High Performance Computing for Neutron Tomography Reconstruction

THE UNIVERSITY of TENNESSEE KNOXVILLE
GEORGIA COLLEGE — GEORGIA'S PUBLIC LIBERAL ARTS UNIVERSITY
NSF

Students: Cain Gantt (GCSU), Zongpu Li (CityU)
Mentors: Rick Archibald (ORNL), Kwai Wong (UTK)

JICS Joint Institute for Computational Sciences
香港城市大學 City University of Hong Kong
OAK RIDGE National Laboratory

Background

Neutron tomography is the process of imaging an object with neutrons, then reconstructing a model of the object, or phantom, much like a medical CT scan does using X-rays. However, unlike X-rays, neutrons can travel further into a material before being scattered or absorbed, allowing imaging of a wider range of materials, most notably metals. Laminography is different from tomography only in that the axis about which the phantom is rotated is not perpendicular to the neutron beam (Fig. 1). After imaging, the 3D model of the phantom is created with a reconstruction method, such as the commonly used filtered backprojection (FBP) method. The layout of a simple tomography experiment in two dimensions is shown in Fig. 2, and the equation for the 2D inverse radon transform that the FBP algorithm is based upon is shown in Eqn. 1.

Oak Ridge National Laboratory (ORNL) hosts a large number of researchers from around the world for collaboration and to conduct experiments using their two renowned neutron sources: the Spallation Neutron Source and the High Flux Isotope Reactor. Neutron imaging experiments can be completed in the span of a few hours, but the data collected from imaging experiments currently can take up to days to process. It is often the case that visiting researchers may have to leave before knowing if their experimental setup was viable. This necessitates a faster way to process the data collected.

Methods

Our goal is to run FBP in parallel to use HPC resources.

We started with serial MATLAB code that simulated a laminography experiment, and filtered with a standard tomography filter. The phantom data it generated and the model it reconstructed are displayed in Fig. 3. The overhead required to run the MATLAB environment severely limited code scalability.

To remedy these limitations, we set out to first implement a filter designed for laminography, and then to implement the FBP algorithm into the C programming language. The algorithm in serial is outlined to the right. This can be later modified to run in parallel using the Message Passing Interface (MPI) library to take advantage of tasks that can be run simultaneously, which are highlighted.

- LOAD projections, phis, theta
- **FOR EACH** projection
  - DO FFT along columns
  - apply filter
  - DO inv. FFT along columns
- **FOR EACH** filtered projection
  - INIT coord. matrices
  - CALC rotations
  - **FOR EACH** voxel in volume
    - CALC interpolation
    - ADD to volume
- apply weight to volume

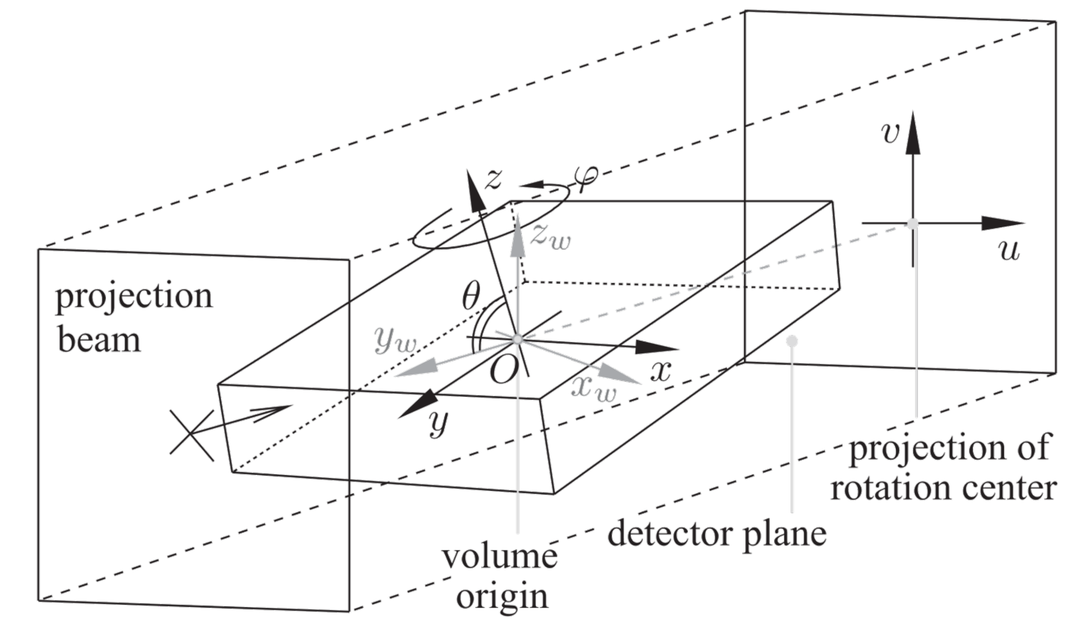


Fig. 1. Basic geometry of a laminography experiment. The laminographic angle θ measures the tilt of the rotation axis *z*.

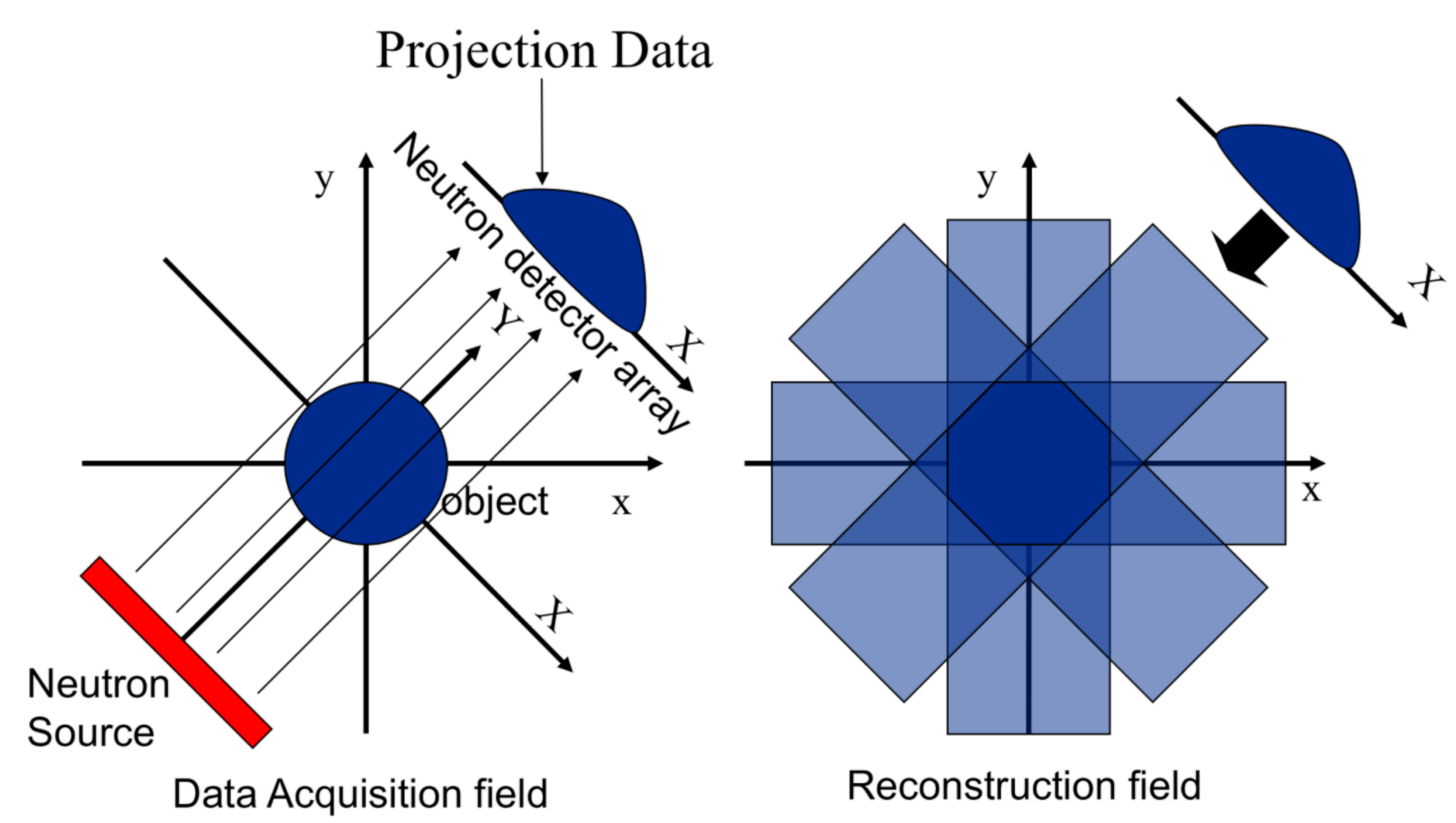


Fig. 2. Depiction of backprojection in 2D.

$$f(x,y) = \int_0^{\pi} p_f(x \cdot \cos\theta + y \cdot \sin\theta)\, d\theta$$

Eqn. 1. Inverse Radon Transform for 2D tomography.

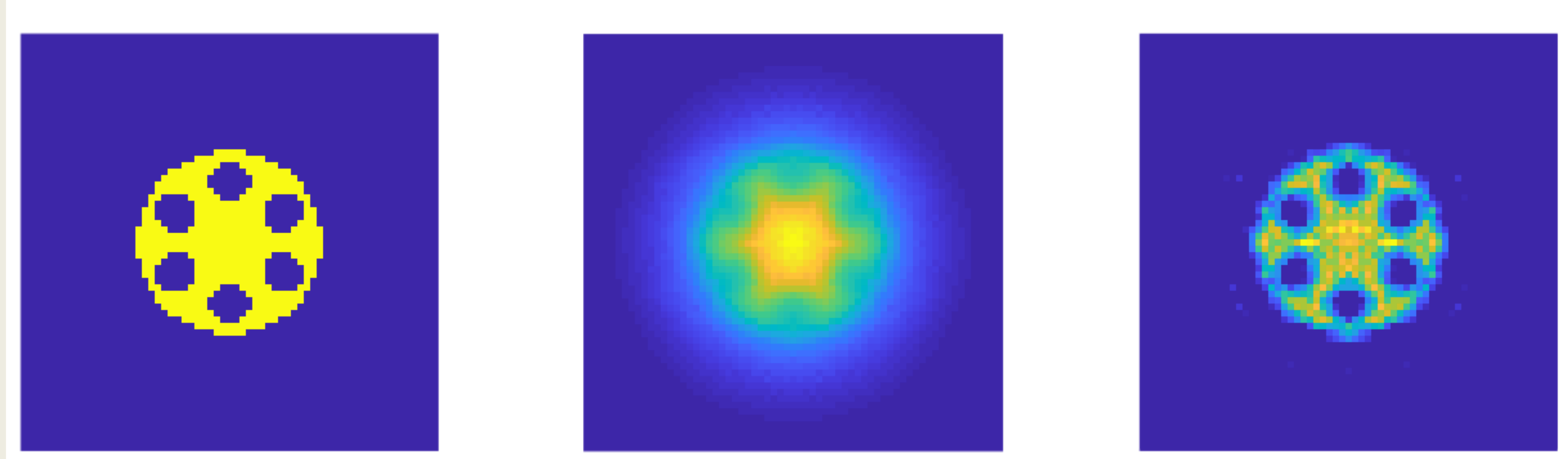


Fig. 3. Central cross-sections of simulation 65x65x65 volumes, including (left) the 3D model created before projections were taken, (center) the volume reconstructed without a filter, and (right) with the laminographic ramp filter.

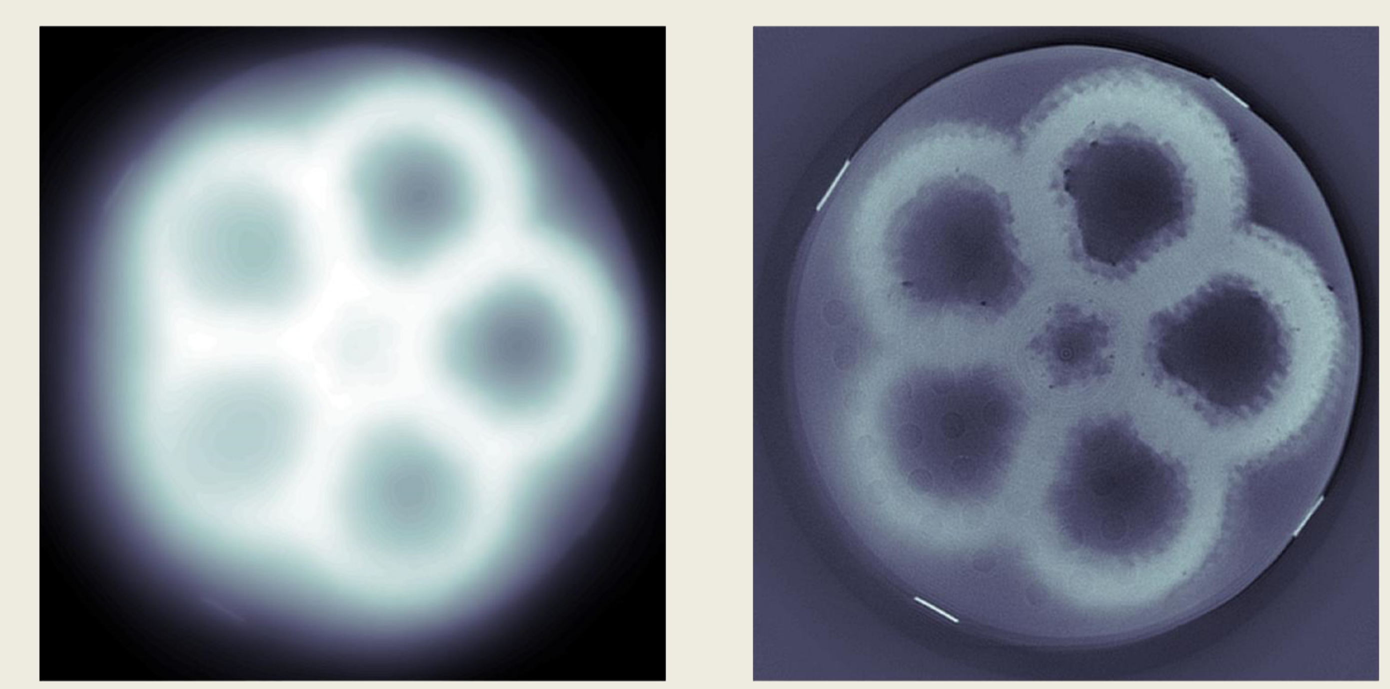
Analysis

The amount of detail recovered by using the improved filter is plain to see in Figures 3 and 4. The phantom in Fig. 4 is a thin additively manufactured metal plate fabricated and imaged at ORNL. In the unfiltered image (left), only the basic structure is visible, with most details being lost to blurring. The filter eliminates a large degree of this blur. The filtered cross-section (right) allows closer examination of the internal structure of the plate, notably imperfections in the manufacturing process. The thin strips of tape used to hold the plate in place during imaging can also be seen (four white lines around the edges).

Figure 4 shows serial backprojection times for simulations at different resolutions, and the time to reconstruct the single cross section.

resolution param.	resolution (px)	time elapsed (s)
32	$65 \times 65 \times 65$	4.563
64	$129 \times 129 \times 129$	45.630
128	$257 \times 257 \times 257$	(failed to complete)
⋮	⋮	⋮
(1× cross-section of ORNL data)	$1501 \times 1501 \times 1$	737.59

Fig. 4. Cross sections of volume reconstructed from ORNL data. The complete dataset is 1094 projections, each 701x1501.



References

A. Myagotin, A. Voropaev, L. Helfen, D. Hänschke, and T. Baumbach, “Efficient Volume Reconstruction for Parallel-Beam Computed Laminography by Filtered Backprojection on multi-core Clusters”, IEEE Transactions on Image Processing, vol. 22, no. 12, pp. 5348–5361, Dec. 2013.

Acknowledgements

We would like to give special thanks to our mentors, CityU and Georgia College for their support. This research was made possible by contributions from the National Science Foundation. The computing resources were provided by the XSEDE project.