Parallel Iterative Reconstruction Tomography

Matthew Kehoe

University of Illinois at Chicago
Mentors: Wendy Di and Matthew Otten

July 30, 2019

Outline

Introduction and Background

Parallelization

Results

Future Work

Introduction and Background

- A fundamental mathematical tool in tomography is the Radon transform.
- For a compactly supported function $f : \mathbb{R}^2 \to \mathbb{R}$, the Radon transform is defined by

$$Rf(\tau, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y)\delta(\tau - x\cos\theta - y\sin\theta)dxdy$$

  where δ is the dirac delta function and the domain is restricted to $\tau \in [0, \infty)$ and $\theta \in [0, 2\pi)$. It is assumed that f is well behaved.
- The Radon transform of a function is frequently called its sinogram.

Introduction and Background

Radon Transform

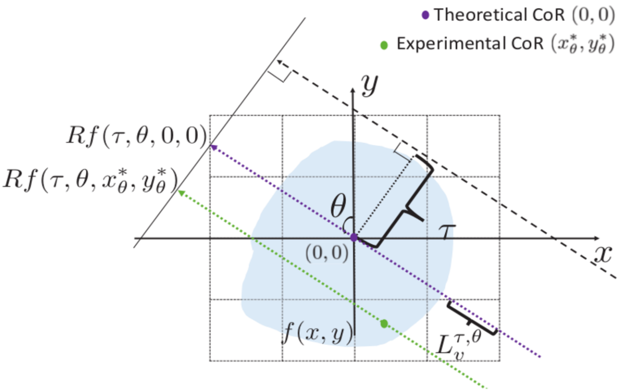

Figure 1: Geometric sketch of the Radon transform, which maps f from (x, y) space to (θ, τ) space. The purple line and the green line denote rotations of their previous position, as the y-axis, with respect to different CoRs, respectively.

Introduction and Background

- We implement an optimization-based reconstructive algorithm which estimates parameters and reconstructs the image concurrently.
- The code we developed interacts with PETSc/Tao.
- We calculate the function value and gradient through the use of FormFunctionGradient. To do this, we implement two methods
  - The first method is the standard reconstruction. The gradient is calculated without recovering the coordinates of the CoRs.
  - The second method is reconstruction with error correction. This method uses an implicit approach to recover the coordinates of the CoRs for (x_θ^*, y_θ^*). A normalized Gaussian filter is used to dampen high-order Fourier coefficients.
- Variable bounds are set through TaoSetVariableBounds.
- The optimization problem is then solved through TaoSolve.

Parallelization

- We measure the total execution time of reconstructing the Shepp–Logan phantom image with $1, 2, 4, 8, 16$, and 32 processors.
- The head phantom image is created through phantom(1000) in Matlab. The image data is saved to a file with 497581 lines.
- The number of angles (N_θ) is set to 100 while the number of beam positions (N_τ) is set to 1415.
- We analyze the results to calculate speedup and parallel efficiency.

Parallelization

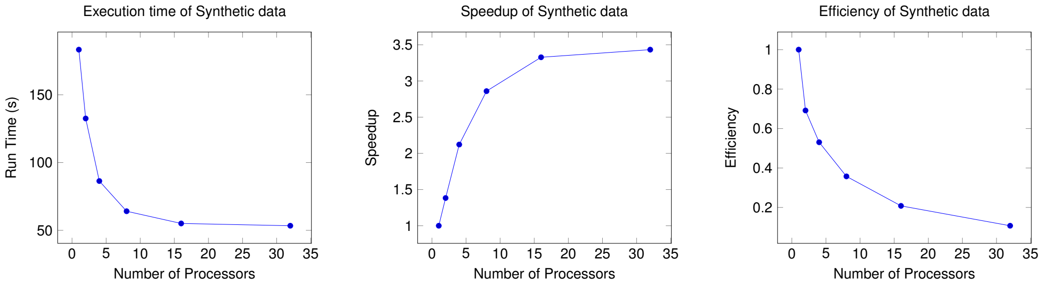

Figure 2: **Left:** Execution time for the Synthetic data set as a function of the number of processors. **Middle:** Performance of a parallel implementation - Speedup as a function of the number of processors. **Right:** Performance of a parallel implementation - Efficiency as a function of the number of processors.

Results

- We create a high resolution image of the the Shepp–Logan phantom image by phantom(2000) in Matlab. The image data is saved to a file with 1992337 lines.
- The number of angles (N_θ) is set to 1000 and the number of beam positions (N_τ) is set to 2829. The number of processors is set to 32.
- Analysis is performed through reconstructing the image with and without error correction.

Results

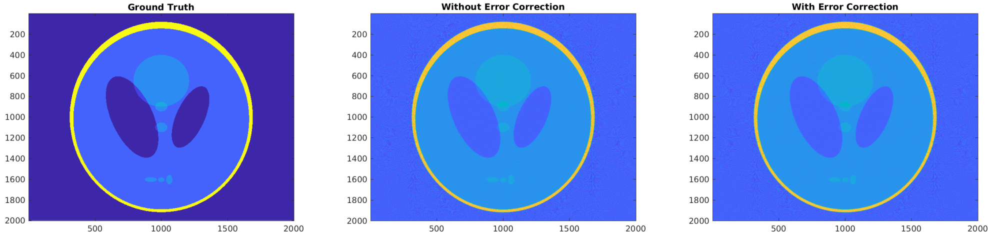

Figure 3: **Left:** Ground truth of the Shepp–Logan phantom image created by phantom(2000) in Matlab. **Middle:** The reconstructed Shepp–Logan phantom image without error correction. **Right:** The reconstructed Shepp–Logan phantom image with error correction.

Future Work

- Investigate PIRT functionality with and without error correction on more data sets.
- Analyze the existing code for further performance improvements (replace VecSetValue with VecSetValues and make other improvements).
- Decide which Tao solver is best for use with PIRT.
- Investigate theoretical results after our reconstructive algorithm has been fully optimized.