Fast Gamma-Ray Burst Source Localization Pipelines

Marion Sudvarg[a], Jacob Wheelock[a], Jeremy Buhler[a], James Buckley[a], Wenlei Chen[b]
msudvarg@wustl.edu jacobwheelock@wustl.edu jbuhler@wustl.edu buckley@wustl.edu chen6339@umn.edu
[a]Washington University in St. Louis [b]University of Minnesota, Twin Cities

Software Repositories
https://sbs.wustl.edu/ADAPTsoftware.html

Overview

The Advanced Particle-astrophysics Telescope (APT) [1] is a planned space-based observatory to survey the entire sky for gamma-ray bursts (GRBs). It seeks to promptly detect these transient events, then communicate with narrow-band instruments for follow-up observations. To this end, we are developing analytical methods for real-time detection and localization of GRBs, then parallelizing and accelerating the software pipeline to maintain sufficient throughput for computing hardware that might fly onboard the orbiting platform.

As described in [2], we focus on detecting events for which a GRB's photons Compton-scatter one or more times within the instrument until they are eventually photoabsorbed. All scatterings for one photon appear simultaneous at the time resolution of the detector. Our localization pipeline, then, has three primary stages: (1) reconstructing each photon's trajectory in the instrument to estimate an annulus containing the photon's source direction; then combining the annuli from all detected photons to estimate the most likely direction by (2) finding a rough approximation of the direction from a set of initial candidates according to a maximum-likelihood approach; then (3) performing iterative least-squares refinement to produce a final estimate of source direction. Such analysis must be simultaneously accurate and fast, even on a low-power, embedded computational platform.

Fluence: Fluence describes the energy density induced across the APT detector by a gamma-ray burst. In our simulations [3], the number of photons scales linearly with fluence.

Input size is proportional to fluence.

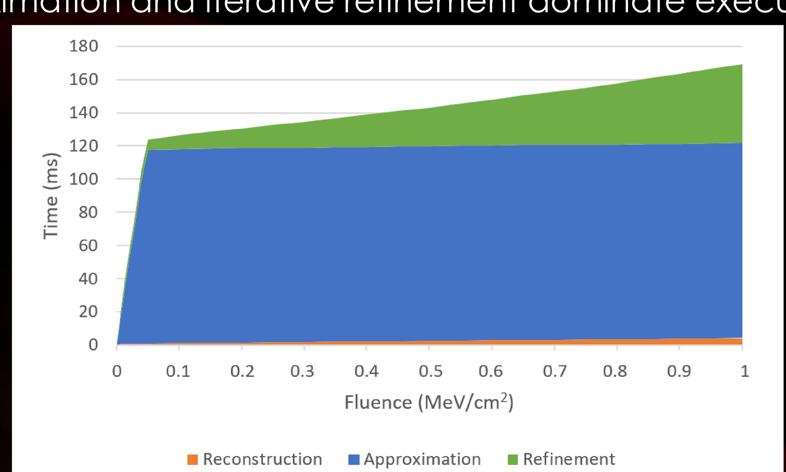
Localization Accuracy

- In [2], we measured our pipeline's ability to accurately localize GRBs.
- Localization errors are measured in degrees over 1000 trials for each fluence.
- We achieve consistent subdegree localization for high fluence events.

Fluence	Mean Error	Std Dev	68% Containment	95% Containment
0.03 MeV/cm^2	2.15	1.22	2.53	4.42
0.1 MeV/cm^2	1.21	0.64	1.45	2.32
0.3 MeV/cm^2	0.70	0.36	0.87	1.32
1.0 MeV/cm^2	0.35	0.20	0.42	0.72

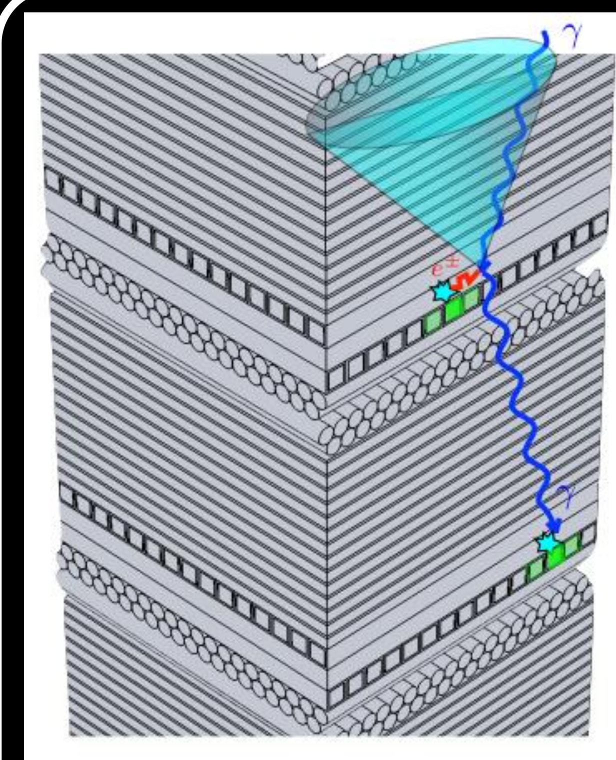
CPU Performance

- In [2], we parallelize the pipeline to target a low-power ARM Cortex-A53 processor.
- Execution times are averaged over 200 trials for each fluence.
- Initial source approximation and iterative refinement dominate execution time.



In this work, we accelerate the approximation and refinement stages and estimate running time on an NVIDIA Jetson NX Xavier system. Its 10-watt power requirement makes it comparable to what might fly onboard the APT platform.

Reconstruction



A gamma-ray photon, γ, enters the instrument from the top, then Compton-scatters, before finally being photoabsorbed. FPGA-based hit detection will sporadically write collections of scattering interactions into device memory. The order of interactions must be reconstructed, as described in [2]. The first two implied scatterings define an

annulus ($\mathbf{c}$, φ, σ)

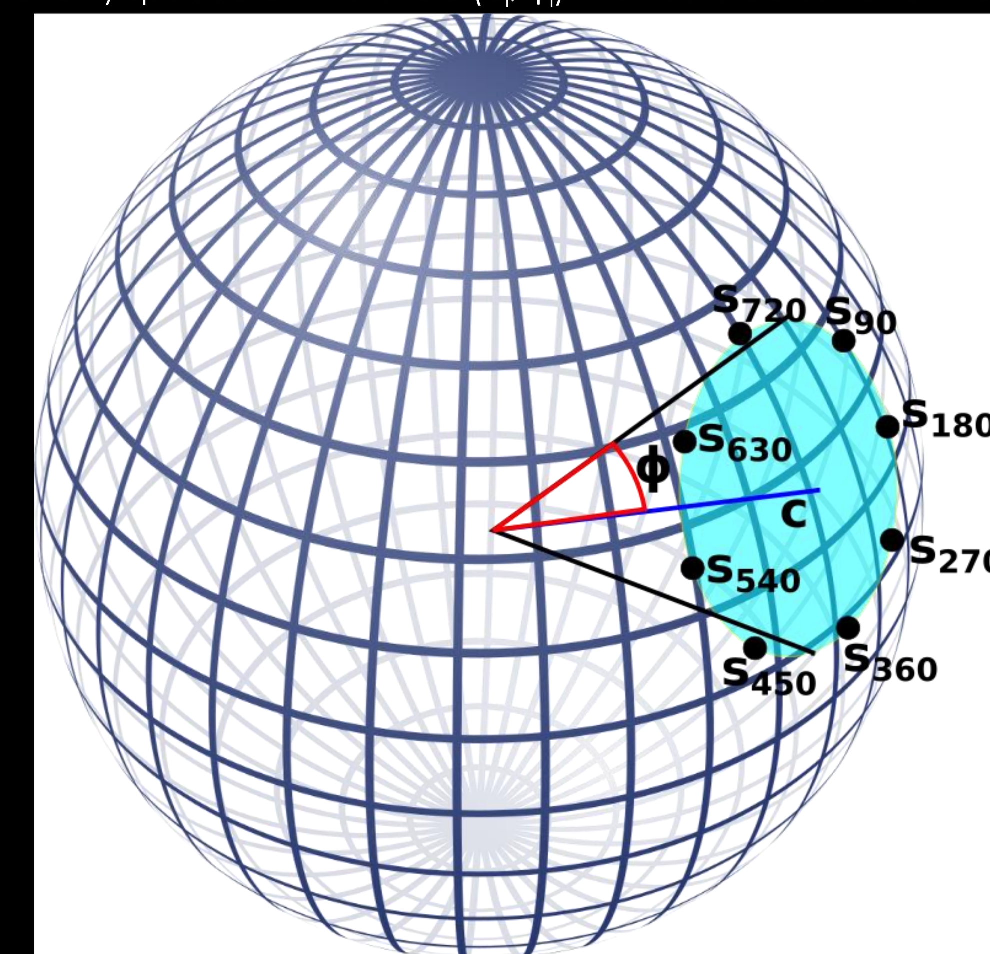
containing an estimated source direction:

- $\mathbf{c}$: The vector between the first two interactions
- φ: The scattering angle implied by the photon's energy before and after the first interaction
- σ: Propagated uncertainty based on spatial and energy measurement error

During a typical GRB, several thousand photons must be reconstructed.

Initial Source Approximation

1. From N annuli, select 20 at random, assign each to a CUDA block.
2. For each annulus i, test a set of 720 candidate source directions $\mathbf{s}_j$ evenly spaced on the circle ($\mathbf{c}_i$, φ_i). Distribute across GPU threads.



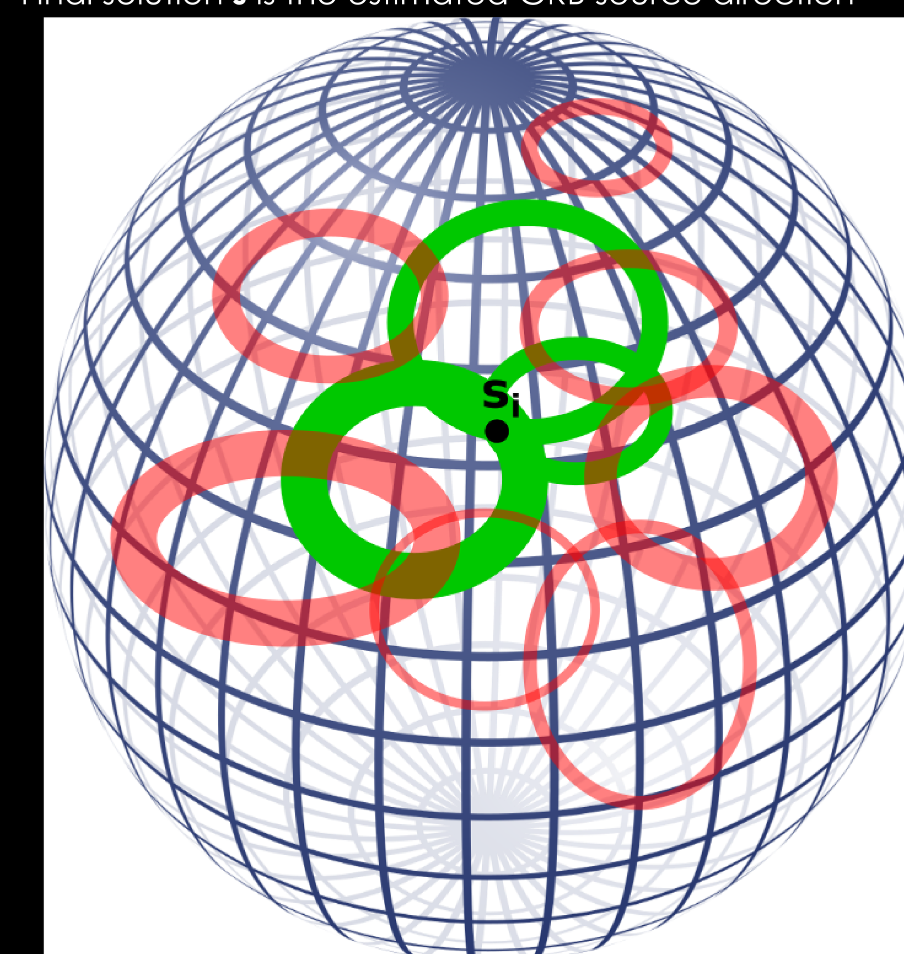
3. Calculate the joint log-likelihood for each candidate source direction with respect to all input annuli.
4. Select the candidate direction from each annulus with the highest likelihood.
5. Average over these 20 estimates, weighted by likelihood, to produce an initial approximation $\mathbf{s}_0$.

Iterative Refinement

1. Begin with the estimate $\mathbf{s} = \mathbf{s}_0$ (from the approximation stage)
2. For each annulus i, test whether the angle arccos($\mathbf{c}_i \cdot \mathbf{s}$) lies within $3\sigma_i$ of φ_i.
3. For those that do, generate linear constraints $\mathbf{c}_i \cdot \mathbf{s} = \cos \varphi_i$
4. Require $\mathbf{s}$ to be a direction vector; this unit-norm constraint is quadratic in the coordinates of $\mathbf{s}$
5. Reduce the problem to a quadratic eigenvalue problem [4]
   - Forming the matrix for the problem is O(N^2) from N annuli
   - The matrix dimension is proportional to the 3 coordinates of $\mathbf{s}$

   (Steps 2-5 are parallelized in purpose-specific CUDA kernels)

6. Solve on the CPU with Eigen [5], to get a refined estimate for $\mathbf{s}$
7. Iterate 20 times, repeating steps 2-6
8. Final solution $\mathbf{s}$ is the estimated GRB source direction

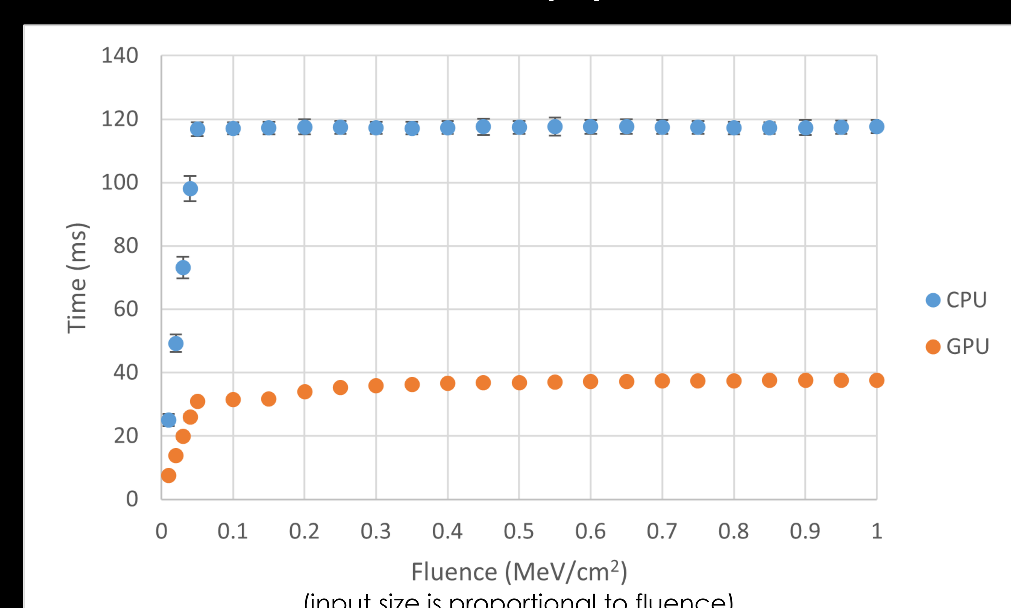


At refinement step i, only annuli (green) within 3 std dev of the current estimated source vector $\mathbf{s}_i$ are used as constraints for the least-squares problem. Other annuli (red) are ignored.

Pipeline Performance

- Mean execution times over 200 trials per fluence.
- Error bars denote a single standard deviation about the mean.
- GPU execution times were measured on an NVIDIA GeForce RTX 2080
- We scaled execution times to estimate performance on an NVIDIA Jetson NX Xavier, which has 6 SMs, which we conservatively estimate to perform half as quickly as each of the RTX 2080's 46 SMs

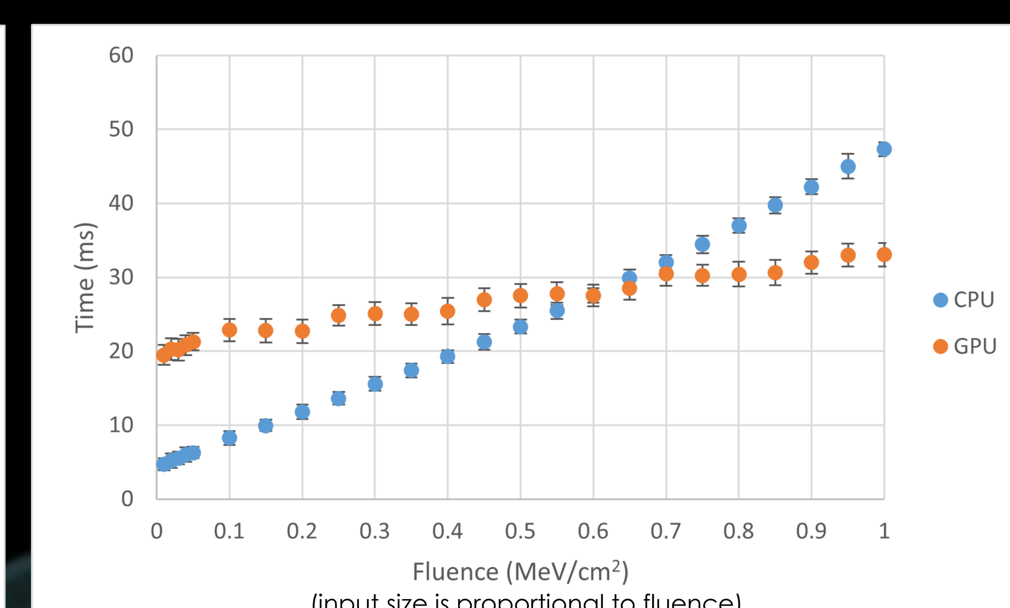
Initial Source Approximation



Execution times increase rapidly with fluence until the number of annuli exceeds 1000, after which we sample a constant-size subset.

GPU acceleration demonstrates an estimated speedup of 3-3.5x for initial source approximation

Iterative Refinement



GPU acceleration of iterative refinement slows the increase of execution time with input size. Execution switches from CPU to GPU for each iteration, incurring overhead.

References

[1] James Buckley. 2021. The Advanced Particle-astrophysics Telescope (APT) Project Status. In *Proceedings of 37th International Cosmic Ray Conference — PoS(ICRC2021)*, Vol. 395. 655.

[2] Marion Sudvarg, Jeremy Buhler, James Buckley, Wenlei Chen, et al. A Fast GRB Source Localization Pipeline for the Advanced Particle-astrophysics Telescope. In *Proceedings of 37th International Cosmic Ray Conference — PoS(ICRC2021)*, Vol. 395. 588.

[3] Wenlei Chen, James Buckley, S. Alnussirat, et al. 2021. The Advanced Particle-astrophysics Telescope: Simulation of the Instrument Performance for Gamma-Ray Detection. In *Proceedings of 37th International Cosmic Ray Conference — PoS(ICRC2021)*, Vol. 395. 590.

[4] W. Gander, G. H. Golub, and U. Von Matt. 1989. A constrained eigenvalue problem. *Linear Algebra Appl.* 114-115 (1989), 815–839.

[5] Gaël Guennebaud, Benoît Jacob, et al. 2010. Eigen v3. http://eigen.tuxfamily.org

Acknowledgements

This work was made possible through current and ongoing support from NASA grant 80NSSC19K0625, NASA APRA award 20-APRA20-0148, and NSF award CNS-1763503. The collaboration also acknowledges generous ongoing support from the Washington University McDonnell Center for the Space Sciences and the Peggy and Steve Fossett Foundation. We are grateful to the Washington University technical staff including Richard Bose, Dana Braun and Garry Simburger who made invaluable contributions to the development and construction of the prototype detectors and the Antarctic flight of the APTlite instrument.