{and.} \quad h_4 = \frac{1-\sqrt{3}}{4\sqrt{2}} \tag{2.21}$$

Elements of the wavelet filter are specified accordingly. Notably, the scaling and wavelet filters remain the same across all levels of the transform.

So far, we have not explained how a wavelet basis facilitates compression. In practice, compression is achieved by retaining only the approximation coefficients while discarding the detail coefficients, or by simply not reconstructing them. As a result, the jth-level wavelet approximation of a signal is

$$\mathbf{f} \approx \mathbf{H}^{1\top} \cdots \mathbf{H}^{j\top} \mathbf{a}^j \tag{2.22}$$

Furthermore, the forward and inverse wavelet transforms can be represented in block form. To reconstruct the (potentially compressed) signal, $[\mathbf{a}^j;\mathbf{d}^j;...\mathbf{d}^1]$, we first stack the jth-level scaling and wavelet matrices into a single matrix, $S^j = [H^j;G^j]$, and then arrange them in block form:

$$\mathbf{B}^j = \begin{bmatrix} \mathbf{S}^j & \mathbf{0} \\ \mathbf{I} & \mathbf{I} \end{bmatrix} \tag{2.23}$$

where $\mathbf{I}$ and $\mathbf{0}$ are $n/2^{j-1} \times n/2^{j-1}$ identity and zeros matrices, respectively. Note that at the first level, $B^1 = S^1$. The jth-level discrete wavelet transform operator may be expressed as:

$$\mathbf{T}^j = \mathbf{B}^j \cdots \mathbf{B}^1 \tag{2.24a}$$

such that

$$\mathbf{T}^j \mathbf{f} = [\mathbf{a}^j;\mathbf{d}^j;...\mathbf{d}^1] \tag{2.24b}$$

The matrix $\mathbf{T}^j$ is invertible, so the inverse transform is simply

$$\mathbf{f} = \mathbf{T}^{j-1}[\mathbf{a}^j;\mathbf{d}^j;...\mathbf{d}^1] \tag{2.24c}$$

or approximately

$$\mathbf{f} \approx \mathbf{T}^{j-1}[\mathbf{a}^j;\mathbf{0}] \tag{2.24d}$$

where, in this context, $\mathbf{0}$ is an $(n - n/2^j) \times 1$ vector of zeros. Note that only the first $n/2^j$ columns of $\mathbf{T}^{j-1}$ are needed to reconstruct f from $\mathbf{a}^j$. Therefore, we define truncated quantities, $(\cdot)'$, which include only the columns required to compute or invert the approximation coefficients. Note, however, that the complete forward transform is still necessary to construct the truncated inverse.

Up until now, we have considered a 1D signal, $\mathbf{f}$, but in practice, μ is a 3D field and $\boldsymbol{\mu}$ is a vector of coefficients for a 3D voxel basis or a similar multidimensional representation. To compute the wavelet transform of a discrete nD field, we use a separable wavelet transform, applying it successively in each

dimension. In 2D, a single level of the transform generates a set of approximation coefficients, two sets of cross-coefficients (both combining approximation coefficients from one dimension and detail coefficients from the other), and detail coefficients. For an nD signal, the transform produces $2n$ combinations of approximation and detail coefficients.

Special attention must be given to arranging the matrices for the block formulation to ensure that all approximation coefficients are grouped together in the resulting vector, as well as the detail coefficients and sets of cross-coefficients. The final truncated forward and inverse transforms of the target field are

$$\mathbf{T}'^{j}_{3\mathrm{D}}\boldsymbol{\mu} = \boldsymbol{\mu}^{j}_{\mathrm{wav}} \tag{2.25a}$$

$$\boldsymbol{\mu} \approx \mathbf{T}'^{j\ -1}_{3\mathrm{D}}\boldsymbol{\mu}^{j}_{\mathrm{wav}} \tag{2.25b}$$

where μ^{j}_{wav} represents the approximation coefficients from the jth-level wavelet transform of μ. Since the wavelet transform is a linear operation, we retain the discrete forward model established for the voxel basis,

$$\mathbf{p} = \mathbf{A}\mathbf{T}'^{j\ -1}_{3\mathrm{D}}\boldsymbol{\mu}^{j}_{\mathrm{wav}} \tag{2.26}$$

In practice, determining μ j wav involves a multi-resolution analysis. This process begins at the coarsest scale, $j = J$, where μ^{j}_{wav} is first reconstructed. The solution at this level is then used as the starting point to reconstruct μ^{J-1}_{wav} and this process continues iteratively up to the finest level of interest. Implicit regularization of the reconstruction can be achieved by truncating the sequence at a level before $j = 1$.

C.2.3. Continuous Neural Fields

Coordinate-based neural networks, such as neural radiance fields (NeRF), are powerful tools for representing complex 3D and 4D scenes. These networks have been successfully used in fields like computer vision, data assimilation, and tomography. They model target fields within O as continuous functions of the spatial coordinates **x**. This continuous mapping allows NeRF and similar models to capture fine details from limited projections, enabling efficient storage, rendering, and manipulation of high-dimensional data for high-resolution reconstructions. The neural-implicit approach offers several advantages: it provides a compressed representation of the target field, leverages machine learning environments for optimized implementation, and can facilitate advanced prior information, such as physics-based governing equations. It also supports truly 4D reconstructions of time-resolved, multi-view X-ray radiographs, including bona fide space–time regularization. Moreover, the continuous imaging models can be approximated to arbitrary precision.

Neural-implicit extinction coefficient fields are defined using a neural network,

$$\mathcal{N}:(\mathbf{x},t)\mapsto\mu \tag{2.27}$$

Here, the neural network N outputs a mean or monochromatic extinction coefficient, μ, as a function of **x**. The network could also be adapted to predict ρ (density) or ρ plus a set of mass fractions from

which μ or even μ_E is derived. With representation of the target field in Eq. (2.27), the imaging model from Eq. (2.8) is approximated as

$$p_i = \frac{1}{|\mathcal{X}_i|}\int_{\mathcal{V}_i} \mu_E(\mathbf{x})\mathrm{d}\mathbf{x} \approx \frac{|\mathcal{V}_i|}{|\mathcal{X}_i| n_{\mathrm{smp}}}\sum_{k=1}^{n_{\mathrm{smp}}} \mu(\mathbf{x}_k) \tag{2.28}$$

In this formulation, μ is computed by evaluating N at random points $\mathbf{x}_k$ that are uniformly sampled from $\mathcal{V}_\mathrm{i}$. This process is repeated for each element of p, and the network is trained to match experimental projection data while satisfying any prior constraints included in its objective function. Importantly, when the simplifications used to derive Eq. (2.8) break down, N can be used to approximate higher-fidelity models, such as those in Eq. (2.5) or (2.7). The neural networks used in our reconstruction algorithm consist of an input layer, an output layer, and a series of n_l hidden layers, N can be used to approximate higher-fidelity models, such as those in Eq. (2.5) or (2.7).

The neural networks used in our reconstruction algorithm consist of an input layer, an output layer, and a series of n_l hidden layers,

$$\mathcal{N}(\mathbf{z}^0) = \mathbf{W}^{n_\mathrm{l}+1}\left[\mathcal{L}^{n_\mathrm{l}} \circ \cdots \circ \mathcal{L}^2 \circ \mathcal{F}(\mathbf{z}^0)\right] + \mathbf{b}^{n_\mathrm{l}+1} \tag{2.29a}$$

with

$$\mathbf{z}^l = \mathcal{L}(\mathbf{z}^{l-1}) = \sigma(\mathbf{W}^l\mathbf{z}^{l-1} + \mathbf{b}^l) \quad \text{for} \quad l \in \{2,3,\ldots,n_\mathrm{l}\} \tag{2.29b}$$

Here, $\mathbf{z}^l$ represents the values of the neurons in the lth layer, $\mathbf{W}^l$ is the weight matrix, and $\mathbf{b}^l$ is the bias vector for the lth layer. The function s is a nonlinear activation function applied element-wise to its input. Note that z^0 corresponds to the input coordinate, x, or to (x, t) in 4D cases, and z^{nl+1} is the output, i.e., μ in Eq. (2.27).

The vector θ contains all the trainable weights and biases in the neural network, N. We use swish activation functions,

$$\sigma(z) = \frac{z\exp(z)}{1+\exp(z)} \tag{2.30}$$

which have been shown to improve the stability of gradient flow during training compared to hyperbolic tangent functions and others. To further mitigate the spectral bias inherent in gradient descent training, we replace the $\mathcal{L}^1$ input layer with a Fourier encoding,

$$\mathbf{z}^1 = \mathcal{F}(\mathbf{z}^0) = \left[\sin(2\pi\boldsymbol{\omega}_1\cdot\mathbf{z}^0), \cos(2\pi\boldsymbol{\omega}_1\cdot\mathbf{z}^0), \ldots, \sin(2\pi\boldsymbol{\omega}_k\cdot\mathbf{z}^0), \cos(2\pi\boldsymbol{\omega}_k\cdot\mathbf{z}^0)\right]^\mathsf{T} \tag{2.31}$$

In this layer, k is the number of Fourier features and $\boldsymbol{\omega}$ is a vector of random frequencies, fixed before training; each element of z^0 is associated with a unique frequency, and it is common to pick different distributions for spatial and temporal coordinates in time-resolved applications. This neural network approach provides significant data compression, allowing for a low-dimensional parameterization of

the target field and enabling robust optimization. Additionally, using neural networks allows us to leverage ongoing advances in machine learning algorithms and hardware.

C.3. Tomographic Reconstruction

The previous section developed a rigorous physics-based model of X-ray radiography and a sequence of simplified models. These models were then discretized using voxel and wavelet bases as well as a continuous representations based on coordinate neural networks. The next step is to invert these models to reconstruct a 3D function from its 2D projections—a process known as tomographic reconstruction. Tomography is a classic instance of an ill-posed problem, meaning that there are multiple possible solutions or that small changes in the data can lead to large changes in the reconstruction. To address this, a range of algorithms has been developed, broadly categorized into analytical, algebraic, and neural methods.

Analytical methods, such as FBP, assume a dense and evenly spaced set of projections, which can provide accurate reconstructions of static objects under ideal conditions. However, under the practical constraints imposed by the ultra-high-speed imaging applications of interest, the number of available views is limited, and analytical approaches are not viable in these scenarios. Algebraic methods, on the other hand, are designed to handle limited and irregularly spaced projections. They can incorporate prior information about the target object, such as physics-inspired constraints in the form of a penalty term, to better condition the reconstruction problem and make it more robust to noise. Neural approaches extend this further by leveraging machine learning techniques to learn a complex mapping from x to μ directly from the data, offering a powerful tool for cases with very sparse projections or where advanced priors can significantly enhance reconstruction quality.

This section will discuss the fundamentals of tomographic reconstruction, detailing the ill-posed nature of the problem, the analytical and algebraic algorithms used to solve it, and the development of a neural algorithm.

C.3.1. Reconstruction as an Ill-Posed Inverse Problem

Tomographic reconstruction is an inherently ill-posed inverse problem that divides into two categories: discrete ill-posed problems and rank-deficient ones. **Discrete ill-posed problems** occur when there is a dense set of projection data, and the forward (measurement) operator is fully determined, even for a high-resolution basis. The forward equations dampen high-frequency components of the extinction coefficient μ in the line-integrated projections, p. When inverting these equations to reconstruct the target field, noise and model errors can cause large, non-physical spikes in μ due to frequency amplification. To mitigate these artifacts, spectral filters—such as those used in FBP algorithms—are applied to stabilize the solution.

Rank-deficient problems arise in limited-data tomography when the number of projections is much smaller than the number of unknown parameters to estimate. In such cases, the forward operator, $\mathbf{A}$, has a column rank below the number of projections in $\mathbf{p}$, resulting in infinitely many solutions for the system $\mathbf{A\mu} = \mathbf{p}$. Least-squares inversion of this underdetermined system typically produces sparse, non-physical solutions that fail to accurately recover the internal structure of the target object. To obtain a unique and physically plausible estimate, therefore, additional information must be incorporated into the reconstruction process, termed regularization. Ultrahigh- speed X-ray imaging systems commonly face this type of ill-posedness due to limitations on the number of projection

angles that can be captured experimentally. As a result, an advanced reconstruction algorithm that effectively integrate prior information is needed to enhance reconstruction quality and reliability.

What follows is an overview of the reconstruction algorithms included in our software package, with a focus on how incorporating prior information and regularization can effectively address the ill-posed nature of tomographic reconstruction.

C.3.2. Analytical Reconstruction Algorithms

Analytical reconstruction algorithms represent the target field as an explicit function of the projection data. They are conceptually based on the Fourier slice theorem, which establishes an equality between the 1D Fourier transform of the projection data and the 2D Fourier transform of a cross-section of the object. This theorem serves as a theoretical foundation for analytical methods, enabling the evaluation of reconstruction properties, such as spatial resolution as a function of the angular sampling density.

Consider a 2D extinction coefficient field $\mu \in R^2$ that decays to zero as x and y approach ±8. Suppose we have a continuous set of projection data, $(r,\theta)\mapsto p$, where $r \in R$ is the radial offset from the origin and $\theta \in [0,\pi)$ is the angle of the projection. According to the Fourier slice theorem,

$$(\mathfrak{F}_{\mathrm{r}}p)(\omega_r,\theta) = \left(\mathfrak{F}_{\mathrm{x,y}}\mu\right)(\omega_{\mathrm{x}},\omega_{\mathrm{y}})\Big|_{\omega_{\mathrm{x}}=\omega_{\mathrm{r}}\cos(\theta)\,,\omega_{\mathrm{y}}=\omega_{\mathrm{r}}\sin(\theta)} \tag{3.1}$$

where ω_r, ω_{x}, and ω_{y} are the frequencies in the r-, x-, and y-directions. In this expression, $\mathfrak{F}_{\mathrm{r}}$ is the 1D Fourier transform in the radial offset domain and $\mathfrak{F}_{\mathrm{x,y}}$ is the 2D Fourier transform in the x–y plane. This equation implies that the 1D Fourier transform of a projection at a given angle θ. corresponds to a line through the 2D Fourier transform of μ at the same angle. By collecting projections over a range of angles, the 2D Fourier space of the object field, $\mathfrak{F}_{\mathrm{x,y}}\mu$, can be populated. An estimate of $\mathfrak{F}_{\mathrm{x,y}}\mu$, denoted $\hat{\mu}$, is obtained by combining the Fourier-transformed projections. Taking the inverse 2D Fourier transform of $\hat{\mu}$ then yields an estimate of the original field,

$$\mu \approx \mathfrak{F}_{\mathrm{x,y}}^{-1}\hat{\mu} \tag{3.2}$$

An important conclusion from the Fourier slice theorem is that the reconstruction process inherently amplifies high-frequency artifacts in $\hat{\mu}$. To mitigate the impact of noise and suppress these artifacts, filters are incorporated into analytical algorithms.

Filtered back-projection is the most widely used analytical reconstruction algorithm. It operates on discrete data, as real X-ray sensors produce a finite "sinogram" in (r,θ)-space rather than continuous projections. We represent the sinogram with an $m \times n$ matrix P, where n is the total number of equally spaced projections and m is the number of pixels per projection. Our FBP algorithm processes each row of the 2D detector sinogram independently, effectively performing a series of 2D reconstructions

to build a 3D field—an approach known as 2.5D reconstruction— which is commonly used in X-ray tomography applications.

To reduce noise-related errors, the sinogram is filtered through convolution in the radial offset space,

$$P^{\star}_{i,j} = \sum_{k=-n_k}^{n_k} h_k P_{i-k,j} \tag{3.3}$$

where P^{*} represents the filtered sinogram, and h is a filter of length $2n_k + 1$. Note that $P_{i-k,j}$ zero for $i-k<1$ and $i-k>m$. We implemented the two most commonly used filters for FBP: the Ram–Lak (RL) filter and the Shepp–Logan (SL) filter, represented by $\mathbf{h}_{\mathrm{RL},k}$ and $\mathbf{h}_{\mathrm{SL},k}$. The RL filter is defined as

$$h_{\mathrm{RL},k} = \begin{cases} \frac{1}{4d^2}, & k = 0 \\ -\frac{1}{(4\pi d)^2}, & k \neq 0, k \text{ is even} \\ 0, & k \text{ is odd} \end{cases} \tag{3.4}$$

and the SL filter is given by

$$h_{\mathrm{SL},k} = -\frac{2}{\pi^2 d^2 (4k^2-1)} \tag{3.5}$$

In both cases, d represents the side length of a pixel on the detector. Analogous to the Fourier slice theorem, FBP provides a direct expression of the target field in terms of its filtered projections,

$$\mu(x,y) = \frac{1}{n}\sum_{j=1}^{n} \eta P^{\star}_{i,j} + (1-\eta) P^{\star}_{i+1,j} \tag{3.6}$$

where i and η depend on the spatial coordinates (x, y) and projection angle θ_j. The relationship between the radial offset and the spatial location is given by

$$x\cos(\theta_j) + y\sin(\theta_j) = (i+\eta)d \tag{3.7}$$

which can be solved for i and η with constraints $i \in \{1,...,m\}$ and $\eta \in [0,1)$. While FBP performs well with clean, densely sampled data, it struggles with limited-data scenarios, characteristic of ultra-high-speed imaging applications, leading to poor reconstructions in such cases.

C.3.3. Algebraic Reconstruction Algorithms

Algebraic reconstruction algorithms provide an effective alternative to analytical methods, particularly in limited-data scenarios. These techniques invert a discrete formulation of the measurement equations, $\mathbf{A\mu} = \mathbf{p}$ (Eq. (2.10)), where the target field is represented using voxels, wavelets, or other suitable basis functions. Unlike FBP algorithms, which require dense angular sampling, algebraic methods are robust to sparse or irregular projection data. They can also incorporate sophisticated prior information, such as spatial smoothness or statistical properties of time-resolved fields, through regularization. This flexibility makes algebraic methods particularly advantageous when experimental constraints limit the number of projections, including ultra-high-speed X-ray tomography. In the

following sections, we will explore key approaches, including the simultaneous algebraic reconstruction technique (SART), multiplicative algebraic reconstruction technique (MART), and classical regularization methods such as Tikhonov (L^2) and Total Variation (TV, L^1) smoothness penalties.

C.3.3.1 Simultaneous Algebraic Reconstruction Technique

Algebraic reconstruction techniques (ARTs) iteratively update a discrete estimate of the target field, μ^k , where k is the iteration index, to solve a matrix system like the one in Eq. (2.10). The first and most basic ART method is the additive ART,

$$\boldsymbol{\mu}^{(k+1)} = \boldsymbol{\mu}^{(k)} + \gamma \frac{p_i - \mathbf{A}_{i,*}\boldsymbol{\mu}^{(k)}}{\|\mathbf{A}_{i,*}\|_2^2} \mathbf{A}_{i,*}^{\top} \tag{3.8}$$

where $\mathbf{A}_{i,*}$ indicates the ith column of A, $\gamma \in (0,2]$ is a relaxation parameter, and the initial estimate μ^{k+1} is typically set to 0. Note that $\|\cdot\|_2$ denotes the Euclidean norm, we also make use of the Manhattan norm, $\|\cdot\|_1$, throughout this report. While historically significant, the additive ART method requires an inner loop over each projection at every iteration, which can number in the hundreds of thousands to millions. This makes the computation slow in a scripted programming environment, limiting its practicality for some large-scale tomographic problems.

Simultaneous iterative reconstruction technique (SIRT) methods belong to a family of algorithms that minimize the residuals for all projections at each iteration. Among these, the SART is the most widely known. SIRTs, including SART, are particularly well-suited for array programming environments like MATLAB or Python, where they can leverage efficient built-in matrix operations, providing faster computations compared to the additive ART. A generic SIRT update is given by

$$\boldsymbol{\mu}^{(k+1)} = \boldsymbol{\mu}^{(k)} + \gamma_k \mathbf{N}\mathbf{A}^{\top}\mathbf{M}\left(\mathbf{p} - \mathbf{A}\boldsymbol{\mu}^{(k)}\right) \tag{3.9}$$

where, γ_k is the relaxation parameter for the kth iteration and M and N are $m \times m$ and $n \times n$ diagonal matrices, respectively. The matrix M weights residuals along each LoS while N weights the back-projection operation at each basis function. The specific definitions of the elements of M and N depend on the SIRT variant—including Cimmino's method, Landweber iteration, SART, etc. The SART is defined by

$$M_{i,i} = \|\mathbf{A}_{i,*}\|_1^{-1} \quad \text{and} \quad N_{j,j} = \|\mathbf{A}_{*,j}\|_1^{-1} \tag{3.10}$$

Iterations of Eq. (3.9) converge to a weighted least-squares solution,

$$\boldsymbol{\mu}_{\text{SART}} = \arg\min_{\boldsymbol{\mu}} \|\mathbf{M}(\mathbf{A}\boldsymbol{\mu} - \mathbf{p})\|_2^2 \tag{3.11}$$

However, ART methods, including SART, do not inherently incorporate spatial information about μ. As a result, if there are voxels or wavelets that do not directly contribute to any of the projections in

p, these algorithms leave those regions unchanged from their initial guess. This often leads to streaky or gappy features in the solution.

C.3.3.2 Multiplicative Algebraic Reconstruction Technique

Multiplicative ART algorithms update $\boldsymbol{\mu}$ using a multiplicative approach rather than an additive one. This method has several advantages, such as ensuring that voxels initialized to zero remain zero throughout the reconstruction, which improves the speed of convergence. Further, the formulation maximizes the Kullback–Leibler divergence between $\boldsymbol{\mu}^{(0)}$ and $\boldsymbol{\mu}^{(k)}$ as k grows large, which promotes spatially sparse fields, which is advantageous in some contexts. There are several MART variants, and our package implements the following:

$$\mu_j^{(k+1)} = \mu_j^{(k)} \left(\frac{p_i}{\mathbf{A}_{i,*}\boldsymbol{\mu}^{(k)}} \right)^{\gamma \frac{A_{i,j}}{A_{\max,i}}} \tag{3.12}$$

Typically, γ is confined to (0, 1]. As with additive ART, each MART iteration corresponds to a single projection, p_i, and the update is applied to each element of μ upmu before advancing to the next iteration k. MART demonstrates semi-convergence, which requires iterative regularization (premature halting of the algorithm), and the rate of convergence can be influenced by the order in which projections are selected. This algorithm performs particularly well for sparse target fields like a cloud of particulate matter generated by a blast. However, for continuous fields with broadband frequency content, MART often performs poorly or no better than additive methods.

C.3.4. Classical Regularization Penalties

Explicit regularization methods for X-ray tomography incorporate additional information to obtain a unique, stable, and physically plausible estimate. This is achieved by including a penalty term that discourages undesirable characteristics in the reconstruction, such as erratic variations in μ. For example, penalizing the spatial derivatives of μ can reduce noise and enforce smoothness. However, this penalty is not strictly aligned with the measurement equations or the true solution in most practical imaging scenarios, resulting in a joint optimization problem. First, the residuals from Eq. (2.10) are summed in a data fidelity term:

$$\mathcal{L}_{\text{data}} = \|\mathbf{A}\boldsymbol{\mu} - \mathbf{p}\|_2^2 \tag{3.13}$$

Next, a penalty or regularization term is added, typically (although not always) of the form

$$\mathcal{L}_{\text{reg}} = \|\mathbf{L}\boldsymbol{\mu}\|^2 \tag{3.14}$$

where the matrix $\mathbf{L}$ depends on the type of regularization, as discussed below. The objective is to minimize

$$\mathcal{L}_{\text{total}} = \|\mathbf{A}\boldsymbol{\mu} - \mathbf{p}\|_2^2 + \gamma\|\mathbf{L}\boldsymbol{\mu}\|^2 \tag{3.15}$$

Here, $\|\cdot\|$ is an arbitrary norm that depends on the regularization scheme and γ is a tunable parameter that balances the data fidelity and the regularization terms. It must be carefully selected to ensure the

best trade-off between fitting the data and enforcing desired properties in the solution. The regularized solution is then obtained by solving the following optimization problem:

$$\boldsymbol{\mu}_\gamma = \arg\min_{\boldsymbol{\mu}} \mathcal{L}_{\text{total}} \tag{3.16}$$

The strategy for solving Eq. (3.16) depends on the structure of $\mathbf{L}$. Our software package includes two of the most common regularization techniques for X-ray tomography: Tikhonov regularization, which promotes global smoothness, and TV regularization, which enforces piecewise smoothness. These methods are presented in next two subsections.

Several methods have been developed to automate the selection of the regularization parameter γ, with the most popular being L-curve analysis. In this approach, the optimal parameter is estimated by examining the implicit curve traced by $\boldsymbol{\mu}$. in a plot of $\log(\mathcal{L}_{reg})$ vs. $\log(\mathcal{L}_{data})$ as a function of γ. This curve is typically shaped like the letter "L": for small values of γ, $\mathcal{L}_{data}$ is minimized but $\mathcal{L}_{reg}$ is high, and the opposite occurs for large values of γ. The optimal trade-off between these poles is at the "bend" of the L, where the curvature of the plot is maximized. This point is referred to as γ^*, and in the absence of additional prior information about the target field and the distribution of measurement noise, γ^* is the recommended choice for γ.

C.3.4.1 Tikhonov Regularization

Tikhonov regularization introduces a penalty term to minimize the derivatives of the target field. The most commonly used form in tomography is second-order Tikhonov regularization. In continuous form, the second-order Tikhonov penalty approximates the following term:

$$\mathcal{L}_{\text{Tik}} = \frac{1}{|\Omega|}\int_\Omega \|\nabla^2 \mu(\mathbf{x})\|_2^2 \mathrm{d}\mathbf{x} \tag{3.17}$$

where

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \tag{3.18}$$

is the Laplacian operator. This operator can be discretized to form the matrix $\mathbf{L}_{\text{Tik}}$, such that the discrete second-order Tikhonov penalty is

$$\mathcal{L}_{\text{Tik}} = \|\mathbf{L}_{\text{Tik}}\boldsymbol{\mu}\|_2^2 \tag{3.19}$$

Note that the Euclidean norm has the following property:

$$\|\mathbf{x}\|_2^2 = \mathbf{x}^\top \mathbf{x} \tag{3.20}$$

which allows for a convenient simplification of the objective function from Eq. (3.15),

$$\mathcal{L}_{\text{total}} = \|\mathbf{A}\boldsymbol{\mu} - \mathbf{p}\|_2^2 + \gamma^2 \|\mathbf{L}_{\text{Tik}}\boldsymbol{\mu}\|_2^2 \tag{3.21a}$$

$$\mathcal{L}_{\text{total}} = \left\| \begin{bmatrix} \mathbf{A} \\ \gamma \mathbf{L}_{\text{Tik}} \end{bmatrix} \boldsymbol{\mu} - \begin{bmatrix} \mathbf{p} \\ \mathbf{0} \end{bmatrix} \right\|_2^2 \tag{3.21b}$$

Here, γ is squared to simplify the augmented system matrix. This formulation makes the optimization problem straightforward: $\boldsymbol{\mu}_\gamma$ is the least-squares solution to the augmented system. However, the size of the augmented system can be very large, making it impractical to store the inverse of the augmented

operator in memory (as it is not generally sparse). To address this, iterative methods such as the SART algorithm can be used to efficiently compute the regularized solution.

C.3.4.2 Total Variation Regularization

The TV norm is another widely used penalty for tomographic reconstruction, particularly in scenarios where the target field exhibits piecewise smoothness with distinct boundaries, such as in imaging solid objects or flames. Unlike Tikhonov regularization, which penalizes the square of second derivatives and favors globally smooth solutions, TV regularization penalizes only the total gradient of the target field. This property allows TV regularization to preserve sharp edges while promoting smoothness within contiguous regions, making it well-suited for reconstructing fields with a small number of large discontinuities.

The TV regularization penalty is defined with respect to the TV norm. For a continuous 3D field $\boldsymbol{\mu}$ in the domain $\boldsymbol{\Omega}$, the TV norm is given by:

$$\mathcal{L}_{\mathrm{TV}} = \|\mu\|_{\mathrm{TV}} = \frac{1}{|\Omega|}\int_{\Omega} \|\nabla\mu(\mathbf{x})\|_1 \mathrm{d}\mathbf{x} \tag{3.22}$$

where $\nabla = [\delta/\delta x, \delta/\delta y, \delta/\delta z]^{\top}$. is the gradient operator. This norm measures the sum of the magnitudes of local gradients, effectively promoting smooth regions while still allowing for sharp transitions. For the discrete case, the TV penalty involves summing the local gradients of $\boldsymbol{\mu}$ over all voxels. The resultant α-direction gradient field, for $\alpha \in \{x,y,z\}$, at the jth voxel is

$$(\nabla_{\alpha}\mu)_j = \mathbf{D}_{\alpha,j,*}\boldsymbol{\mu} \tag{3.23}$$

where D_{α} is a discrete gradient operator acting in the α-direction and is typically implemented using finite-difference stencils; $\mathbf{D}_{\alpha,j,*}$. represents the jth row of D_{α}. Using these terms, the discrete TV penalty is

$$\mathcal{L}_{\mathrm{TV}} = \|\boldsymbol{\mu}\|_{\mathrm{TV}} = \sum_{j=1}^{n}\left[\sum_{\alpha}(\nabla_{\alpha}\mu)_j^2\right]^{1/2} \tag{3.24}$$

where α sums over all three Cartesian directions. A TV-regularized reconstruction is obtained by solving the following optimization problem,

$$\boldsymbol{\mu}_{\gamma} = \arg\min_{\boldsymbol{\mu}} \|\mathbf{A}\boldsymbol{\mu} - \mathbf{p}\|_2^2 + \gamma\|\boldsymbol{\mu}\|_{\mathrm{TV}} \tag{3.25}$$

Unlike Tikhonov regularization, this problem is non-convex and more challenging to solve due to the non-smooth nature of the TV norm. To address this, we use a modified fast iterative shrinkage-thresholding algorithm (FISTA), which alternates between SIRT iterations for data fidelity and gradient-based optimization to minimize the TV penalty. This method efficiently balances smoothness within regions and sharp boundaries, even in complex reconstructions.

In order to minimize the TV penalty by gradient descent, it is necessary to compute the gradient of $\|\boldsymbol{\mu}\|_{\mathrm{TV}}$ with respect to $\boldsymbol{\mu}$. This gradient is given by

$$\frac{\partial\|\boldsymbol{\mu}\|_{\mathrm{TV}}}{\partial\mu_j} = \sum_{k=1}^{n}\left\{\sum_{\alpha} D_{\alpha,j,k}\frac{(\nabla_{\alpha}\mu)_j}{\left[\sum_{\alpha}(\nabla_{\alpha}\mu)_j^2\right]^{1/2}}\right\} \tag{3.26}$$

It is common to use forward or backward differences in D_α to avoid instabilities with central differencing when computing this gradient. An element-wise non-negativity constraint is enforced at each iteration of the SIRT-FISTA-TV algorithm.

Figure A-10. Demonstration of TV regularization on a 2D slide of the Shepp–Logan phantom: (left) ground truth distribution, (middle) TV estimate, and (right) SART estimate.

C.3.5. Neural-Implicit Reconstruction Technique

Lastly, we discuss the neural-implicit reconstruction technique (NIRT), which uses a coordinate-based neural network to represent the target field. Similar to the classical regularization methods NIRT minimizes an objective loss function,

$$\mathcal{L}_{\text{total}} = \gamma_0 \mathcal{L}_{\text{data}} + \sum_i \gamma_i \mathcal{L}_{\text{reg},i} \tag{3.27}$$

where $\mathcal{L}_{\text{data}}$ is the data loss and $\mathcal{L}_{\text{reg},i}$ are regularization terms that can incorporate sophisticated prior information, such as temporal dynamics, differences between μ and previous scans, or even the distance from μ to a design specification. While we typically use a continuous version of LTik (Tikhonov regularization) or LTV (TV regularization), additional terms have been employed throughout the report to address specific needs. The loss weight parameters $\{\gamma_0,\gamma_1,\ldots\}$ are used to balance the contributions of each component, ensuring that the network effectively captures both the data and the (weak) prior constraints.

By minimizing Eq. (3.27), the neural network learns to represent μ as a continuous function of $\mathbf{x}$ over the domain Ω, or as a function of $(\mathbf{x}, t)$ in $\Omega \times \mathcal{T}$. The vector θ, which contains all the network's weights and biases and determines the mapping $\mathbf{x} \mapsto \mu$, is updated using a backpropagation algorithm to minimize $\mathcal{L}_{\text{total}}$. This process ensures that the neural representation aligns with both the experimental data and the prior information encoded in the regularization terms, allowing the network to capture the desired characteristics of the target field.

The data fidelity term is defined as

$$\mathcal{L}_{\text{data}} = \|\mathbf{p} - \tilde{\mathbf{p}}\|_2^2 \tag{3.28}$$

where $\mathbf{p}$ contains the experimental radiography data and $\tilde{\mathbf{p}}$ is the predicted projection generated by the neural network N using Eq. (2.28) for each pixel/beam. Continuous Tikhonov or TV penalties can be approximated using Eq. (3.17) or (3.22), respectively. In these cases, integrals over Ω ($\Omega \times \mathcal{T}$ with respect to $dx\ dt$ for 4D scenarios) are approximated through Monte Carlo sampling.

For example, the Tikhonov regularization term may be estimated as follows

$$\mathcal{L}_{\text{Tik}} \approx \frac{1}{n_{\text{smp}}} \sum_{k=1}^{n_{\text{smp}}} \nabla^2 \mu(\mathbf{x}_k) \tag{3.29}$$

where the points $\mathbf{x}_k$ are uniformly sampled from Ω. The derivatives required to compute $\nabla^2\mu$ are obtained via automatic differentiation of N, providing exact partial derivatives of the continuous neural representation.

To facilitate effective reconstruction, a gradient-based auto-weighting scheme has been integrated into the software to automatically fine-tune the weight parameters γ,

$$\gamma_0 = \frac{\|\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\text{data}}\|_2 + \sum_j \|\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\text{reg},j}\|_2}{\|\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\text{data}}\|_2} \tag{3.30a}$$

and

$$\gamma_i = \frac{\|\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\text{data}}\|_2 + \sum_j \|\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\text{reg},j}\|_2}{\|\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\text{reg},i}\|_2} \tag{3.30b}$$

where ∇_θ is the automatic differentiation operator that computes partial derivatives with respect to the network parameters, θ. This auto-weighting scheme minimizes the need for manual tuning and ensures high-quality reconstructions, provided that appropriate regularization terms are chosen.

C.4. Software Package

The software package deliverable offers users a streamlined workflow for reconstructing static and dynamic fields from synthetic or experimental X-ray radiography data. This package supports a wide range of measurement scenarios and reconstruction algorithms, allowing users to specify custom parameters for each case. The code's modular design and use of object-oriented principles enables advanced users to develop their own scripts that utilize the package's core functionality, making it versatile and adaptable to tomographic applications at Sandia National Laboratories. In this section, we discuss the details of the package and provide demonstrations of its capabilities.

The software is organized into five distinct modules that work sequentially to perform a complete tomographic reconstruction from 2D projection data. The modules are the:

1. Initialization Module – This module reads a set of user-provided input files that define the measurement scenario and details of the reconstruction algorithm. Files can also be provided to specify custom phantoms. It sets up the necessary internal parameters and configurations for subsequent modules.
2. Scene Setup Module – This module creates a virtual representation of measurement scenario, including the layout of the scene, configuration of detectors, reconstruction domain parameters, and selection/specification of a measurement model. It establishes the spatial and physical relationships necessary for accurate simulation and reconstruction.
3. Phantom Generation Module – This module creates a discrete representation of custom phantoms and generates their synthetic projections. It can also import user-provided CAD models, allowing for the simulation of complex geometries. If experimental data are provided, this module is bypassed.
4. Reconstruction Module – This module performs the tomographic reconstruction using the specified algorithm. It includes various reconstruction techniques, including analytical, algebraic, and neural methods, to produce high-quality estimates of the target field. Batch reconstructions of time-series data can be performed, and reconstructions can be automatically repeated, using distinct values of any regularization parameters.

5. Plotting and Analysis Module – The final module visualizes the results of the reconstruction. It generates comprehensive plots and analysis of the reconstructed fields, providing users with tools to assess the quality and accuracy of their simulations or experiments.

Each module is designed to function independently while being easily integrated into a complete workflow. This architecture allows users to customize and extend the software for their specific needs, whether for academic research, industrial applications, or educational purposes. The following subsections provide detailed descriptions and demonstrations of each module's functionality.

C.4.1. Initialization

The initialization module reads user-provided input files that define the measurement scenario, custom phantoms, and selection of reconstruction algorithms and regularization parameters. This information is organized into three separate input files:

1. Scene Input File (scene.inp) – The scene.inp file specifies details about the measurement scenario and is divided into two sections: domain parameters and detector parameters. Domain specification includes parameters that determine its physical size, spatial location, and level of discretization for the reconstruction domain. The detector section gives the configuration of the X-ray sources and sensors. For each detector, this file specifies its detector pose in the form of a camera transform, the ray geometry, and the sensor size. Any parameters not provided by the user are assigned default values based on a standard reference measurement scenario.
2. Phantom Input File (phantom.inp) – The phantom.inp file allows users to define custom phantoms or load a CAD model. Custom phantoms are constructed from geometric primitives— spheres, ellipsoids, cones, and rectangular prisms. Users can specify any number of primitives, each defined by six parameters:
   - Primitive Type – The shape of the geometric object (e.g., sphere, cone).
   - Bounds – Dimensions of the shape
   - Origin – The position of the object's centroid.
   - Orientation – A rotation matrix that specifies its pose.
   - Sample Count – The number of samples per basis function to minimize aliasing artifacts.
   - Density – A material density that can be converted to ì. These parameters are used to create a composite field. Alternatively, users can provide a file path to load a standard triangle language (STL) CAD model, which is directly integrated into the simulation.
3. Reconstruction Input File (reconstruction.inp) – The reconstruction.inp file specifies the reconstruction algorithms to be used. Users can choose any of the analytical or algebraic algorithms—FBP, SART, Tikhonov, etc. General parameters like the number of iterations are applicable across all algorithms, while additional parameters are required for specific methods, such as regularization weights for non-convex techniques like TV regularization. Multiple algorithms can be specified in the file, and the results for each will be generated and compared in the final module.

Figure A-11. Sample input files for the initialization module: (left) scene.inp, (middle) phantom.inp, and (right) reconstruction.inp.

C.4.2. Scene Module

The scene module constructs the measurement scenario based on the information provided in the scene.inp file. This includes setting up the reconstruction domain object, configuring detector objects, and indicating the ray geometry that will be used by subsequent modules. These elements are represented through two primary objects: the detector object and the domain object.

1. Detector Object – The detector object models the trajectory of light rays traveling from the source to the pixels on the detector. Ray paths are defined according to the beam geometry, the location of the source and detector (i.e., the source-to-object and source-to-detector distances, SOD and SDD), and the beam geometry—thin, cylindrical, cone, etc.—specified in the scene.inp file. This object ensures accurate modeling of the light-ray paths for the measurement scenario.
2. Domain Object – The domain object represents the reconstruction domain in terms of its physical dimensions and the number of basis functions (e.g., voxels, wavelets) used to discretize the domain. It provides the spatial framework within which the tomographic reconstruction is performed.

Using the information from both the detector and domain objects, the module constructs the measurement model for each detector object. These models are stored for use in subsequent modules to perform tomographic reconstructions.

C.4.3. Phantom Module

The phantom module processes user inputs specified in the phantom.inp file to create synthetic phantoms and calculate and store their projections. This module supports two types of phantoms: composite geometric primitives and CAD models based on an STL file.

C.4.3.1 Types of Phantoms Supported

1. Composite Primitives – Users can define phantoms composed of basic geometric shapes such as spheres, cones, and cubes. These shapes are combined to create synthetic fields with a complex geometry and significant spectral content associated with myriad intersecting edges. Each primitive is defined by parameters such as type, size, orientation, and density. Discrete representation of these fields is implemented by projecting them onto the specified basis using Monte Carlo sampling per Eq. (2.11).

2. CAD Models – Users can also provide CAD models in the form of an STL file to test the reconstruction workflow with a realistic imaging setup for complex geometries that are representative of an end application. In order to provide a ground truth field for benchmarking, the CAD models are converted into voxel-based fields, and each voxel's density is determined through Monte Carlo sampling to ensure accurate representation of the model's geometry.

C.4.3.2 Projection Computation

For both geometric primitives and CAD models, the module projects the fields onto a voxel grid. This process involves Monte Carlo sampling within each voxel to generate a smooth representation of the phantom and mitigate aliasing artifacts at the edges. The voxel-based representation of CAD models is applied to the entire field, while the individual geometric primitives are combined to form a composite phantom.

In addition to creating discrete 3D volumes, the module also computes analytical 2D projection images for each detector in the measurement scene:

1. Primitive Phantoms – Exact projections of the primitive phantoms are computed using analytical formulas, allowing for accurate and efficient calculation of line integrals through the geometric shapes. For complex beam geometry, exact intersections are calculated for each ray and the overall projection is computed by sampling rays from the beam.
2. CAD Phantoms – Projections of CAD-based phantoms are calculated using a fast ray–triangle intersection algorithm, which determines the exact length of individual rays inside the phantom. Hence, the accuracy of these projections is limited primarily by the accuracy of the CAD model.

Both the 3D voxel/wavelet representations and the 2D projection images of the phantoms are stored for use in subsequent reconstruction and plotting/analysis modules.

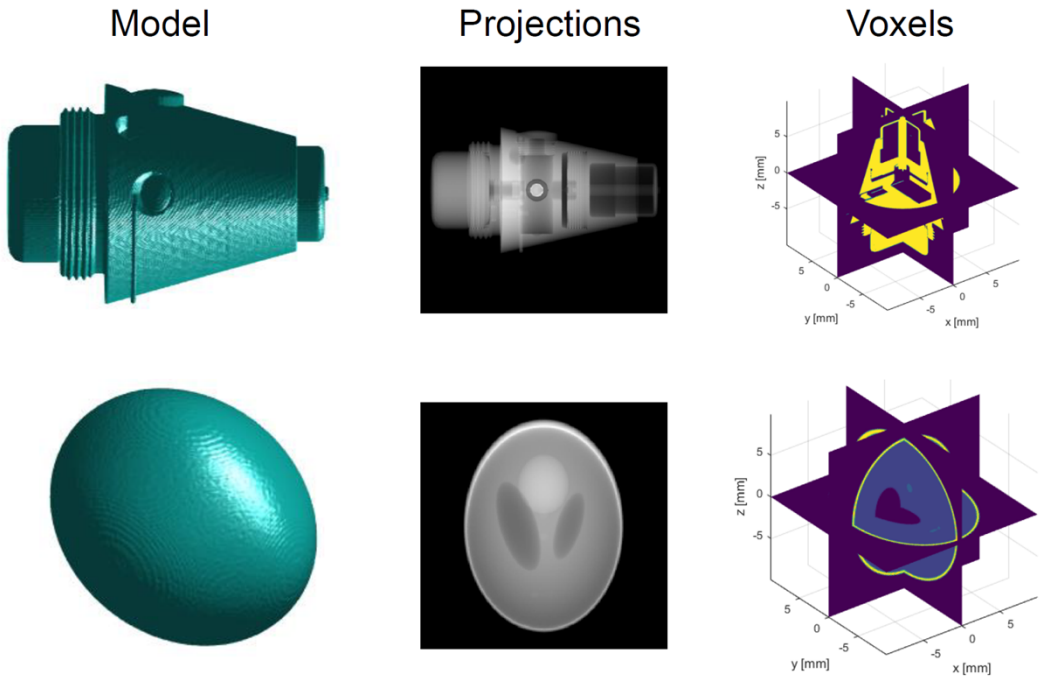

Figure A-12. Sample phantoms used to benchmark reconstruction algorithms. Colums depict the data in different formats: (left) isosurface, (middle) sample projection, and (right) slice plots. Rows depict selected phantoms: (top) mortar shell cap and (bottom) 3D Shepp–Logan phantom.

C.4.4. Reconstruction Module

The reconstruction module applies user-selected reconstruction algorithms to a set of projection data that was recorded by or generated assuming the detector setup provided in the scene.inp file. The primary objective of this module is to invert the measurement model and estimate a field that best matches the recorded projections.

C.4.4.1 Supported Reconstruction Algorithms

1. Filtered Back Projection (FBP) – An analytical method based on the Fourier slice theorem, suitable for scenarios with dense and uniformly sampled projection data.
2. Simultaneous Algebraic Reconstruction Technique (SART) – An algebraic method that iteratively refines the solution by minimizing residuals across all projections simultaneously. The SART algorithm is primarily used as an iterative solver for elements of the Tikohonov and TV regularization solvers.
3. Multiplicative Algebraic Reconstruction Technique (MART) – An alternative algebraic approach that applies multiplicative updates and is particularly effective for reconstructing sparse fields such as the particulate matter generated/transported by a blast wave.
4. Tikhonov Regularization – A regularization technique that penalizes high-frequency deviations in the target field, promoting smooth reconstructions. Absent specific requirements, Tikhonov regularization is a good default choice for tomographic reconstructions.
5. Total Variation (TV) Regularization – A regularization method that preserves edges while smoothing homogeneous regions, making it ideal for fields with sharp boundaries. TV regularization exhibits superior performance to Tikhonov regularization in many cases but requires additional user input. Therefore, it is recommended for users who are experienced with tomographic reconstruction.

In addition to voxel-based methods, the software package includes functionality for reconstruction in the wavelet-domain, which enables an efficient and sparse representation of the target field. For more advanced applications, a separate codebase is provided for performing Neural Implicit Reconstruction Technique (NIRT) reconstructions. This advanced module is designed for experienced users and supports space–time reconstructions of multi-view, ultra-high-speed X-ray radiography time series.

C.4.4.2 Reconstruction Process

For each specified algorithm, the module solves the inverse problem using the corresponding measurement model and projection data. Depending on the user's selection, the module can handle various reconstruction scenarios, from well-conditioned cases with dense data to ill-posed problems where regularization is necessary. Users can configure general parameters, such as the number of iterations and convergence criteria, as well as algorithm-specific parameters, such as regularization weights and neural network architectures. Multiple algorithms can be specified within the same session, enabling comparative analysis of different reconstruction techniques.

C.4.5. Plotting and Analysis Module

The plotting and analysis module is the final step in our workflow. This module generates a series of 3D slice plots for each reconstructed volume, allowing users to visually compare the results from different reconstruction algorithms. The plots display cross-sections of the reconstructed field, which can be interactively adjusted to explore the internal structure of the volume. This functionality

provides a convenient way to assess the quality and accuracy of the reconstructions, facilitating a detailed examination of the spatial characteristics of the target field.

C.5. Synthetic Tests and Validation

Algorithms included in this software package have been validated through a series of synthetic tests designed to assess the algorithms' performance in limited-data reconstruction scenarios. Testing begins with an analysis of the forward model to ensure accurate representation of the measurement process. Subsequently, a variety of inverse test cases are explored, including randomly arranged geometric primitives, a static mortar shell, and the transient density field of an exploding shell. Each scenario is evaluated under different detector configurations—having five, ten, and 20 views—to examine how the algorithms handle sparse and limited-angle data. The algorithms tested include FBP, SART, MART, Tikhonov regularization, and TV regularization. The purpose of these tests is to understand the strengths and limitations of each algorithm in reconstructing complex fields under constrained measurement conditions.

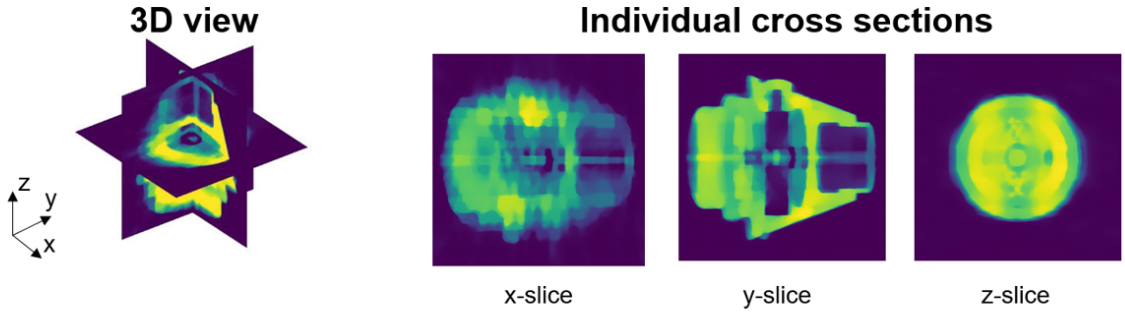

Figure A-13. Slice plots along the x, y, and z planes of a TV reconstruction of the mortar shell cap phantom; plots are generated using the plotting module.

The motivation and scope of the synthetic measurement scenarios considered in this section are detailed below.

1. Forward Model Analysis – A thorough examination of the forward model to ensure that the simulated projection data accurately reflect the true density/extinction coefficient fields, providing a reliable method to baseline reconstruction accuracy.
2. Geometric Primitives – This test case uses a random assortment of overlapping 3D shapes. Projections of these shapes may be analytically computed, and the fields pose a challenge to conventional reconstruction algorithms in the context of limited data.
3. Mortar Shell Cap – A more complex scenario involving a mortar shell cap is used to evaluate the algorithms' performance on realistic geometries that have well-defined boundaries.
4. Exploding Shell – The final test case is designed to assess the NIRT's ability to reconstruct the dynamic, time-resolved density field of an exploding shell, highlighting the performance of the method for 4D reconstructions.

For each scenario, we present the measurement setup, reconstruction results, and a discussion of the performance and limitations of each algorithm. These tests provide a comprehensive evaluation of the software package and demonstrate its capability to handle a variety of challenging reconstruction problems.

C.5.1. Test Case 1: Analysis of Forward Models

This test case analyzes the performance of our simplified forward models using a set of basic geometric shapes. Projections of these shapes have analytical forms, making them ideal for validating the accuracy of different beam geometry models. We compare two models in this section: thin beams and cone beams.

Thin beams: Thin beams comprise an infinitesimally narrow 1D line passing through the volume. This model is computationally simple but less accurate, as it assumes that the response of each pixel is represented solely by the ray intersecting its centroid. Consequently, this model often fails to capture the true interaction between light rays and the volume, particularly around object edges. It particularly struggles to handle blurry radiographs.

Cone beams: A conical geometry, on the other hand, does account for the volumetric nature of detected X-ray beams, providing a more realistic representation of the projection process. The intersection of these beams with the volume can be challenging to compute analytically. However, accurate estimations can be made using Monte Carlo sampling, where points are sampled within each voxel and the fraction of points that lie within the cone is calculated. This approach better captures the interaction between light rays and the volume, leading to more accurate projection images.

Qualitative comparison shows that thin-beam projections produce fewer smooth images compared to cone-beam projections. The thin-beam model introduces aliasing artifacts, particularly along object boundaries, which can lead to errors in the reconstructed volume. While thin beams are computationally efficient, the loss of accuracy can significantly degrade the quality of a reconstruction, especially for detailed or complex geometries and scintillators with large pixels relative to the size of the probe volume. Cone-ray projections, in contrast, exhibit smoother transitions and fewer artifacts, making them a preferable choice for applications where reconstruction fidelity is critical.

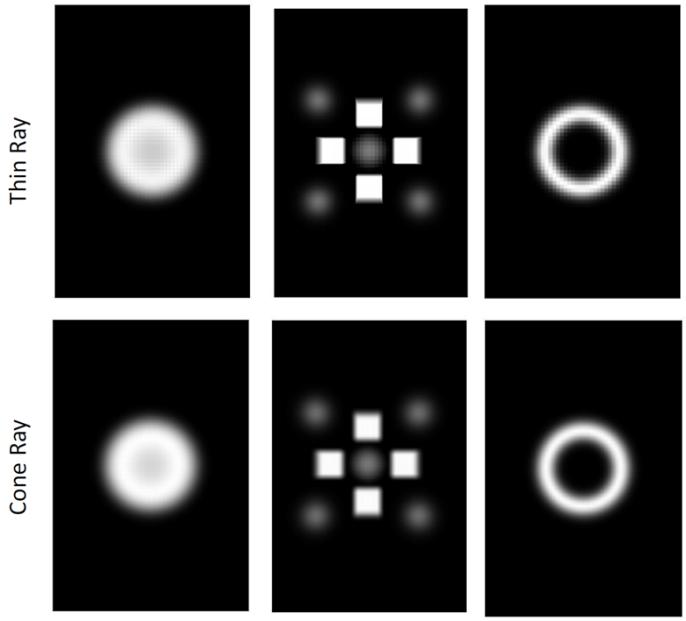

Figure A-14. Projections computed using thin- and cone-ray models.

C.5.2. Test Case 2: Geometric Primitives

Reconstructing fundamental geometric shapes such as spheres, cubes, and cylinders is a classic test for evaluating the accuracy and robustness of a tomographic reconstruction algorithm. These objects have well-defined analytical expressions for both their volumes and projections, making them ideal benchmarks for validating the implementation of reconstruction techniques. In this test, we present 3D reconstructions of a static set of randomly arranged primitive objects. Reconstructions were performed using different sets of viewing angles, including both noise-free and noisy scenarios.

C.5.2.1 Measurement Scenario

The reconstruction of the geometric primitive phantoms is conducted within a domain measuring 200 × 200 × 200 mm3, centered at the origin of the volume. The domain is discretized into 150 × 150 × 150 voxels. Reconstructions are performed using sets of five, ten, and twenty views, corresponding to angular separations of 72°, 36° and 18°, respectively. Identical sensor configurations are used for the detectors. Each one has a resolution of 216 × 216 px with a pixel pitch of 2 mm px.1. The SDD is set to 860 mm, while the SOD is 430 mm. For each configuration, reconstructions are carried out using both noise-free and noisy projection data to assess the impact of measurement noise on the accuracy and quality of the reconstructed volumes.

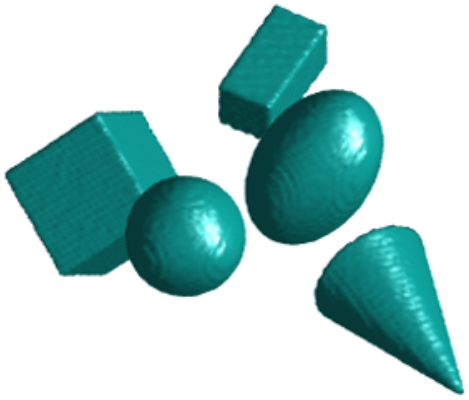

Figure A-15. Field of randomly placed primitive objects.

C.5.2.2 Results

This section presents the reconstruction results for selected algorithms applied to the geometric primitives using the viewing configurations described above.

Figure A-16 depicts cross-sections of the primitives phantom and reconstructions from the five-view case. Reconstructions computed using the FBP algorithm perform poorly in both the noise-free and noisy cases. This is expected, as FBP relies heavily on dense angular sampling to produce accurate results. In contrast, the MART and SART algorithms show relatively stable reconstructions under the same conditions, though artifacts are still visible around the target objects. Both Tikhonov and TV regularization techniques yield smoother reconstructions, with TV outperforming Tikhonov in preserving sharp edges and smooth surfaces. This outcome aligns with the piecewise-constant nature of the field being reconstructed, which is better captured by TV regularization.

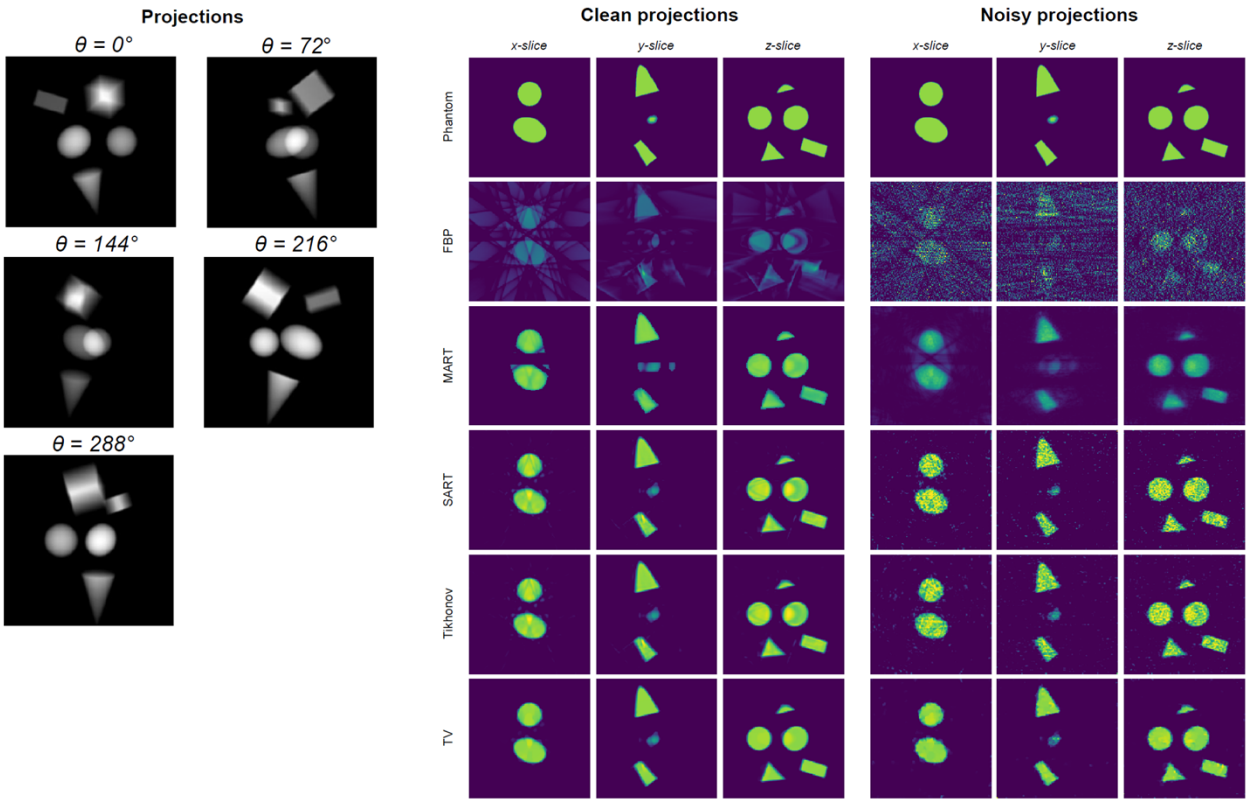

Figure A-16. Five-view projections (left panel) and reconstruction results (right panel) for a geometric primitives phantom.

Results for the ten- and twenty-view radiography data sets are shown in Fig. A-17. As the number of views increases, there is a general improvement in reconstruction accuracy across all algorithms, as expected. Despite the increased number of views, the FBP algorithm still struggles to reconstruct the objects accurately, particularly at boundaries. This is also expected, as the FBP reconstruction usually requires hundreds of views to generate accurate results. MART shows the most significant improvement with additional views, becoming competitive with the other algebraic techniques in the 20-view case. Both SART and TV continue to deliver robust reconstructions, with TV maintaining superior performance in edge preservation and artifact reduction.

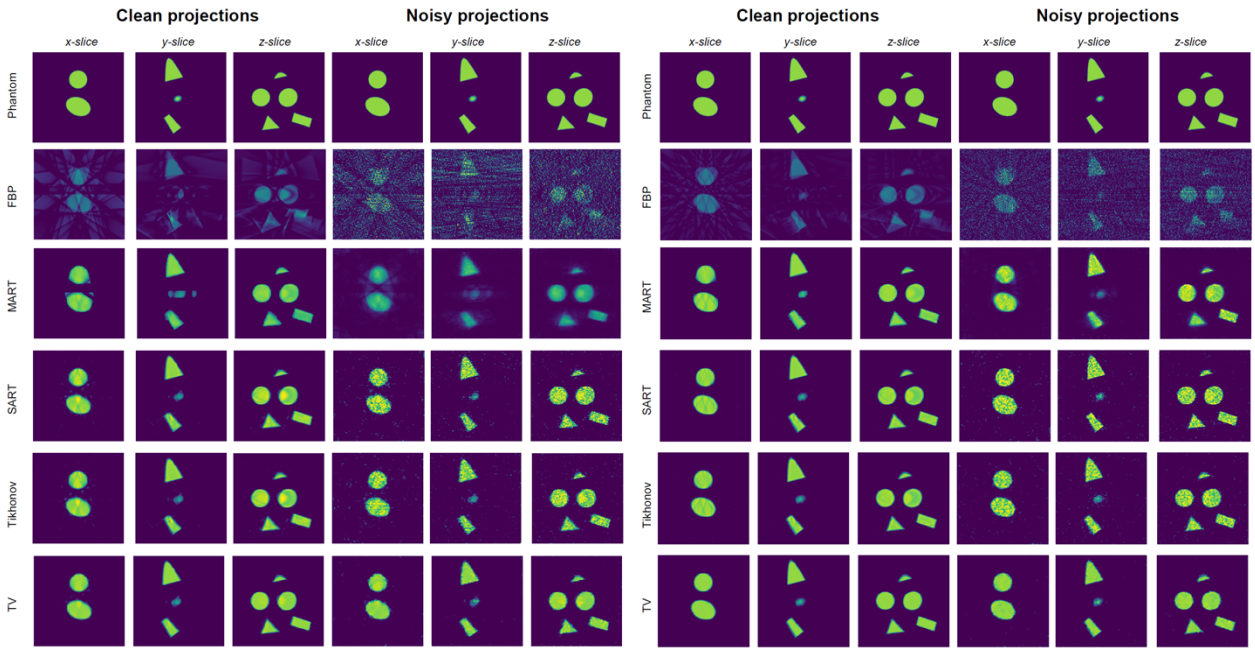

Figure A-17. Ten- and twenty-view reconstruction results for a geometric primitives phantom.

C.5.3. Test Case 3: Mortar Shell Cap

Recovering the internal structure of a complex object poses a significant challenge for tomographic reconstruction, particularly when only a limited number of measurements is available. In the following tests, we adopt a similar protocol as in the previous scenario, but instead of geometric primitives, we use a CAD model of a mortar shell cap. This model consists of numerous intricate, interlocking components, resulting in an extinction coefficient field with broadband spectral content that is difficult to reconstruct. This test enables us to assess the performance of our algorithms on a non-trivial, representative geometry and to identify the limitations of our software in a very limited-data scenario.

C.5.3.1 Measurement Scenario

For this series of tests, we use a reconstruction domain with physical dimensions of $100 \times 100 \times 100$ mm^3. The domain is discretized using a voxel basis with a resolution of 250 voxels along each side. The detectors have a pixel pitch of 2 mm px^{-1}, with a SOD of approximately 430 mm and a SDD of 860 mm. Each sensor has a resolution of 600×600 px. The tests are conducted using five, ten, and twenty viewing angles, with angular separations of 72°, 36° and 18°, respectively, as in the previous case. Once again, reconstructions are performed using both noise-free and noisy projection data to evaluate the algorithm's performance under both measurement conditions.

C.5.3.2 Results

We present the reconstruction results for the mortar shell cap phantom using various numbers of viewing angles. Overall, the performance of the algorithms is consistent with the observations from the geometric primitives test case, but some specific behaviors are noteworthy. The MART algorithm struggles to resolve fine-scale features, such as the internal boundaries of the object, especially when using a limited number of views. A similar trend is observed with the SART algorithm, particularly when noisy projections are used. The primary factor contributing to this limitation is the mortar shell

cap's high-spatial-frequency content, which is insufficiently captured by the projection data. This hinders SART and MART from recovering fine details because these algorithms do not contain any information about spatial regularities of the object. By contrast, the Tikhonov- and TV-based solvers exhibited better stability in the presence of noise, and they effectively recovered the internal structural of the object. The TV algorithm, in particular, excels due to its ability to promote spatial smoothness while preserving sharp gradients, making it well-suited for fields with complex geometries like the mortar shell cap. Selecting the appropriate form of regularization is crucial to capturing these features accurately.

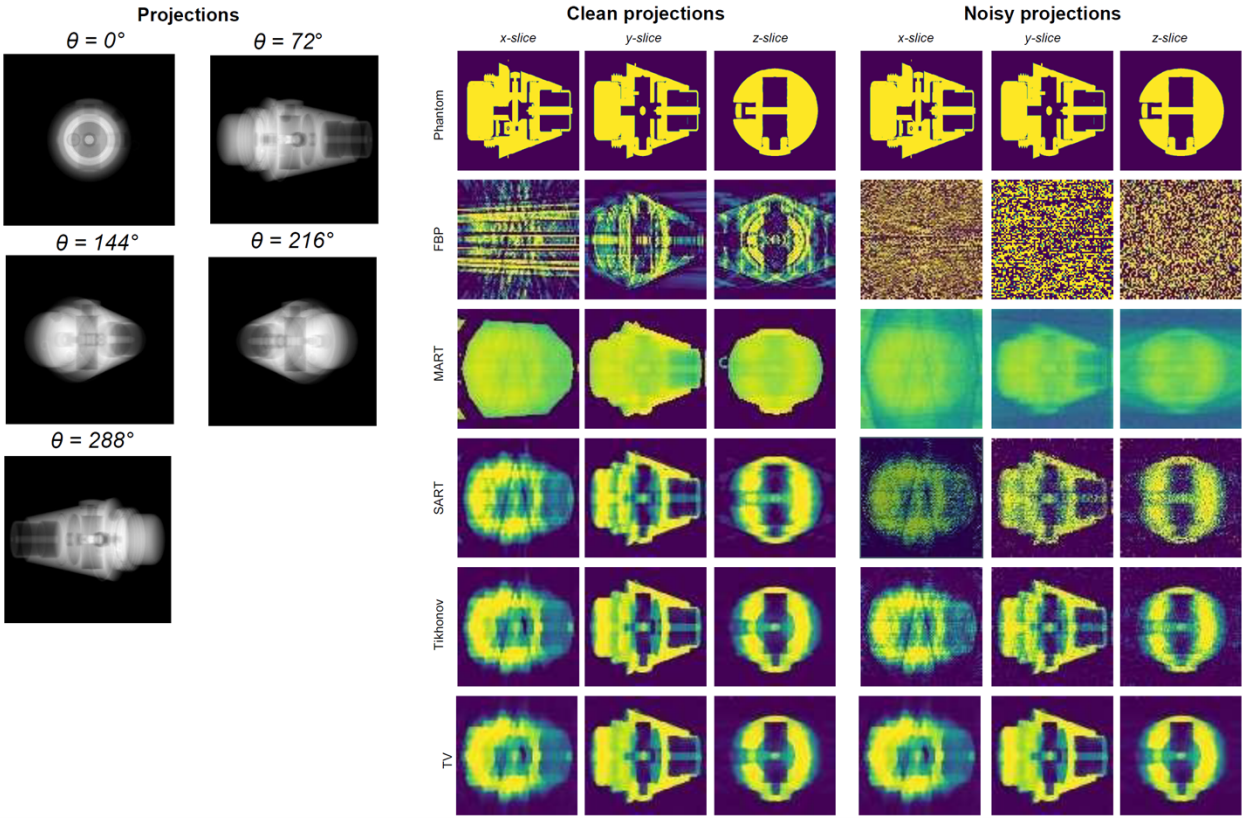

Figure A-18. Five-view reconstruction results for a mortar shell phantom.

In the 10- and 20-view cases, we observe the expected improvement in reconstruction accuracy across the board. However, FBP remains highly unstable in the presence of noise, rendering it unsuitable for most scenarios considered here. While the generic algebraic techniques (MART and SART) show reasonable stability, the TV algorithm consistently outperforms the other methods, providing the most accurate reconstructions even in challenging conditions.

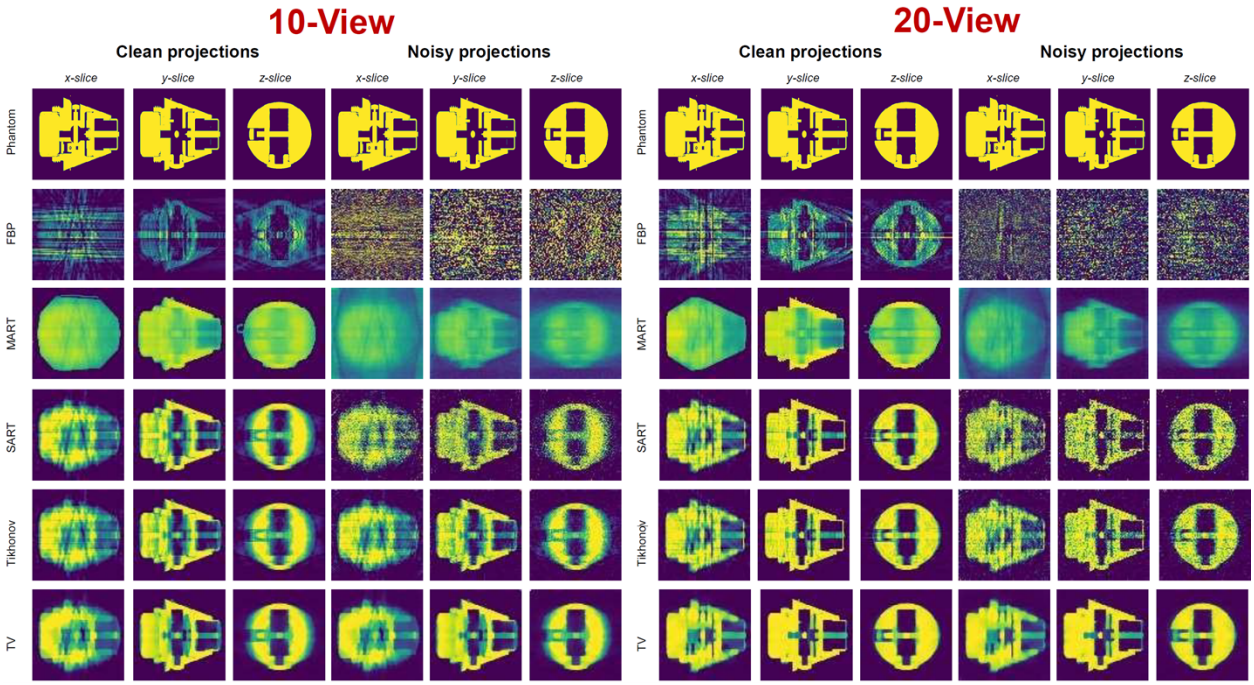

Figure A-19. Ten- and twenty-view reconstruction results for a mortar shell phantom.

C.5.4. Test Case 4: Time Resolved Exploding Shell

This test case evaluates space–time NIRT reconstructions of a dynamic object whose position and structure change over time. Using a set of time-resolved projection data, we aim to reconstruct the object in 4D, capturing both spatial and temporal variations. For this test, we use a CAD model of a detonating spherical shell and perform a full 4D reconstruction over both time and space. The simulation consists of 300 frames in which the shell model undergoes fragmentation and dispersion. The final reconstruction provides valuable insight into the dynamics of the object as it breaks apart, demonstrating the capability of NIRT to capture complex, time-evolving phenomena.

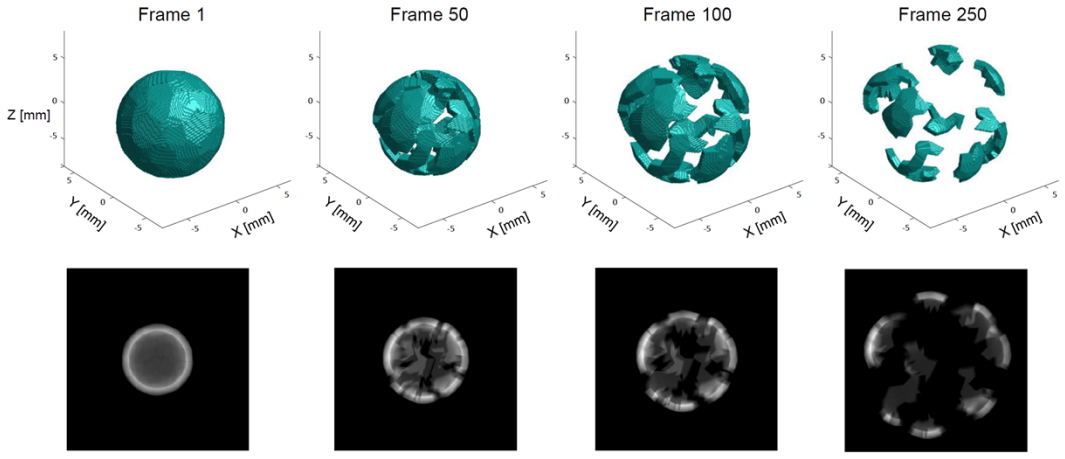

Figure A-20. CAD model of an exploding shell alongside (top row) and high-fidelity projections of the same (bottom row).

C.5.4.1 Measurement Scenario

The NIRT was tested on the exploding shell scenario using a reconstruction domain with dimensions of 100 × 100 × 100 mm^3. The domain was discretized into a grid of 350 × 350 × 350 vx^2. Detectors were configured with a pixel pitch of 0.05 mm px^{-1}, a SOD of 200 mm, and a SDD of 200 mm. Each detector had a resolution of 1000 × 1000 px^2. For this test, reconstructions were performed using only five viewing angles, positioned at 0°, 0°, 66°, 167°, 240°, and 308°. This setup was chosen to evaluate the performance of the NIRT algorithm under constrained conditions, simulating the challenges of high-speed, dynamic imaging scenarios.

C.5.4.2 Results

This section presents results of the spatio-temporal (i.e., 4D) NIRT reconstruction. After training the network on the time-resolved projections, it can be used to output the learned field at arbitrary times within the training interval. Figure A-21 shows isosurfaces of the ground truth field as well as the NIRT reconstructions at selected times throughout the fragmentation process. The large number of closely packed sharp interfaces makes it challenging for the network to accurately reconstruct the individual components from a limited number of views at the outset of fragmentation. During the later stages of the simulation, the fragments become separated and easier to reconstruct. These results demonstrate the NIRT's ability to accurately resolve a dynamic process from limited spatial information.

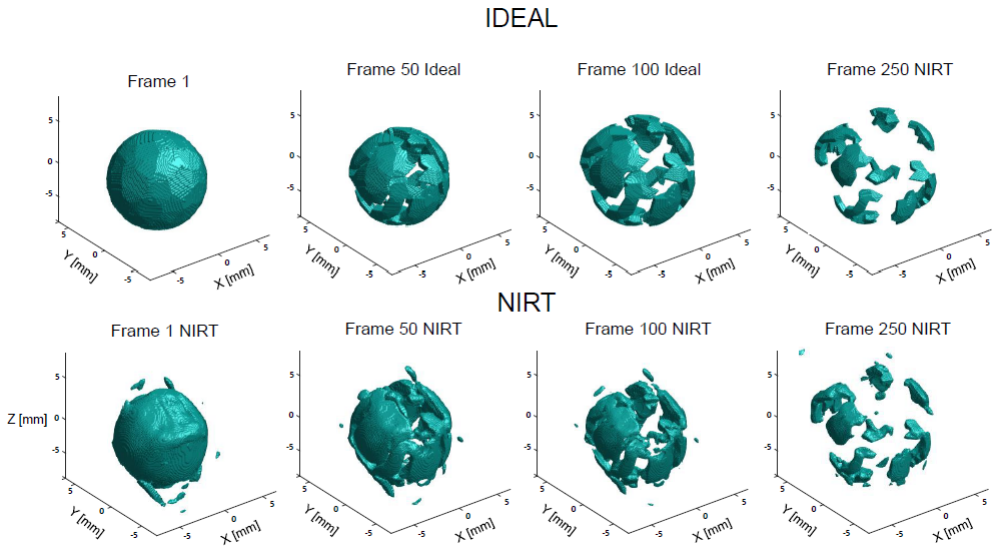

Figure A-21. Isosurfaces of the ground truth exploding shell phantom and the 4D NIRT reconstruction at frames 1, 50, 100 and 250.

C.6. Experimental Test Case

In this section, we present reconstruction results using data collected by our experimental collaborators. The target object is a washer traveling through the reconstruction domain along a defined trajectory. A series of projection images is captured as the washer moves through the domain, resulting in a set of time-resolved projections. This time-series data is then processed using our NIRT code to reconstruct the object in both spatial and temporal dimensions.

C.6.1. Measurement Scenario and Results

The projections of the washer were recorded simultaneously on three separate X-ray detectors positioned around the reconstruction domain. Each detector was located approximately 250 mm from the object and around 400 mm from the source. The temporal reconstructions were performed using a domain size of 13×13×13 mm^3, centered at the origin. Although the reconstruction domain is represented continuously by the neural network, the final reconstructed object is stored in a voxel grid with a resolution of 350 voxels per side. The detectors have a sensor size of 768 × 768 px^2 with a pixel pitch of 0.0926 mm px^{-1}. Figures A-22 and A-23 show the experimental projection data and results, respectively. The NIRT recovered a plausible estimate of the transient density field.

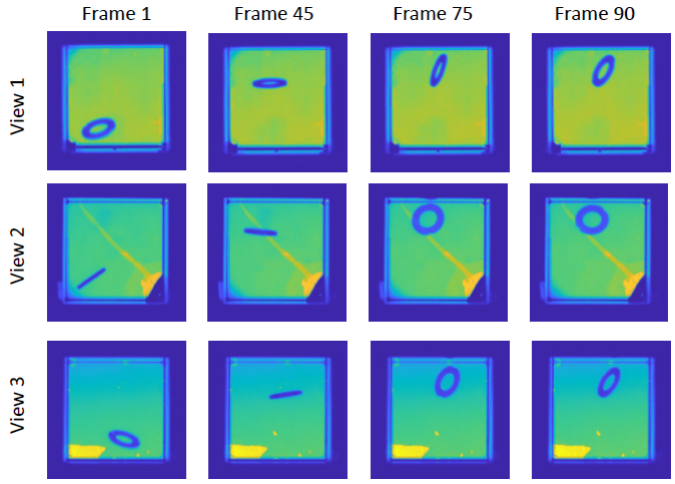

Figure A-22. Raw radiographs from the experiment for each view at frames 1, 45, 75, and 90.

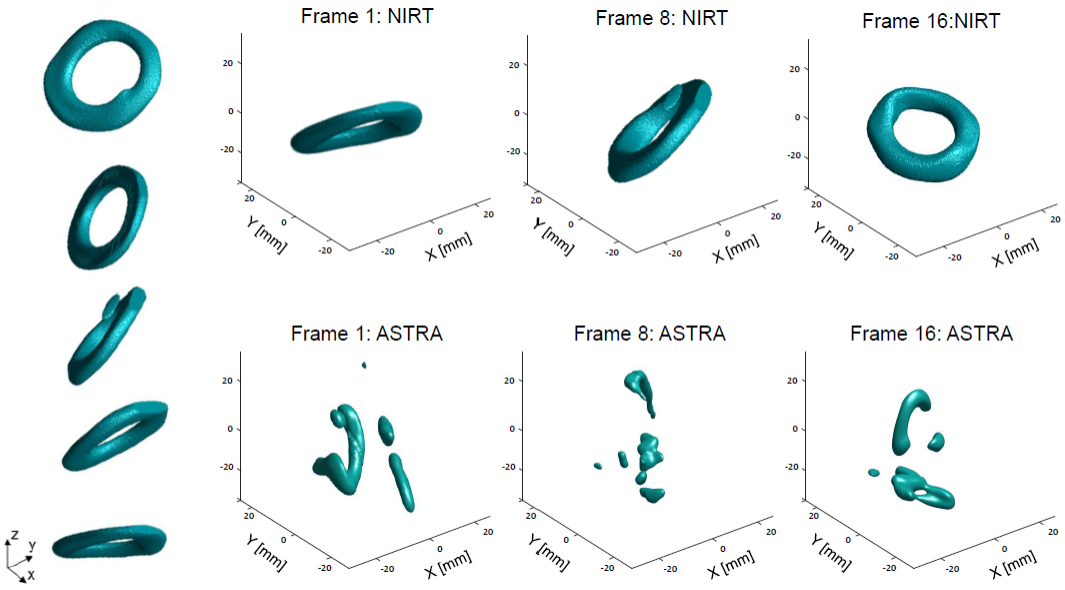

Figure A-23. Sequential frames from a 4D reconstruction of a tumbling washer.

C.7. Conclusions and Future Work

We have developed a comprehensive framework for the reconstruction of both static and dynamic fields and have validated its performance using synthetic and experimental data. The software package includes a suite of algebraic, analytical, and neural-implicit methods, along with tools for validating these algorithms and the capability to upload custom CAD models for reconstruction. For future work, we plan to extend the current neural implicit reconstruction technique to perform joint density estimation and object tracking. This enhancement will enable the simultaneous reconstruction of an object's trajectory and its internal geometry, providing a more complete understanding of dynamic systems.

APPENDIX D. BENCHMARK DATA: EXPERIMENTAL

Jeremy W. James, Rees P. Verleur, and Terrence R. Meyer

School of Mechanical Engineering, Purdue University

D.1. Overview

To validate the accuracy and uncertainty of three-dimensional tomographic reconstructions of high-speed object mass and/or shape dynamics, an X-ray experiment with three sources and three views was established and implemented. Mass conservation experiments used a precision pipette to measure exact volume of a droplet while an acoustic levitator suspends droplets within the field of view. Deformations of the droplet are done by varying the acoustic frequencies and resulting field strength of the acoustic levitator. Dynamic events with solid materials were generated using precise three-dimensional phantoms printed via stereolithography (SLA) with a minimum feature size of 150 μm. Reconstructions could then be verified with computer-aided design (CAD) models at the precision of the printing process. Finally, to evaluate the effects of X-ray imaging dynamic range on the reconstructions, objects with higher aspect ratios and a wider range of optical densities were evaluated, including ringed objects with hollow centers. Solid objects where imaged during falling and rotation to impact temporal dynamics and to help investigate methods for improving reconstruction accuracy in collaboration with the experimental and modeling team members.

D.2. Experimental Setup

A three-view X-ray imaging system was used to capture time sequence images of various dynamic events. Each view was oriented 120 degrees from each other. One A-272 and two RAD-60 Varian rotating anode X-ray tubes were used with a focal spot size of 0.6 mm. The X-rays were projected toward Hamamatsu CsI(Tl) + Al, high-light-output scintillator plates. The scintillators converted the X-ray photons to visible light which were reflected upward by 50 mm right-angle prisms to Phantom cameras coupled with Nikon Nikkor 50 mm lenses. The camera and f-stop combinations used were VEO 640S at f/1.2, MIRO R321S at f/1.2, and MIRO M320S at f/1.4. The distance from the X-ray tube to each object was 205 mm and from each object to each scintillator was 206 mm. The cameras captured 768×768 px^2 images at 1 kHz. Camera exposures and X-ray tube settings varied depending on the object being imaged. Top and side view schematics of the setup are shown in Fig. A-24. Also shown in Fig. A-24 are the positions of each specific camera and X-ray tube. A photograph of the setup shown in Fig. A-25

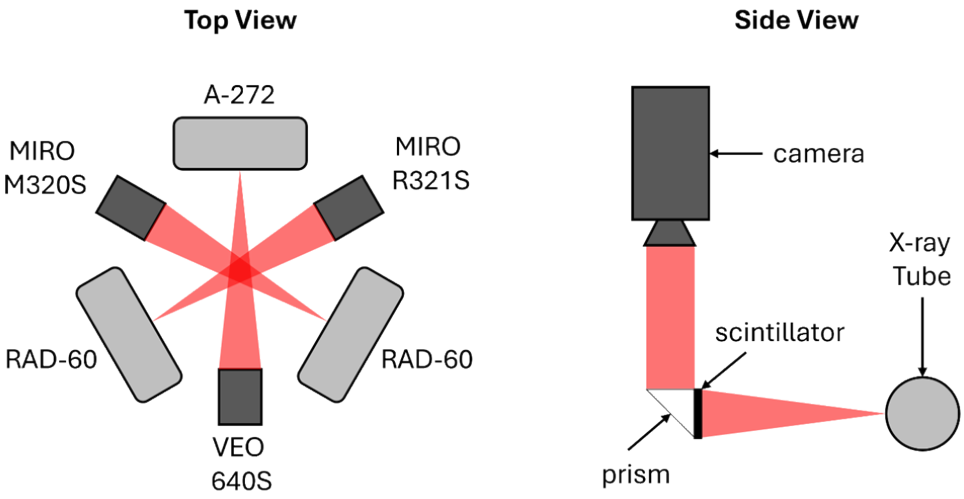

Figure A-24. Top and side view of the three-view X-ray imaging system.

Figure A-25. Image of the X-ray imaging system.

D.2.1. Washer Drop Experiment

A 1 mm thick flat steel washer, shown in Fig. A-26, with an outer diameter of 10 mm and an inner diameter of 5 mm was imaged falling and rotating through the field of view to test the ability of the imaging system to capture highly varying aspect ratios for each viewing direction with a known internal feature. An electromagnet was used to hold the washer about 150 mm above the field of view. A fishing line was used to help promote rotations of the washer when it fell. Notches were filed into the washer to help wrap the fishing line. Another method involved tying a bolt screw further down from the washer to create a higher tension force at the end of its fall and promote more dynamic events. Fig. A-27 shows an assembly of the electromagnet, washer, screw, and fishing line. Timing the X-ray exposures, camera triggers, and release of the electromagnet were accomplished using a delay generator. The X-ray tubes operated with a voltage of 110 kV, a current of 160 mA, and a time duration of 0.2 seconds. The cameras had an exposure time of 400 μs.

Figure A-26. Flat washer used in drop experiments.

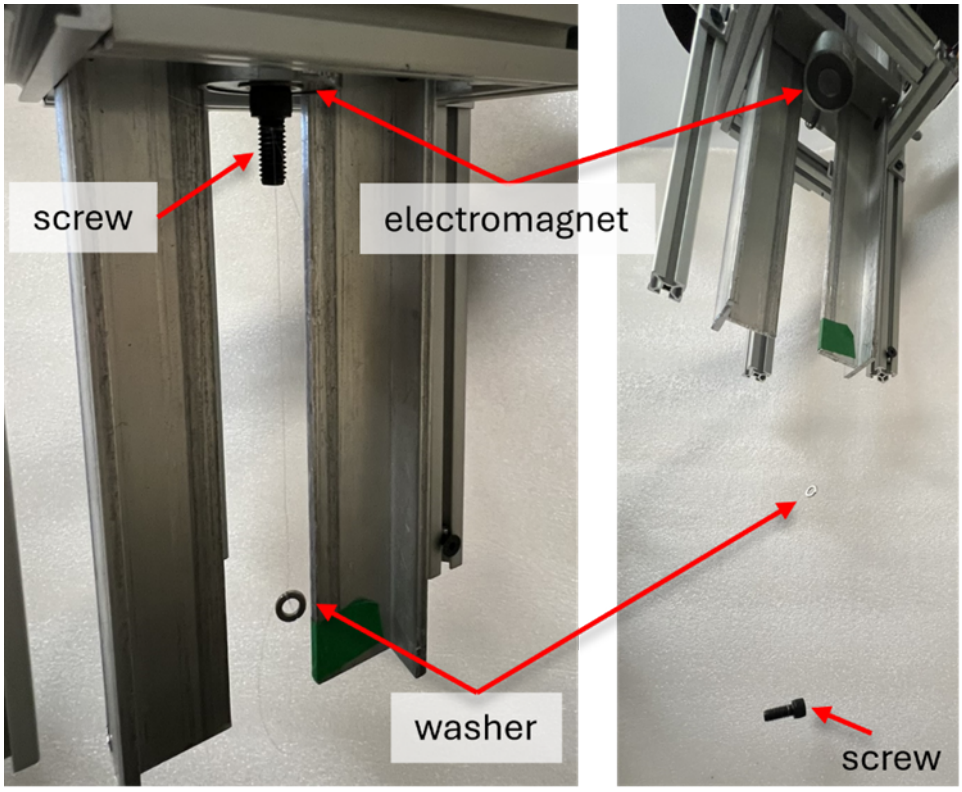

Figure A-27. Method used to drop washer into the field of view of the X-ray system. The washer was held up either by itself or by a screw to the electromagnet and tied together by a fishing line. Right shows the initial positions of all parts and left shows final positions.

D.2.2. 3D Printed Object Drop Experiment

Using stereolithography (SLA), a precise three-dimensional object with a minimum feature size of 150 μm was printed and dropped through the field of view of the X-ray imaging system. This experiment allowed for direct comparison between tomographic reconstructions and the CAD model. Pennsylvania State University's lion was used as a test phantom for this case. The lion was printed to about 25 mm in length and shown in Fig. A-28. Because the lion is not magnetic, the method similar to the washer was used involving a screw to allow for accurate timing when dropping the lion. Again, fishing line was used to wrap around the lion to promote rotations and the screw caused more dynamic events when it reached the bottom of its fall. The X-ray tubes operated with a voltage of 80 kV, current of 100 mA, for 0.5 seconds. The cameras had an exposure time of 990 μs.

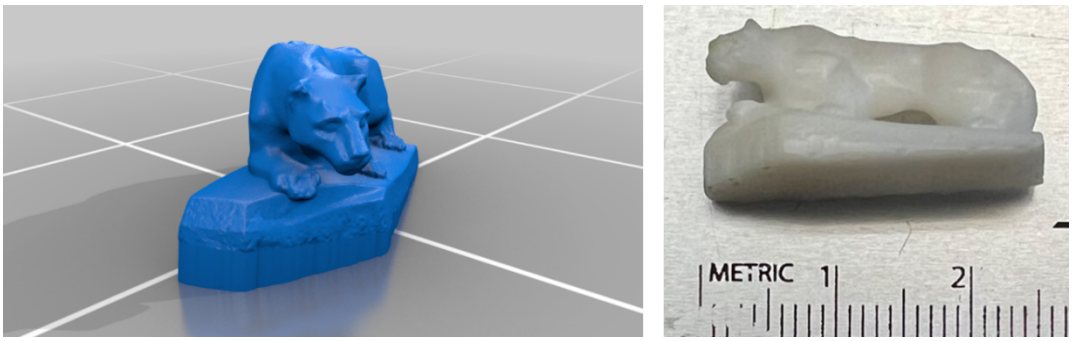

Figure A-28. PSU Lion (left) in CAD model (right) from 3D printing.

D.2.3. Water/Potassium Iodide Droplets in Acoustic Levitator

An acoustic levitator was used to suspend droplets within the field of view of the X-ray system. Fig. A-29 shows the acoustic levitator as well as some droplets suspended within the standing waves. For these experiments, only one droplet was suspended. A micropipette was used to dispense about 20 μL of water to give a 3 mm diameter droplet. This would allow for error analysis when compared to the tomographic reconstructions. The droplets consisted of 15% potassium iodide (KI) to enhance contrast in the droplet for X-ray imaging. The field strength of the levitator caused the droplet to be flattened initially. By powering off the device, the field strength slowly decreases and changed the shape of the droplet shown in the shadowgraphs in Fig. A-30. The X-ray tubes operated at a voltage of 80 kV, a current of 160 mA, and exposed for 0.5 seconds. The camera exposures were 990 μs.

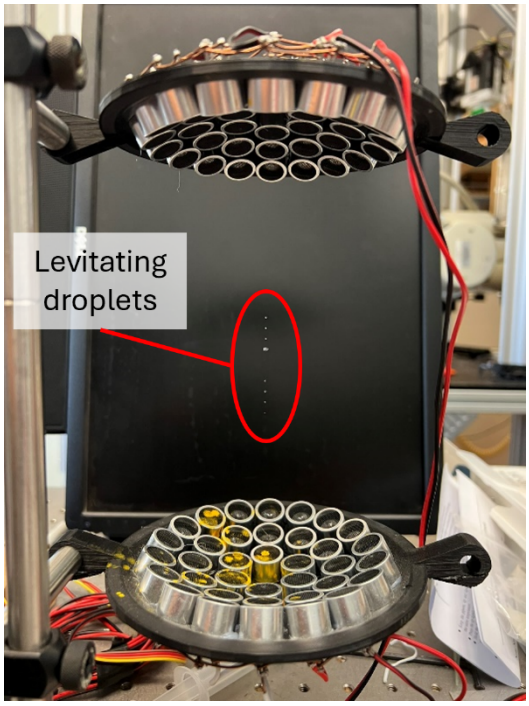

Figure A-29. Suspended droplets in acoustic levitator.

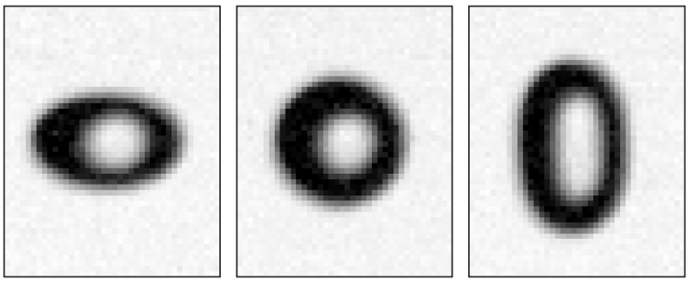

Figure A-30. Shadowgraph of droplet in acoustic levitator. From left to right shows decrease in acoustic field strength and its result on droplet shape.

D.3. Results

The projections of the washer are shown in Fig. A-31. The dynamic nature of the washer also shown. There are several cases in this time sequence where reconstructing the washer may be difficult. Some major examples are t_3 where the washer is viewed from its side for all cameras as well as t_2 where the washer is viewed from the side on one camera.

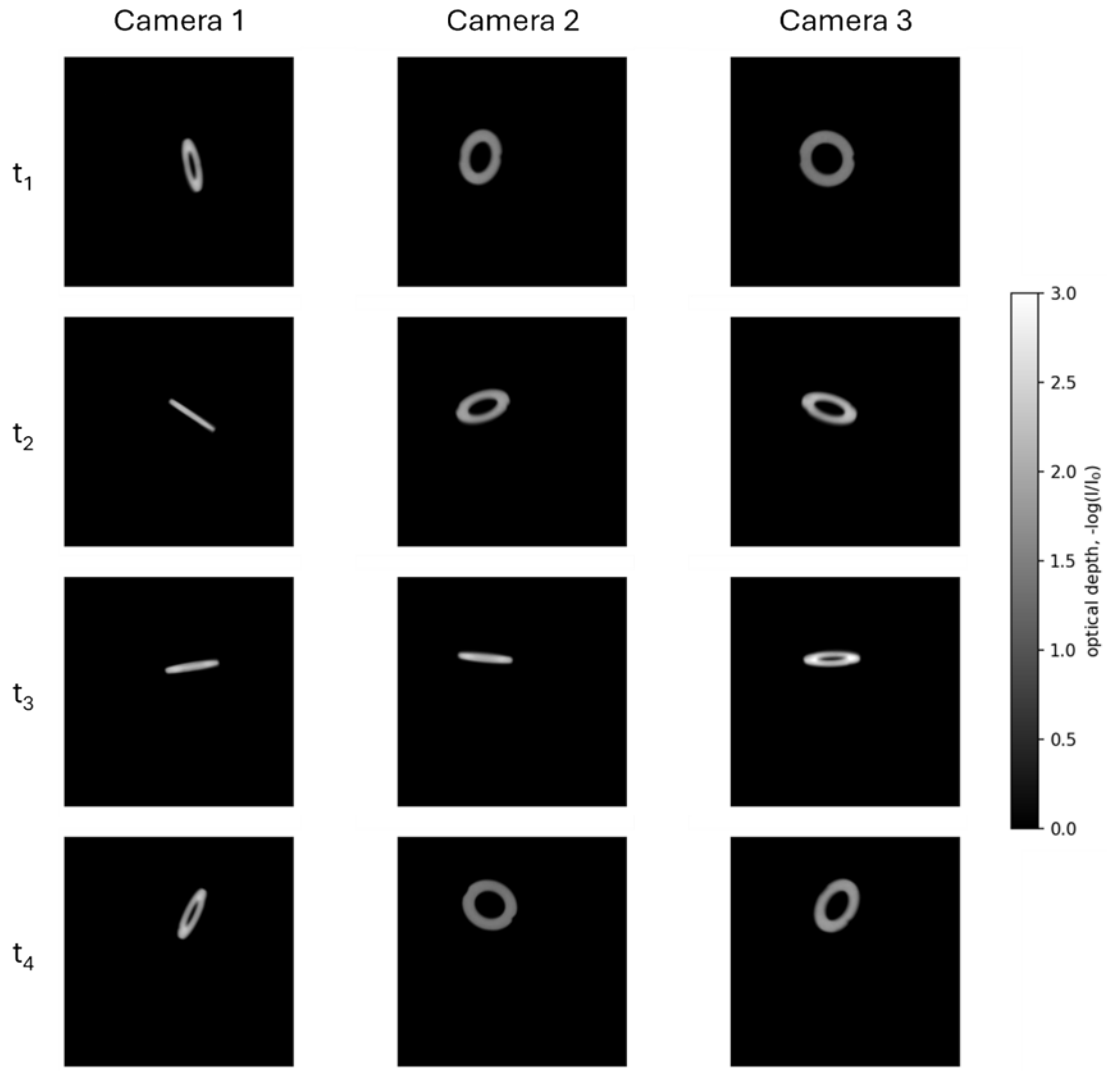

Figure A-31. Projections of washer as it fell and rotated through the field of view.

The first case where the washer is view from the side by all cameras are shown in Fig. A-32. The projection images were masked to have pixel intensity values only where the washer was located. The processed projection images were used to create a volumetric mask for the reconstruction volume. Using MLOS, the projection images created an initial volume of the washer where all three cameras intersected. This initial volume is binarized to create the mask which will be used at each iteration. MLEM reconstructions for the side view case are shown in Fig. A-33. This view proved to be challenging as the annular shape of the washer is mostly lost in the reconstruction. Fig. A-33 shows the result of using a three-view system as the reconstructed object looks triangular. Despite the failure to maintain the shape of the washer the thickness does appear to be transferred to the reconstruction object. One source of potential error is spatial alignment of the entire system. This would cause three intersection points if a single object is being imaged. To minimize this effect, a post was imaged and aligned such that each image was horizontally centered to the center of the post. All views were estimated to be less than 1.5 mm off a global origin in the horizontal plane.

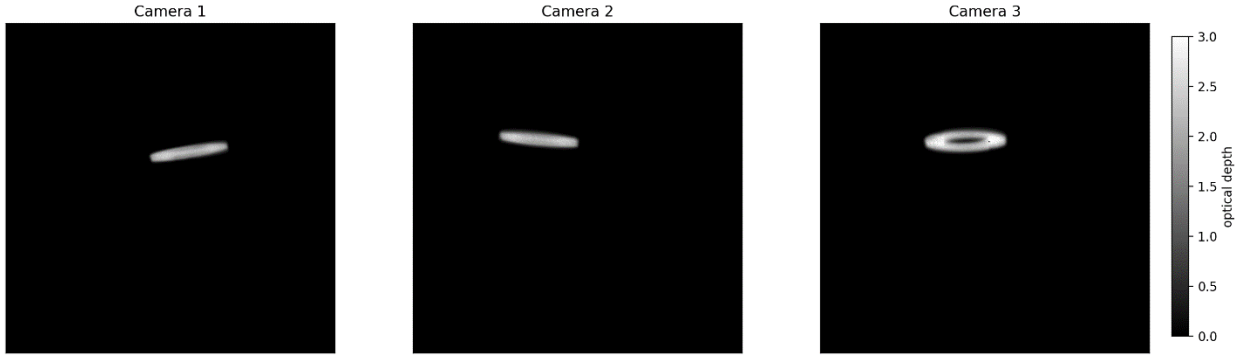

Figure A-32. Optical depth projection images of the washer viewed from the side from the three cameras.

Figure A-33. Reconstruction of washer when viewed from the side for all cameras.

The second case where the washer is view from the side by only one camera are shown in Fig. A-34. This case helps encourage the reconstruction algorithm to promote the annular structure of the washer as shown in Fig. A-35. The reconstruction generated has a structure similar to the washer with artifacts possibly due to the side view image from one of the cameras. Another observation is the reconstructed washer's thickness is much larger. For a washer falling and rotating through the field of view, resulting reconstructions will vary drastically depending on which angle the cameras capture.

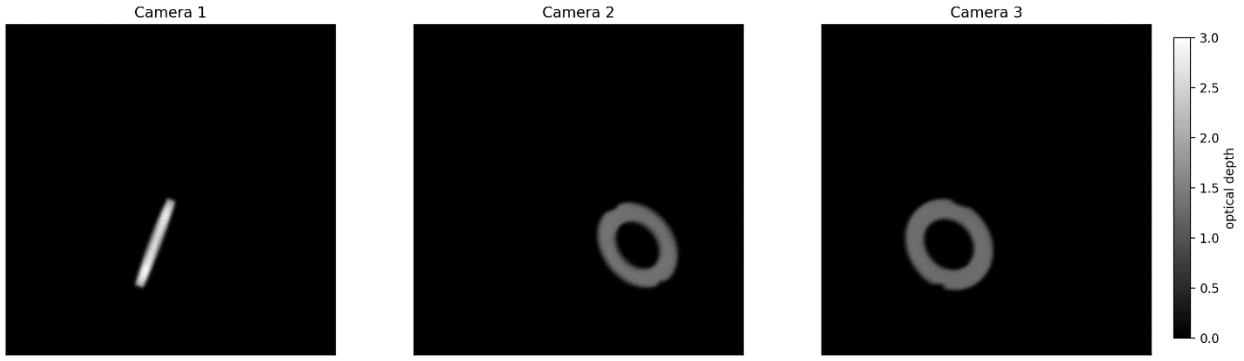

Figure A-34. Optical depth projection images of the washer from the three cameras.

Figure A-35. Reconstruction of washer shown at 3 different angles.

The PSU lion filled the entire field of view of the scintillators. When falling much of the images cut off parts of the 3D printed pieces, so Fig. A-36 only shows a single snapshot of the event. The same image processing was done to the projection images. For reconstruction, the same procedure used for the washer was followed where we initialized the volume using MLEM and produced a volumetric mask for the MLEM algorithm. Fig. A-36 also shows the difference between the washer and 3D printed material X-ray absorptions. A reconstruction of the lion is shown in Fig. A-37. There is resemblance of the lion as the head, body, and tail are shown. The base has many voids, and the lion is shown to be planar instead of having some width when viewed from the top. Possible reasons for this poor reconstruction may be due to the 3D printed lion being much more complex in structure compared to the washer. Low absorbance may also play some role as the dynamic range is not large enough to capture all the features.

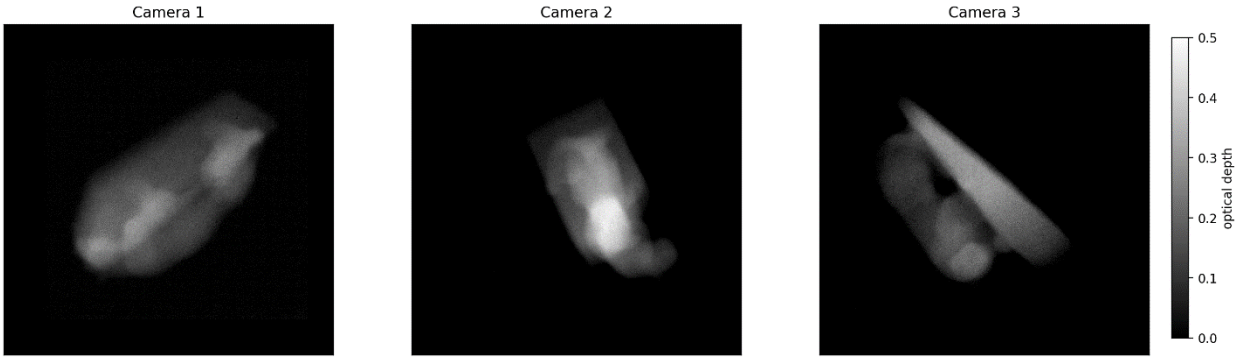

Figure A-36. Optical depth projection images of the 3D PSU Lion from the three cameras.

Figure A-37. PSU Lion reconstructed showing three different angles.

The droplet suspended by the acoustic levitator in the field of the view of the cameras are shown in Fig. A-38. The field strength is at its maximum, causing the droplet to have a more flattened shape than spherical. In this case, the droplet is mostly symmetrical such that the resulting reconstruction in Fig. A-39 has no problems maintaining shape. The side view of the droplet shows the flattening effect.

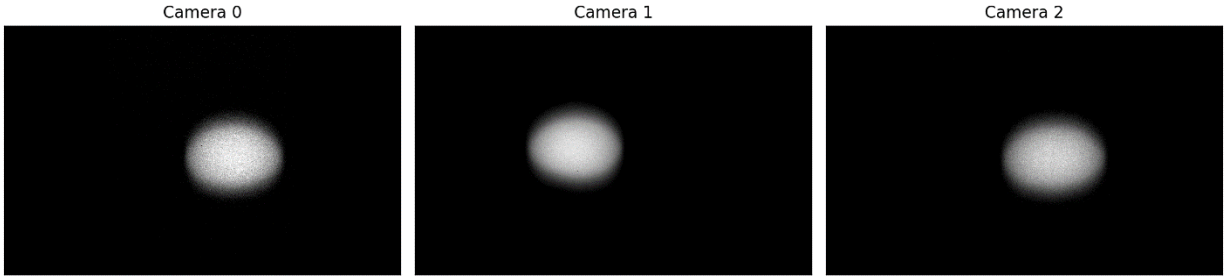

Figure A-38. X-ray projection images of a water/KI droplet for three cameras.

Figure A-39. Reconstruction of suspended droplet. left is an angled view of the droplet and right is the side view showing the flattened shape.

D.4. Conclusion

Imaging dynamic events using a three-view X-ray imaging system was completed to provide a wide range of possible shapes, materials, aspect ratios, and optical depths. These datasets can be used to test tomographic algorithms and perform error and uncertainty analysis. Washers and 3D printed objects can be used to analyze the maintainability of shape through the reconstruction. The levitated droplet experiments can be used to perform mass uncertainty analysis as the droplet changes shape due to acoustic forces. The tomographic reconstructions shown were done with conventional algorithms and further improvements can be made using more advanced methods such as incorporating more informative priors or latest advancements in machine learning tomography. These experiments address the major problems resulting from using limited view X-ray tomography. In future work, improvements can be made to 3D printed objects by adding a contrasting agent to enhance X-ray absorption. Objects with varying structural complexity can also be imaged with rotations through the field of view. For acoustic levitated droplets, frequency modulation can be used to have more control over droplet shapes and further test reconstruction algorithms.

APPENDIX E. BENCHMARK DATA: SIMULATED

E.1. Overview

A physics-based, high-fidelity X-ray imaging software package was developed in MATLAB. This simulation software is designed to be used as a general-purpose tool in experimental design and engineering applications. The software produces realistic scintillator and camera images from X-ray imaging systems and is designed to be highly customizable and capable of simulating a large variety of potential imaging systems. The physical models used within the simulation are described along with the computational methods used to run the simulations. Examples showing that the simulated data closely resembles experimental data are given. Then the advantages and limitations of the software are described along with future plans for improving the overall fidelity of the software.

Software was developed using a combination of MATLAB and Python tools to simulate high-fidelity, physics-based images produced by X-ray imaging systems. This software was created for use in engineering applications where simulations could be used to accelerate experimental studies and decrease overall project uncertainties. The images produced by the software have been compared to experimental data in a variety of situations and found to produce both qualitatively and quantitatively similar data.

X-ray imaging experiments are crucial for performing measurements in environments that are not optically accessible for a variety of reasons that may include the objects of interest being obscured by mechanical components or being located within optically opaque or oversaturated environments such as within the fireball of an energetic event. However, completing X-ray imaging experiments is frequently time consuming, uses many experimental resources, and is expensive. Simulating these experiments and generating X-ray images that would be produced by these experiments can be advantageous for several reasons.

Since these experiments may be resource intensive or involve tests where only a limited number of shots may be completed, determining the optimal experimental configuration prior to a test is highly valuable. Simulating the experiment beforehand provides a means of quickly and cheaply testing multiple such configurations and finding the best geometric setup for collecting the X-ray imaging data and ensuring that the components being imaged are both visible in the experimental images and yielding useful data. This ensures that when the experiment is completed, the data provides valuable, high-fidelity information.

Additionally, experimental setups often require making design decisions related to manufacturing or assembling parts that will be used in the experiment. If the components being manufactured will be imaged by the X-ray system, running a simulation of the experiment allows the design parameters to be optimized prior to manufacture. This decreases the overall risk in performing experiments since the simulations provide confirmation that the experiment will yield the most informative data.

Once the experimental data is collected, understanding the uncertainties in the data is often necessary. However, quantifying these uncertainties can be difficult, especially in cases of single-shot experiments being completed. However, performing simulations of the experiment provides a means of estimating these experimental uncertainties. The sources of noise in the experiment may be quantified, by analyzing background image noise for example, and replicated in the simulation. Then multiple simulations can be run where the solution is known, and measurements can be collected from the simulated images. The uncertainty in the experiment may then be estimated by comparing the measured image data from the simulations with the known true solution.

Processing the data from an experiment often introduces errors from the nature of the processing algorithms themselves. For example, calculating tomographic reconstructions requires making a priori assumptions about the nature of the input data. If these assumptions are not valid, then the reconstruction process inevitably introduce errors. Even if the assumptions are correct, due to the under-determined nature of the data, the reconstruction process will still introduce errors. However, the true values in experimental data are never known, so identifying these errors can be difficult or impossible when only experimental data is considered. Using these same processing algorithms on simulated data where a true solution is known allows these errors to be quantified. This not only allows uncertainty in measurements to be quantified, but also provides a means of improving these processing algorithms and comparing between different algorithms.

Much of the development and advancement of X-ray imaging has been traditionally motivated by the medical field. X-ray imaging provides many advantages for medical diagnostic measurements. For this reason, X-ray imaging simulation software has been primarily developed for the medical community. While this software provides many useful advantages within this community, it has less utility as a tool for X-ray imaging in a wider, general context.

For one reason, the medical community tends to use lower energy X-rays for imaging. The human body is relatively transparent to X-rays compared to many engineering materials such as metals, therefore lower energies provide higher contrast in medical imaging. Additionally, lower energies are less likely to produce negative effects due to ionizing radiation that are harmful to the patient. Therefore, most medical X-ray imaging software is designed to function at low X-ray energy levels that would have limited use in engineering contexts.

Additionally, medical X-ray image simulation software is often designed to model specific X-ray machines. This is useful for analyzing the medical data collected from these machines, however the inability to model more generalized systems limits the utility of this software.

Other X-ray imaging simulation software that is more generalized, often is developed with specific end-use goals that are not useful to engineering applications. For example, many rendering software tools have 'X-ray' modes that create images of transparent objects. However, the images created with this software tend to be non-physical and are primarily used as a visualization tool. Some physics-based simulation software has been developed within the academic community, but this software is often designed to model very specific contexts and has not been optimized to be computationally efficient for general simulation purposes.

The software described here was built to be a generalizable, physics-based X-ray imaging system that can be used in engineering applications. The software is designed to be computationally efficient and is capable of generating realistic images of highly complex designs in reasonable time frames. In most cases, the software performs well and produces high-fidelity simulated images. The experimental situations in which the software will produce lower fidelity images are clearly described and delineated so that the users may determine if the software is applicable to specific engineering needs.

E.2. Methods

The X-ray simulation software calculates scintillator and camera images by modelling X-ray photons traveling from a source, through objects, and then producing an image on a scintillator screen that is imaged by a camera. The software breaks the modelling down into sections that model physical objects and phenomena. Since the software is physics-based and intended to produce realistic output images, the modelling calculations are often relatively computationally complex. So, in addition to

discussing the physical processes that are modelled, the algorithms that are used to efficiently perform the calculations are also discussed.

E.2.1. Overview

The X-ray simulation software is primarily based upon ray tracing light rays in X-ray and visible light wavelengths. Scintillator and camera sensor images are generating by propagating individual X-ray photons or bundles of photons with identical energy levels. The light rays are defined by a multi-dimensional vector that includes their origin coordinates, the propagation direction, the light ray energy level or equivalent wavelength, and the intensity of the light ray if it is being modelled as a bundle.

Computationally, the simulation process involves several steps. The light rays are first emitted as X-rays from the generator source and pass through the specified number of X-ray filters. Next, the light rays propagate through free space, intersect, and are attenuated by the defined imaging objects. The light rays then intersect the detector filters and the scintillator, producing the first image. This image is then used to generate a second set of light rays in the visible light or near visible light frequency range that propagate through free space to the camera system. These light rays intersect and are refracted by the camera lens before being focused onto the camera sensor where the second image is produced. Finally, the simulation saves both the scintillator and camera sensor images to memory. Figure A-40 shows a schematic of the components of the X-ray simulation.

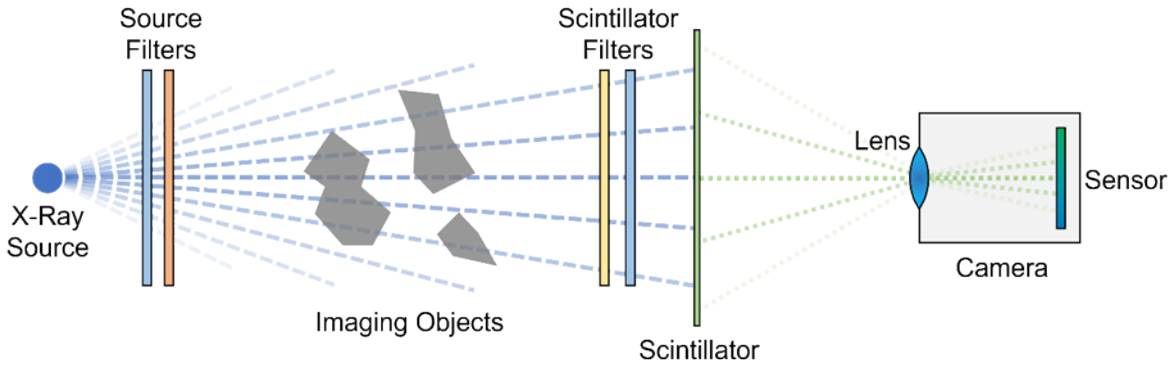

Figure A-40. This diagram shows the components of the X-ray simulation and roughly represents the flow of the simulation data from left to right. Any component may exist multiple times within a simulation, however only the filters and imaging objects are shown in multiplicity here for ease of visualization.

The software is designed to allow complex experimental setups to be modelled. Any number of X-ray sources, filters, imaging objects, scintillators, and cameras can be included in the simulation model. The positions and orientations of the system components can all be independently defined. This allows diverse systems to be modelled including flash X-ray imaging where there are multiple X-ray sources directed at a single scintillator screen or tomographic X-ray imaging where there are multiple in-line sources and scintillators for example. Additionally, dynamic scenes can be simulated by passing multiple frames of imaging objects into the simulation which will yield a sequence of image frames as output.

Since X-ray photons created by medical and flash X-ray sources are polychromatic covering a wide range of energy levels with a very non-linear spectrum, the SpekPy modelling software was used to generate the X-ray photon spectrums [1]. This software specifically models the physical process of

X-ray photon creation given an incoming electron energy level and the anode material generating the X-ray photons and yields a photon fluence spectrum. This photon fluence spectrum is then used to create the distribution of X-ray photon energy levels used in the simulation. The spectrums can be calculated by either loading a pre-calculated spectrum table or by directly calling the SpekPy software. Calling SpekPy from MATLAB requires installing Python and SpekPy and linking MATLAB to the Python instance that SpekPy was installed within.

E.2.2. Physical Modelling

The simulation breaks the modelling process down into several components that are based upon physical objects or phenomena that would exist in an experimental X-ray imaging setup. The physical objects that are modelled include the X-ray source, X-ray filters, imaging objects, the scintillator, and the camera imaging the scintillator. The phenomena that are modelled include the X-ray photon energy spectrum, absorption or attenuation, fluorescence of the scintillator screen, and the imaging process of the camera. The fidelity of each of these models is discussed later.

E.2.2.1 Absorption and Attenuation

Light rays are modelled either as individual photons or as bundles of photons that all have the same origin, direction, and energy level. When the light rays that are modelled as individual photons interact with materials, the probability that the photon would be absorbed by the material is compared to a random variable to determine whether the photon passes through the material or is absorbed. If the photon is absorbed, then the corresponding light ray is removed from subsequent calculations and does not contribute to the scintillator image calculation.

Alternatively, when the bundles of light rays are modelled as passing through a material, the amount of attenuation is calculated from the Beer-Lambert law and applied to the overall intensity of the light rays. The light rays are modelled as all having initial intensities equal to one with each subsequent material that they pass through decreasing this value by some extent. Then all light rays that intersect the scintillator screen contribute to the overall image.

Modelling individual photons is more physically realistic, however, in typical experimental systems the number of X-ray photons that interact with the imaging system is beyond the means to computationally model with a reasonable amount of time and resources. Therefore, the simulations will generally under sample the number of physical light rays. In practice, any imaging system will be subject to a noise floor due to background sensor noise and discretization error. Except in extremely low-light conditions, the contribution of individual photons will be below this noise level. In these cases, the bundle model will generate realistic images. Alternatively, if the contributions of individual photons become important, then the single photon model can be used to generate the shot noise that would appear in images in this case. Computationally, the absorption model is faster than the attenuation model when the same number of photons are modelled.

E.2.2.2 Meshes Defining the Imaging Objects

The objects that are imaged by the simulation are represented by a three-dimensional triangulation mesh. Specifically, any object loaded into the simulation must be represented by two arrays: an array storing three-dimensional vertex coordinates and an array giving sets of three indices into the vertex coordinate array representing the three vertices of each triangle. These two arrays are stored within a mesh data structure that contains the data representing all meshed objects. This system allows any complex geometry with any number of objects of different materials to be represented.

The X-ray absorption is calculated by intersecting the light rays representing the individual X-ray photons or bundles of photons with the triangles within the mesh data structure. The distance between the points of intersection then gives the distance over which the X-rays are absorbed. To properly calculate this distance, any possible light ray must be able to both enter and exit the imaging object. For this reason, the triangulation mesh must represent a manifold surface, i.e., the mesh represents a surface with a well-defined interior and exterior and the surface contains no open boundaries.

In practice, a mesh with these properties can be generated by exporting a CAD file representing the imaging object as an STL filetype which can then be imported into MATLAB. Therefore, the mesh objects must be saved as STL files for loading into the simulation software.

E.2.2.3 X-ray Source

The X-ray source is modelled as a planar region of non-zero area from which the light rays emanate. The light rays are assigned random origin coordinates within this area sampled from a super-Gaussian distribution that is either rectangular or ellipsoidal in shape. This allows a variety of different source geometries to be modelled. A cone-beam angular distribution is randomly assigned to the light rays. This distribution may be constant or super-Gaussian as well and the angle of the cone-beam is user defined. Finally, a random energy level is assigned to the light rays that is sampled from the X-ray energy spectrum generated from the SpekPy library.

E.2.2.4 Filters

Filters are defined at two locations within the overall imaging system: between the X-ray source and the imaging objects and between the imaging objects and the scintillators. While the order that filters are applied does not alter the resultant images and the filters could be grouped together, realistically in a physical setup, there is a small amount of filtration applied both by the source and the scintillators. Commonly a thin beryllium or aluminum window is built into X-ray tubes. Additionally, the scintillation material is printed onto a rigid backing that is frequently made of aluminum. By separating the filtration into both source and scintillator filters, system components corresponding to physical parts can be built within the system and easily saved and used with different simulations.

Computationally, filtering is applied by passing all light rays through a material with constant thickness and calculating the light rays that are either absorbed or the amount that the light rays are attenuated if they are being simulated as photon bundles. Since this process is uniformly applied to all light rays, it is computationally efficient. This contrasts with the process of calculating the effects of the imaging objects on the light rays.

E.2.2.5 Imaging Objects

Modelling the process by which the light rays pass through the imaging objects is relatively computationally complex due to the variety of ways in which the light rays may interact with the objects. The light rays may not intersect any objects and simply pass by without any absorption or attenuation. Or the light rays may intersect an object only one time before propagating to the scintillator. Alternatively, the light rays may pass through multiple objects comprised of multiple different materials and experience absorption or attenuation several times.

To account for all of these possibilities, each light ray is tested against every triangle that defines the object meshes to see if an intersection occurs. If the meshes are properly constructed, then there should always be an even number of intersections (including zero intersections). The first intersection

that occurs will be the light ray entering a material while the second intersection will be the light ray exiting the material. The location of these intersection points is calculated and the Euclidean distance between them is used to calculate the amount of absorption or attenuation. This process is repeated for all intersection pairs.

E.2.2.6 Scintillators

Once the light rays have passed through the imaging objects and the scintillator filtration, the scintillator image is produced by summing the contributions of the intersecting light rays. There are two options used to generate the scintillator image. First, all light rays that intersect the scintillator are added together to create the resultant image. And second, the scintillator material and thickness are specifically described by the user input and the rate of absorption by the screen is calculated.

The first option is less physically realistic but will produce images that converge more quickly. The second option models the physical process of image generation in an experiment, but since some light rays will pass through the screen, the resultant image will converge more slowly. In practice, many scintillator screens use materials consisting of elements with high atomic numbers that absorb a significant number of the incoming X-ray photons, so for many common scintillator materials, the two methods will produce similar results.

If a camera is modelled within the simulation, the scintillator screen then re-emits light rays in the visible light energy range that then intersect the camera lens. The flux density of the emitted light rays is proportional to the image intensity that is calculated from the incoming X-rays. The direction of the light rays is sampled from a Lambertian cosine distribution to ensure that the radiance remains constant regardless of the camera viewing angle. Rejection sampling is used to only generate light rays that will intersect the camera lens to ensure computational efficiency when calculating the camera image.

E.2.2.7 Cameras

The cameras imaging the scintillator screens are represented by a combination of an optically thin lens and an imaging sensor placed some distance behind the lens. The lens is defined by its focal length and f-number, while the sensor is defined by multiple parameters including its resolution, pixel size, quantum efficiency, and spectral response. The lens is assumed to be oriented parallel to the image sensor since this is common for most camera systems and simplifies the sensor image calculation.

The light rays emitted by the scintillator screen are represented by three-dimensional direction vectors with three-dimensional origin coordinates in the lab reference frame. However, these light rays are converted to a camera reference frame that is calculated from the user defined orientation of the camera sensor. In the camera reference frame, the light rays may be represented by simpler two-dimensional direction vectors with two-dimensional origin coordinates. This simpler form allows ray-matrix operations to be used to calculate the refraction of the light rays by the camera lens and the image formation on the camera sensor, thus increasing computational efficiency.

The lens may be either focused at a user defined distance or automatically focused on the center of the scintillator screen. The defined aperture of the lens is used to calculate the depth-of-focus of the imaging system, that ratio of incoming light to all light emitted by the scintillator screen, and the effects from diffraction on the overall image quality. Smaller apertures, corresponding to larger f-numbers, will result in larger depth-of-focus, while minimizing the amount of incoming light and decreasing the image quality due to diffraction. Typically, in an experiment, the lens aperture will be opened to as large a diameter as is possible.

The sensor resolution and pixel size will determine the overall size and magnification of the output image. The quantum efficiency and spectral responses are then used to calculate the intensity of the image. The diffraction effects from the lens aperture are applied to the final output image by convolving a Fraunhofer diffraction pattern with the calculated image.

E.2.3. Computational Methods

Creating X-ray scintillator and camera image data with a high-fidelity simulation is computationally expensive. A large number of both X-ray and visible light wavelength light rays need to be simulated and either attenuated in the case of X-rays or refracted in the case of the visible light. Additionally, the computational load exponentially increases with the complexing of the imaging objects. Therefore, multiple techniques were used to increase the computational efficiency of the ray-tracing simulations.

E.2.3.1 Boundary Volume Hierarchy

Calculation of the light ray-object intersections will scale with the product of the number of light rays and the number of triangles within the object mesh. Typically, around one billion light rays will be simulated to create converged images. And complex CAD designs may have millions of triangular facets in the exported mesh. So direct calculation of these intersections quickly becomes computationally infeasible. Therefore, all but the simplest meshes are stored in a data structure known as a boundary volume hierarchy that allows for computationally efficient intersection calculations.

In a boundary volume hierarchy, the mesh objects are broken down into the constituent triangular faces which are stored in a tree structure based upon the geometric location of the triangles. Each branch of the tree represents a bounding volume that contains all triangles within that volume, with the top level of the tree consisting of a volume that contains the entire mesh of all triangles. Different levels of the tree correspond to nested boundary volumes that contain fewer and fewer triangles further down the branches of the tree. Each branch of the tree is broken down into two children branches corresponding to smaller boundary volumes.

Light ray-object intersections are then efficiently calculated by testing whether light rays intersect each of the two branch volumes. If a light ray does not intersect a boundary volume, then subsequent children branches do not need to be searched for potential light ray intersections. This allows light rays to only be tested against a small number of triangles that have a high probability of intersection. By searching the tree in this way for potential intersections, the number of intersection tests is reduced to both logarithmic time and memory requirements compared to a direct comparison.

Since MATLAB relies on vector operations for efficient calculation, creating a tree with only a single triangle assigned to each leaf node is not efficient since this would require iteratively traversing a large number of branches individually. Therefore, the number of tree branches is capped so that there are typically hundreds to several thousand triangles assigned to each leaf of the tree that are all tested for possible intersection simultaneously.

E.2.3.2 Light Ray Triangle Intersections

Since a very large number of intersection tests between light rays and triangles must be completed to calculate the X-ray absorption, it was necessary to use a highly efficient intersection test algorithm. The format of the computational data where the light rays are represented by vectors that include a direction and origin and the triangles are represented by a set of three three-dimensional coordinates lends itself well to using the algorithm developed by Möller and Trumbore [2].

This algorithm finds the intersection point of the light rays with the plane in which the triangle lies by calculating the intersection point in the barycentric coordinates of the triangle. This method never requires calculating the equation defining the plane. Additionally, the values of the barycentric coordinates directly yield whether an intersection has occurred. If an intersection does occur, the Cartesian coordinates of the intersection point may then be quickly calculated from the barycentric coordinates.

The algorithm was implemented in a vectorized format so that the calculations could be performed for a large number of light rays being tested against a large number of triangles simultaneously. This ensures a high degree of efficiency when performing the calculations in MATLAB.

E.2.3.3 Inverse Cumulative Distribution Functions

To perform a physically realistic Monte-Carlo simulation of an X-ray imaging system, random samples need to be generated from several non-uniform distribution functions including the one-dimensional X-ray energy spectrum and the two-dimensional super-Gaussian cone beam angular distribution. Samples from these distributions could be generated using a rejection sampling scheme where a uniform random sampling is calculated over the domain of the functions and only samples that conform to the distribution are saved. However, this will generally be inefficient. For a tungsten anode X-ray energy spectrum at 100 keV, only about 4% of generated points will be valid using rejection sampling. Therefore, the inverse cumulative distribution functions or quantile functions were used to efficiently calculate samples from these distributions.

Given a probability density function f_X of a random variable X, the cumulative distribution function is given by the integral

$$F_X(x) = \int_{-\infty}^{x} f_X(t)dt$$

which can be numerically calculated using an adaptive quadrature technique. Since the probability density function is always positive, the cumulative distribution function F_X must also monotonically increase, implying that an inverse cumulative distribution function F_X^{-1} exists. This inverse function may be calculated from the F_X using numerical interpolation. If f_X is properly scaled, then $0 \leq F_X(x) \leq 1$ and random samples of f_X may be generated by passing a uniform random variable r where $0 \leq r \leq 1$ into the inverse cumulative distribution function.

This algorithm allows samples from distributions to be efficiently generated. Small errors in the distribution will be generated due to the numerical precision of the integration and interpolation steps, but these can be calculated to any degree of accuracy required. The integration only needs to be calculated one time. And the interpolation can be efficiently calculated by creating an interpolation kernel during the initial calculation that can be called for each new set of samples. This one-dimensional algorithm is used for calculating the X-ray energy spectrums during the simulations.

For the two-dimensional distribution functions, a slightly different method must be used to evaluate the inverse cumulative distribution functions. The cumulative distribution function is now given by:

$$F_{XY}(x,y) = \int_{-\infty}^{y}\int_{-\infty}^{x} f_{XY}(s,t)ds\,dt$$

where it is assumed that this is normalized such that $\lim_{x,y\to\infty} F_{XY}(x,y) = 1$. Then samples may be generated from this distribution in two steps. Given two uniform random variables r_1 and r_2, first the marginal cumulative distribution function is calculated:

$$F'_X(x,y) = \int_{-\infty}^{x} f_{XY}(s,y)ds$$

then the full cumulative distribution function is calculated:

$$F'_{XY}(x,y) = \int_{-\infty}^{y}\int_{-\infty}^{x} f_{XY}(s,t)ds\,dt.$$

Then an interpolant function is created for the marginal CDF function:

$$z_X = F_X(x,y) \rightarrow x = F_X^{-1}(y,z_X)$$

and similarly for the full CDF function:

$$z_{XY} = \lim_{x\to\infty} F_{XY}(x,y) \rightarrow y = F_{XY}^{-1}(z_{XY}).$$

Then to generate the random numbers from the defined distributions, the uniformly distributed variables r_1 and r_2 are passed into the following functions:

$$Y = F_{XY}^{-1}(r_1)$$

$$X = F_X^{-1}(Y,r_2)$$

to yield the two-dimensional distributions. This algorithm is used to generate the two-dimensional super-Gaussian cone beam angular distribution. Using the inverse cumulative distribution functions becomes prohibitively difficult for higher dimensional distributions. Therefore, a general purpose, arbitrary dimension sampling algorithm was developed that could be used for generating the four-dimensional Lambertian cosine distribution of light rays.

E.2.3.4 Adaptive Rejection Sampling

In a physical experiment, many of the photons generated by an X-ray source are not directed towards the scintillator screen and thus do not contribute to the generated scintillator image. This effect is even more pronounced when a camera images a scintillator screen and most photons emitted by the screen never intersect the camera lens. Since these photons never contribute to the generated images, the simulation can avoid modelling light rays that do not intersect the scintillator or camera lens to increase computational efficiency, while maintaining the same level of fidelity.

The light rays may be rejection sampled by testing whether individually generated light rays will intersect the scintillator or camera lens and only saving those light rays that will contribute to the images. This testing process is computationally fast, however, if the majority of light rays do not intersect, then the overall process of generating the light rays will still be computationally slow since many potential light rays are tested that are never used in the simulation. Ideally, light rays should be generated in such a way to ensure that there is a high probability that they will contribute to the simulation.

When generating visible light data emitted by the scintillator, the light ray origin coordinates and directions can be calculated by passing uniform random numbers sampled from the unit hypercube into the inverse cumulative distribution functions to yield the cosine distributions. Unfortunately, only a small domain of the hypercube corresponds to intersecting light rays in typical simulation geometries and this domain may be complexly shaped as illustrated in Fig. A-41. The low ratio of the volume of the hypercube that is valid and the complex shape of the valid region, make determining an optimal was of generating this distribution difficult. Moreover, directly inverting the light ray

generation functions to determine which light rays will be valid is analytically impossible. Therefore, the hypercube needs to be selectively sampled to yield a higher proportion of valid light rays.

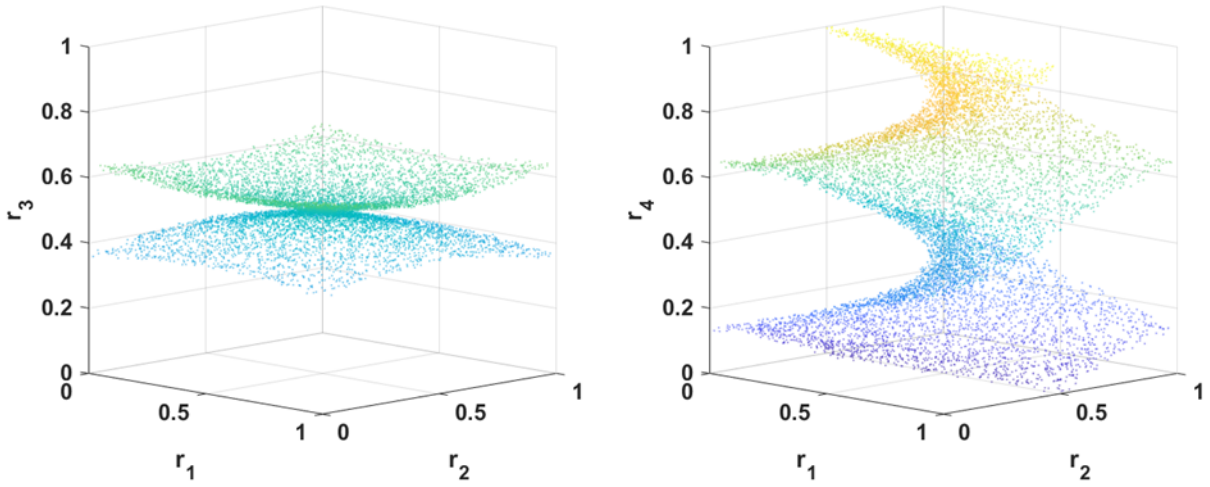

Figure A-41. An example four-dimensional hypercube sampling showing the valid intersection points for a typical scintillator-camera geometry. The valid domain of the hypercube only corresponds to approximately 0.2% of the total volume and the domain is characterized by an hourglass shaped surface on the left and a helical shaped surface on the right, illustrating the difficult nature of efficiently sampling this space. The colormap encodes the vertical axis height.

Adaptive sampling begins by uniformly sampling the entire hypercube with some number of points resulting in a uniform density ϱ of samples. Next, all points within the hypercube are tested to see whether they result in valid intersections, yielding a first approximation to the valid domain. The hypercube is then sub-divided into a number of smaller hypercubes along each dimension and the regions corresponding to each smaller hybercube are tested to determine whether they should be used in the next sampling. Since the nearest neighbor to any sampling point will follow a Rayleigh distribution, a distance d can be calculated that is likely, within some specified probability P, to contain another valid sample point. This distance d is taken from the center of each smaller hypercube or subcube.

$$d = 4\sqrt{-\frac{\log(1-P)}{\pi\,\rho}}$$

Each subcube is then tested whether it lies within the specified distance to a valid point. This process can be efficiently executed if the set of valid points is stored in a k-d data structure. Then if the number of valid subcubes equals n, the probability that the full valid domain lies inside the set of valid subcubes becomes P^n. This implies that the initial probability P should be chosen to be very close to one and that the number of sub-divisions of the hypercube should not be chosen to be too high to ensure that the valid region is found to a high degree of confidence.

Once all subcubes have been tested, a second iteration of sampling is completed within only the valid subcubes, and this process is repeated. After several iterations, the valid domain of the hypercube will be found with a high degree of confidence and a high proportion of the samples taken from the valid subcubes will result in valid intersection points. This allows the four-dimensional spatial and angular

distributions of the light rays to be efficiently sampled for the cosine Lambertian distribution generated by the scintillator screen.

Since there is a non-zero probability that a valid region of the hypercube will not be identified, there is a chance that the light rays may be under sampled, particularly with rays that emanate from the outer edges of the scintillator and intersect the outer edges of the camera lens. However, since these light rays only correspond to a small portion of the overall number of valid light rays, under sampling only minimally affects the simulation output images. Additionally, by increasing the probability P and decreasing the total number of subcubes, the under-sampling effects can be minimized.

E.3. Results

Simulated X-ray images were generated from a variety of situations during testing of the software. Simple material attenuation images were generated and then were compared to experimental images taken of calibration step-wedges and shown to yield similar attenuation levels. This provided a means of validating that the simulation produced images with similar intensity levels to those created by a physical imaging system. Additional testing generated tomographic imaging datasets for validating reconstruction algorithms. These images demonstrated that coherent data could be generated from multiple viewpoints that could be processed with standard X-ray image processing algorithms. The final testing involved generating images from CAD files to simulate a real-world setup prior to physical experiments. The simulations were then used to optimize the experiment prior to resource intensive test shots.

E.3.1. Attenuation Comparisons

The primary purpose of the simulation software is to be able to generate images that closely mimic those that are produced in physical experiments. While many other effects control the final output images, the ability to closely model the intensity attenuation as the X-rays pass through objects is particularly important since this controls the structural details that are resolvable in images. If the simulations show that the X-ray images from an experiment will be too dark to sufficiently resolve the measurement of interest, then the experimental setup can be modified prior to the tests for example. Conversely, for particularly thin objects, the simulations may show that higher density materials need to be used in the experiment to attain the contrast necessary to complete a specific measurement.

To validate that the simulation produces images with similar levels of attenuation, experimental images were collected of step-wedge style calibration objects composed of three different alloys and at two different X-ray energy levels. Then simulated images were generated of the same materials at the same energy levels. The three calibration materials were 7075 aluminum, 304 stainless steel, and pure tungsten which were chosen since they span a wide range of X-ray attenuation rates. The exact composition of the alloys used in the step-wedges was unknown, so the industry standard nominal mass ratios [3] were used in the simulations. The images were collected on a Scintacor Rapidex Bright scintillator screen that was imaged by a Photron SA-Z camera using an 85 mm f/1.4 Nikkor lens. The X-ray energy levels were set to 75 keV and 150 keV and were generated by a Varian G-297 rotating tungsten anode X-ray tube. The X-rays generated by the tube pass through a window that corresponds to 0.50 mm of aluminum and the scintillator screen is backed with 0.76 mm of aluminum as well, so these were added as filters to the simulation. The intensity of both the experimental and simulated images was normalized by dividing the images by the brightfield background images that did not contain any imaging objects. Graphs comparing the results of the experiments and simulations are shown in Fig. A-42.

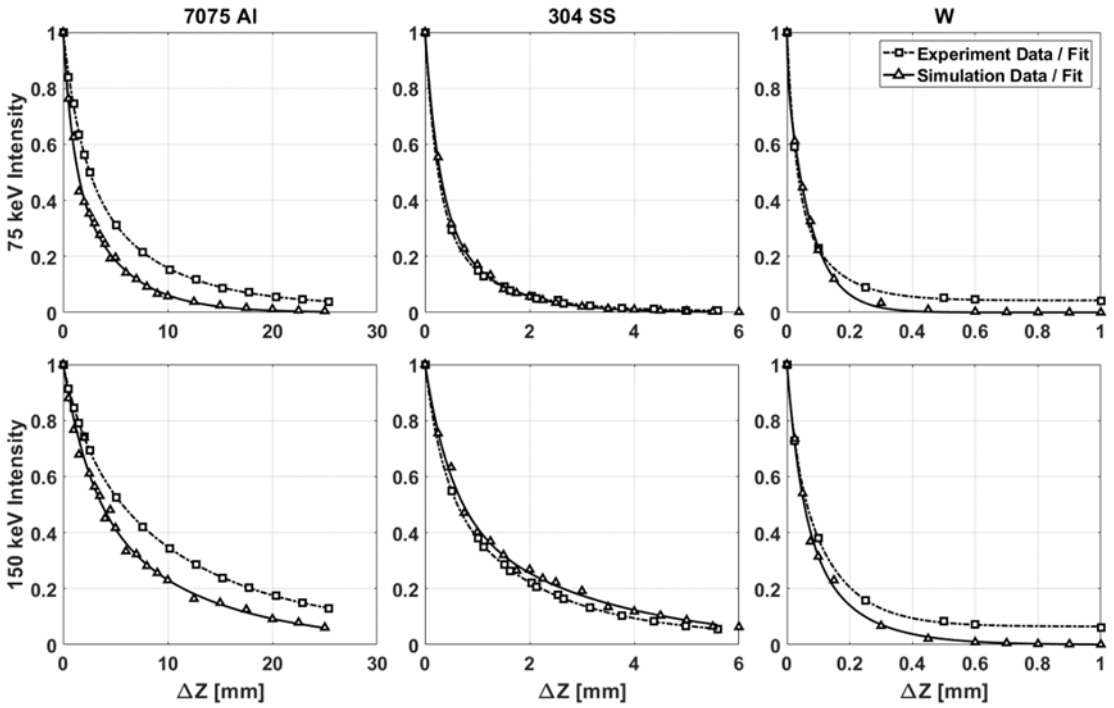

Figure A-42. These graphs compare the normalized intensity attenuation with material thickness for the experimental and simulated images for 7075 aluminum (left), 304 stainless steel (middle), and pure tungsten (right). The top row corresponds to a 75 keV X-ray source and the bottom row to a 150 keV source.

The experimental and simulated attenuation data show close agreement amongst all three materials and the two X-ray energy levels. The largest normalized intensity difference between the experiment and simulation is 0.16 with the average difference equaling 0.05 across the three materials and two energy levels. Similar levels of agreement were found with other materials, including other aluminum, steel, and tungsten alloys as well as for several composites and powder-based tantalum depositions. Additionally, since the three materials investigated here were chosen to span a wide range of attenuation rates, it is reasonable to assume that comparably accurate attenuation levels would be found with many other materials.

E.3.2. Tomographic Imaging Dataset

Tomographic imaging where images are collected from multiple viewpoints and then used to reconstruct the geometry of the imaged objects is a common application of X-ray imaging. For this reason, a tomographic dataset was generated using the X-ray simulation software and reconstructions were calculated from the object images.

The tomographic dataset consisted of sets of calibration and object images taken from five different viewpoints horizontally spaced 72° degrees apart. A CAD file of a Lego minifig toy was used as the imaging object since this object contained relatively complicated geometries while being an object that is easily recognizable with correct images and reconstructions being easy to visually confirm. The

object was simulated with a 6061-aluminum composition and vertically scaled to a height of 50 mm. X-ray energy levels of 150 keV were used to generate the simulated images.

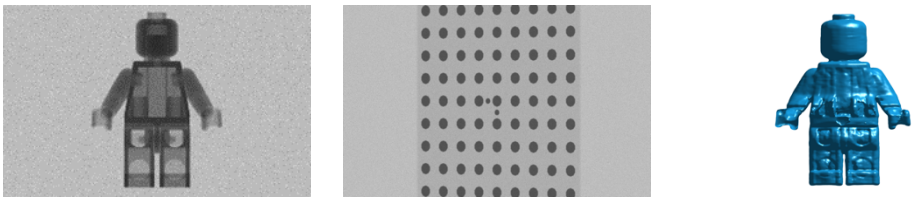

Figure A-43. The left image shows one of the five simulated X-ray images. The middle image is one simulated calibration grid used to establish the relative positioning of the imaging system. And the right image shows an isosurface plot of the reconstruction.

The calibration data consisted of simulated images of a calibration grid comprised of 0.5 mm thick copper circles over a 1 mm composite PCB board. The grid was rotated to five different positions that were spaced so that each of the five imaging systems could view and extract useful calibration data from at least four of the five calibration grid images. Images where the grid was viewed approximately edge on were not used in the calibration. A pinhole camera model was used to calculate the camera calibration of each of the imaging systems.

The tomographic reconstruction was completed using a maximum likelihood expectation maximization (MLEM) algorithm [4] using the ASTRA toolbox [5-7] to perform the backprojections. Example object, calibration, and reconstruction images are shown below in Fig. A-43. Given that only five views were used to calculate the reconstruction, the quality of the reconstruction is relatively high. This indicates that the geometry of the simulated images is consistent both between the calibration and object images and amongst all five imaging systems. Further, the reconstruction quality shows that using a pinhole camera model is an acceptable calibration as would be expected from a cone-beam projection. Taken together, the results from the tomographic reconstruction test show that the X-ray simulation produces results that are geometrically consistent with experimental X-ray data.

E.3.3. Simulation of Experimental CAD Objects

The X-ray simulation software was used to generate simulated images of an experimental setup to investigate the expected imaging capabilities and modify the experimental geometry prior to a resource intensive test. In the experiment, cantilevered plates were mounted within a ring that was subjected to a large impulsive force. The goal of the experiment was to measure the dynamic motion of the plates using X-ray imaging. Both the plates and the ring were composed of aluminum which is relatively transparent to high-energy X-rays. A speckle pattern comprised of X-ray opaque tantalum dots was deposited on the cantilevered plates as well as a portion of the ring wall. Prior to the experiment, the X-ray simulation software was used to optimize several parameters.

Since the goal of the experiment was to measure the plate motion, several different tantalum dot thicknesses and diameters were simulated to determine the optimal dot geometry that resulted in the highest fidelity motion measurement. Additionally, multiple ring orientation angles, speckle pattern placement positions, and overall system magnifications were simulated. By adjusting these parameters in simulations, both the quantity and the quality of the data was optimized prior to performing the expensive experimental test.

The simulated X-ray images were created by importing CAD models of the experimental components into the simulation software. The specific alloy compositions of the aluminum plates, aluminum ring,

and steel mounting hardware were specified within the software parameter files. Additionally, since the tantalum deposition process resulted in the speckle dots being slightly porous, an estimate of the tantalum density was included as well. Both experimental and simulated images of the experimental setup are shown in Fig. A-44.

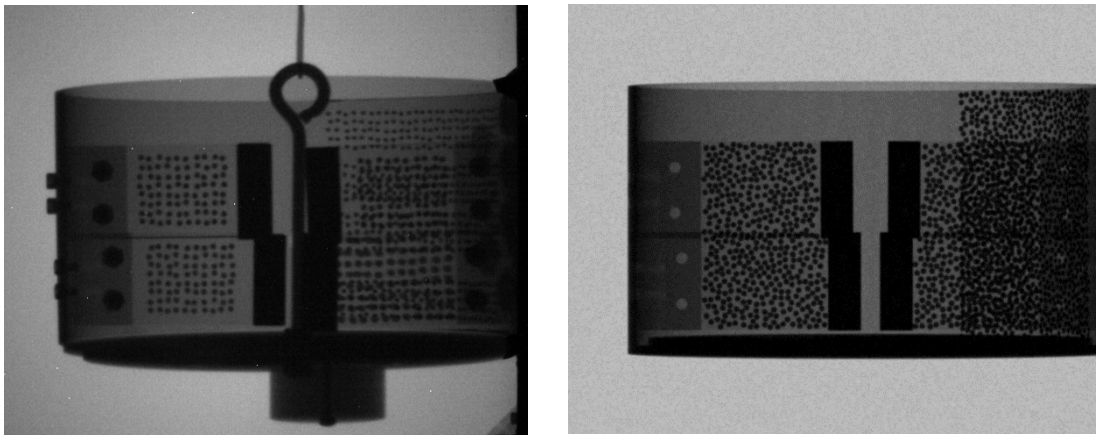

Figure A-44. This figure shows the similarity between images from an experiment (left) and from a simulation designed to model the experiment (right). The experimental image shows additional components used to support the test article including mounting hardware. Additionally, the experiment geometry was modified after the simulation was completed, resulting in additional foreshortening in the experimental image.

The experimental and simulated images showed excellent agreement overall. The geometry, relative intensities, and noise levels of the images closely match. The experimental image has a slightly increased foreshortening effect compared to the simulated image, but this is due to the fact that the test article and imaging scintillator were placed a small distance closer to the X-ray source in the experiment to increase the X-ray flux intensity to create brighter scintillator images. Additionally, the spatial density and the specific pattern of the tantalum dots was also changed between running the simulations and completing the experiment resulting in the speckle patterns appearing different.

Simulations that focused specifically on the speckle pattern were also completed that investigated the ability to accurately measure the cantilever plate motion. In these simulations, the speckle pattern was the only test article component geometry included, however the filtering effects of the cantilever plates and test ring were also included in the simulation. Additionally, the image noise in preliminary small-scale experiments was quantified and replicated in the simulated speckle pattern images by adding spatio-temporally filtered Gaussian noise to the simulated images.

These simulations then estimated the motion tracking uncertainty by measuring the variation in the calculated motion in a sequence of frames where the speckle pattern remained stationary. The deviation from a zero-valued measurement gave an approximation of the uncertainty that would be measured in the actual experiment. The measured uncertainty in the experiment was 0.32 pixels while the simulated images resulted in an uncertainty of 0.42 pixels, thus demonstrating the ability to estimate uncertainties from the simulated images.

To further quantify the similarity between the simulated and experimental images, the statistics of the image noise was analyzed. To avoid contaminating the measurements with contributions from the speckle pattern, a region containing only the background of the images and no speckle dots was selected from both the simulated and experimental images. First, the distribution of pixels values was measured within this region. The similar histograms are shown in the top two graphs of Fig. A-45.

Additionally, the spatio-temporal power spectrums of the regions were calculated. Since the scintillator acts as a low-pass filter to the original image projected by the X-rays, it was assumed that there would be correlations between adjacent pixels in both space and time. The closely related spatial power spectrums are shown in the bottom two graphs of Fig. A-45. The temporal power spectrums were similar as well, although the correlations only proceeded forward in time and showed less overall correlation since the decay time of the scintillator was fast compared to the frame rate of the camera.

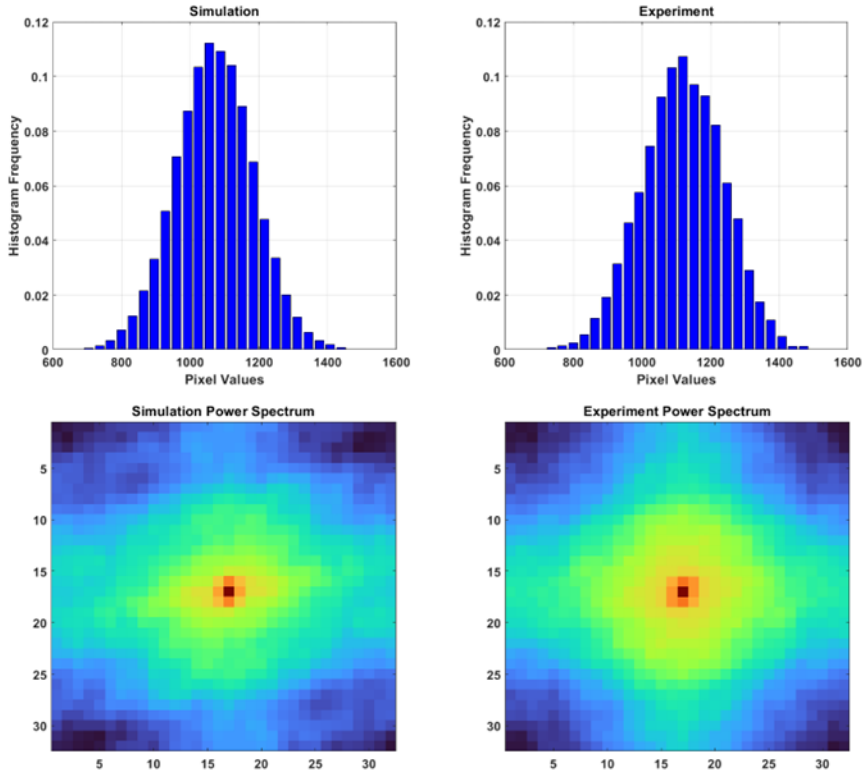

Figure A-45. The top two graphs show the pixel value histograms of the background of the speckle pattern simulation (left) and experimental (right) images and demonstrate the close statistical agreement in the image noise. The bottom two graphs show the spatial components of the power spectrums of the background noise of the simulation (left) and experimental (right) images. The comparable power spectrums indicate that the spatial components of the noise were similarly distributed as well.

E.4. Discussion

As has been shown, the X-ray simulation software produces images that are quite similar to experimental images both qualitatively and quantitatively. The majority of the phenomena that contribute to the output images are modelled with relatively high fidelity. However, there are limitations to the software that result in the simulated images differing from true X-ray images. These limitations result in relatively small differences between the simulated and experimental images and are mostly due to practical computational concerns.

As an example, including scattering effects of X-rays would require modelling the light ray directions changing a potentially unlimited number of times within objects. This would then require an exponential increase in computational time to perform the simulations. However, for the ranges of

X-ray photon energies being simulated, scattering occurs infrequently, so this effect can be neglected with little effect on the output images. The other effects that are listed here are similar in that including them might increase the fidelity of the simulation, but only at a large computational cost.

E.4.1. High Fidelity Simulation Components

The largest contributing effects to the overall image output are included in the X-ray image simulation software. The geometries of the X-ray source, the imaging objects, and the scintillator are all accurately modelled to within the design tolerances. This results in images where the shapes of imaged objects will be correctly simulated. The intensity and the sharpness of the objects are controlled by other factors.

Several sources of image blurring that decrease the overall sharpness of images are included in the software modelling. The non-zero area of the X-ray focal spot is included in this model resulting in geometric blurring of the images. Additionally, image blurring in the scintillator is included by allowing the user to specify the modulation transfer function of the detector. Motion blur can be accounted for by simulating multiple closely spaces frames with different object positions and then adding these images together. Finally, blurring due to focusing errors and diffraction effects are included in the camera model.

The relative intensities of the imaged objects are a function of the X-ray spectrum, the rates of attenuation by the imaged objects, and the sensitivities of the scintillator and camera sensor. The X-ray source spectrum is accurately modelled by the SpekPy software which has been validated against experimental data [1]. The material attenuation rates are taken from tables compiled by NIST [8, 9] that were calculated from a combination of theoretical and experimental data and then validated against experimental attenuation rates. The scintillator and camera sensitivities are specified by the user and are typically published by the manufacturers.

Together, the majority of the information in an X-ray image is controlled by the combination of the system geometry, the sources of image blur, and the object intensities. So, by accurately modelling these contributions to the images, high fidelity images are created by the simulation software. Despite this, there are some approximations used by the X-ray imaging model.

E.4.2. Simulation Approximations

Several approximations are made by the X-ray simulation software. Each of these approximations was considered to be an acceptable compromise when the additional complexity required to create a more accurate model was weighed against the small increase in simulation fidelity. The approximation that has the largest contribution to the overall image fidelity is the sub-sampling of the number of photons from what occurs in an actual experiment.

In a typical X-ray imaging application, approximately 1013 to 1015 photons will be generated by the X-ray source per image frame. Many of these photons will not directly contribute to the image formation. However, simulating even the smaller number of light rays that do excite the scintillator screen is computationally difficult without devoting significant computational resources, therefore the number of light rays simulated per image is typically decreased to 108 to 1010 light rays depending upon the application of the simulation. For common imaging sensors with approximately one million pixels, this still results in hundreds to thousands of light rays contributing to each pixel. And beyond some threshold, the statistical differences will be less than the contributions of other sources of noise, so this approximation is considered acceptable in most simulations.

An additional model approximation used in the simulations occurs when the camera lens refracts the light rays generated by the scintillator screen. The software uses a thin lens approximation to calculate the changes in the light ray directions. This model produces common camera imaging effects including focusing, depth-of-field, and aperture light control. However, higher order effects including lens aberrations, vignetting, and diffraction are not produced by the thin lens model. A full description of all lens elements, which is rarely provided by lens manufacturers, is necessary to account for lens aberrations and vignetting, so these effects are not accounted for in the simulation model. Lens diffraction is primarily a function of the effective lens aperture and light wavelength though, so this effect is included later in the simulation by applying an Airy disk convolution kernal to the sensor image. The images produced by the thin lens approximation will primarily deviate from experimental data when a low-quality lens that is subject to significant lens aberrations and vignetting is used. However, these effects are negligible in most modern scientific lenses.

E.4.3. Effects Not Included in Simulation

Several additional X-ray imaging effects were chosen not to be included in the imaging simulation due to their minimal effect on the overall output images. The most prevalent effect that was neglected is X-ray photon scattering. Additionally, refraction and diffraction effects on the X-ray photons were not included as well.

The overall absorption of X-rays in a material is a combination of multiple different effects including photoelectric absorption, Compton scattering, and coherent scattering, among several others that are relatively minimal at photon energies below 1 GeV. Below approximately 100 keV, the dominant absorption process is photoelectric absorption in which all the incident energy of an absorbed photon is transferred to an electron ejected from the atom [10]. Since the majority of photons generated by an X-ray source have energies well below the input tube voltage, this means that in many typical experiments the majority of photons will not experience scattering. The highest tube voltage that the software can simulate is 300 keV. At this voltage 71% of the photons generated by a tungsten anode tube have energies below 100 keV. Therefore, the effects of scattering were considered to be negligible at the energy levels simulated. If later versions of this software are expanded to higher energy levels, scattering may need to be included.

For common materials and typical X-ray wavelengths, the refractive index of X-rays varies from one by only 10-5 to 10-7 [10]. As a result, in all but the most extreme circumstances, the effects of refraction changing the direction of the X-ray photons will result in shifts in position on the scintillator screen that are well below the resolution of the X-ray imaging system. As a result, refraction effects were also not included in the simulation.

Diffraction occurs when wavefronts of light interfere with themselves and produce diffraction fringes that degrade the quality of a produced image. Diffraction primarily becomes important when the light passes through an aperture that is small in comparison with the wavelength of light. However, since the wavelength of X-rays is on the order of nanometers with imaging objects many orders of magnitude larger, the imaging effects from diffraction for typical X-ray imaging systems is also negligible and is thus neglected from the simulation.

E.5. Conclusion

This software provides a means of generating images from simulated complex experimental X-ray imaging systems. These images can be used for determining visible geometry prior to experiments, optimizing experimental parameters before manufacture or assembly, estimating uncertainties in measurements, and generating data for testing experimental analysis methods or processing

algorithms. Data produced by the simulation software was compared against experimental data in a variety of different test cases and shown to produce similar results.

Future versions of the software may expand the energy range of X-rays that can be simulated to higher levels. This would require incorporating additional X-ray spectrum models as well as likely including scattering effects in the light ray propagation simulation. Additionally, since many experiments are inherently dynamic in nature, adding the ability to directly simulate the motion of the imaged objects along with the temporal decay of the scintillator images would add a higher degree of fidelity to the simulations.

E.6. Acknowledgments

This software was developed through the course of discussions and experiments involving multiple people working in a variety of research areas. John Miers processed simulated tomographic data and provided feedback on realistic imaging scenarios. Elizabeth Jones worked with complex simulated imaging data and gave useful feedback on the modelling of realistic sensor noise. And Malerie Baeza lead a physical experiment that was compared to simulation results, providing both CAD models and experimental setup parameters for completing the simulation and validating the results.

E.7. References

[1] Bujila, R., Omar, A., & Poludniowski, G. (2020). A validation of SpekPy: A software toolkit for modelling X-ray tube spectra. Physica Medica, 75, 44-54.

[2] Möller, T. and B. Trumbore, (1997). Fast, Minimum Storage Ray-Triangle Intersection. Journal of Graphics Tools, 2(1), 21-28.

[3] Metals Handbook Desk Edition. 2nd ed. 1998: ASM International.

[4] Lange, K., & Carson, R. (1984). EM reconstruction algorithms for emission and transmission tomography. J Comput Assist Tomogr, 8(2), 306-16.

[5] Palenstijn, W. J., Batenburg, K. J., & Sijbers, J. (2011). Performance improvements for iterative electron tomography reconstruction using graphics processing units (GPUs). Journal of structural biology, 176(2), 250-253.

[6] Van Aarle, W., Palenstijn, W. J., De Beenhouwer, J., Altantzis, T., Bals, S., Batenburg, K. J., & Sijbers, J. (2015). The ASTRA Toolbox: A platform for advanced algorithm development in electron tomography. Ultramicroscopy, 157, 35-47.

[7] Van Aarle, W., Palenstijn, W. J., Cant, J., Janssens, E., Bleichrodt, F., Dabravolski, A., ... & Sijbers, J. (2016). Fast and flexible X-ray tomography using the ASTRA toolbox. Optics express, 24(22), 25129-25147.

[8] Saloman, E. B., & Hubbell, J. H. (1986). X-ray attenuation coefficients (total cross sections): Comparison of the experimental data base with the recommended values of Henke and the theoretical values of Scofield for energies between 0. 1-100 keV (No. NBSIR-86-3431). National Bureau of Standards, Washington, DC (USA). Center for Radiation Research.

[9] Saloman, E.B., J.H. Hubbell, and J.H. Scofield, (1988), X-ray Attenuation Cross-Sections for Energies 100 Ev to 100 Kev and Elements Z = 1 to Z = 92. Atomic Data and Nuclear Data Tables, 38(1), 1-196.

[10] Toda, H. (2021). X-Ray CT: Hardware and Software Techniques. Springer Nature.

DISTRIBUTION

Email—Internal

Name	Org.	Sandia Email Address
LDRD Office		ldrd@sandia.gov
Technical Library	1911	sanddocs@sandia.gov

This page left blank

Sandia National Laboratories

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.