

Hall D Computing

A Report for the 2024 Jefferson Lab Computing Review

The GlueX Collaboration

January 26, 2024

1. Introduction (Alex)

The GlueX Experiment in Hall D has been designed to search for quark-anti-quark states or mesons in which the gluonic field binding the system contributes directly to the quantum numbers of the states. In order to do this very high statistics are needed leading to multi-petabyte sized data sets. In order to reliably extract physics from such data sets, the software and production of all data are managed centrally by the collaboration with the primary goal being to provide consistent reconstructed and simulated data in manageably sized data sets for physics analysis. Analyzers' interaction with the large data sets and Monte Carlo generation are principally through web-based interfaces, ensuring that all compute-intense activities are set up and run in a consistent fashion, leaving very little chance for error. This document gives an overview of all the processes and procedures that are involved in this production, as well as ongoing efforts to improve procedures through the use of AI and machine learning. It also projects future computing needs.

2. Hall D Online Skim System (Sergey)

The first phase of GlueX was successfully completed in 2018 where more than 3.5 PB was acquired with DAQ system data rates of about 400 MB/s. For GlueX Phase II, the data rate more than doubled to approximately 1.25 GB/s. This stressed the original system developed under Phase I which consisted of a single output stream written to a large capacity RAID disk server. While technically within specs for the individual components, the DAQ system exhibited instabilities when pushed to these higher rates. This motivated changes to the system to ensure stable high-intensity running. Specifically the raw data files would need to be distributed among several RAID servers in order to reduce the average rate any one server needed to support. Another issue that came up while processing the Phase I data was the considerable effort required to extract special calibration events from the stored data files. Calibration events were typically made from special triggers for things like LED flashers used by the calorimeters. The calibration events were mixed into the single output stream and were rare (less than 1%)

compared to physics events. The DAQ system implementation for GlueX could not be easily changed to write separate output streams for these events directly. Thus they needed to be extracted from the full raw data set starting from tape producing skim files. An ability to generate these skim files in the counting house before the raw data ever made it to tape would save considerable time and effort. Implementation was done using a separate, new system, the Hall-D Online Skim System (HOSS). Because HOSS needed to transfer a copy of the entire 1.25 GB/s data stream to a small compute farm in the counting house, it also became a natural way to distribute the raw data files among several RAID server partitions, reducing the I/O requirements for each partition. This is illustrated in Fig. 1.

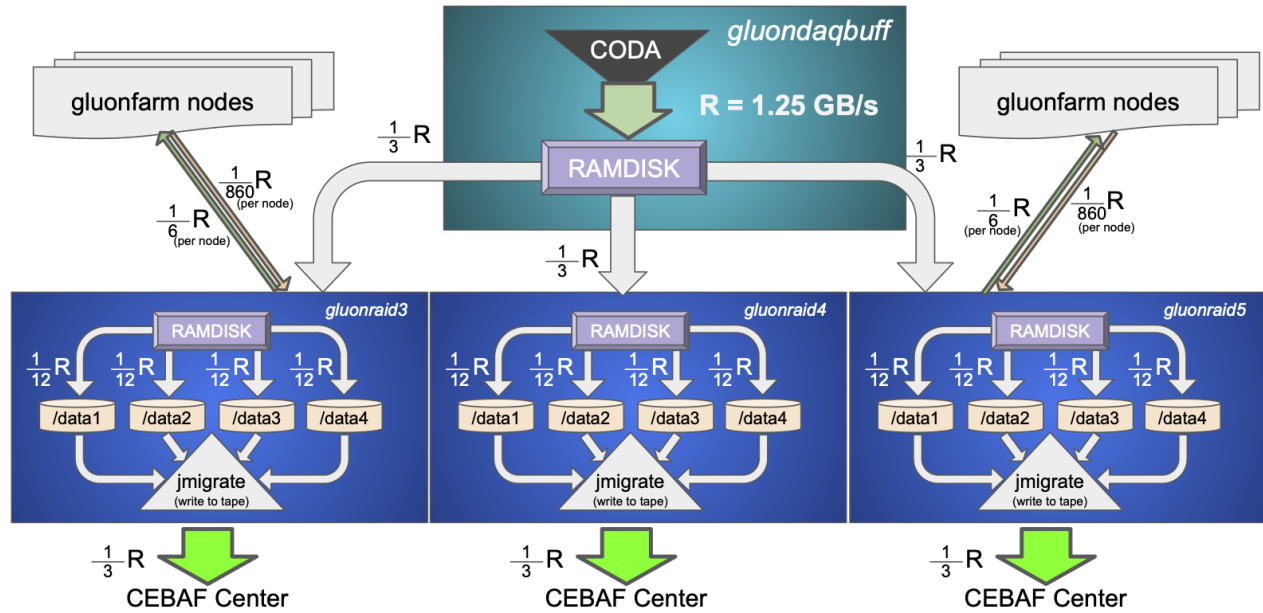


Figure 1: Illustration of how HOSS is configured for GlueX Phase II high-intensity running. CODA is the DAQ system that is configured to write data to a RAM disk. HOSS watches specific directories for files without open file descriptors and then moves them through the system.

The key orchestrator of HOSS is written in Python, but it relies on some key pieces of software to do the high-speed network transfers and CPU-intensive computations. RDMA is used over a 40-56 Gbps infiniband network fabric in the counting house. Custom RDMA servers written in C++ for Hall D are run as system services on almost all of the nodes in the GlueX online cluster. A custom tool was also developed that skims just the header of the 40-event blocks produced by the data acquisition to check if any of the events in the block is a calibration trigger that must be written out to a skim file. This avoids having to do computationally expensive dis-entanglement on every 40-event block. This savings in compute load allows HOSS to consume the entire data stream with a modest complement of 6 older compute nodes (Intel circa 2013). While scanning the headers for event type information, HOSS also records statistics for each data file: counts of each type of trigger as well as starting and ending event numbers. This information is written into a database on a MySQL server. It can be accessed either programmatically or via the web[1]. More details of the HOSS system can be found in Ref. 2.

The planned increase in the luminosity of the GlueX experiment also leads to an increase in data flow. To prepare for the increase in data traffic, we have expanded our computer farm in Hall-D - from 500 cores by another 1400 cores.

New GPU server (gluon201) for L3 trigger test, processing detector data

Also, to keep the data under control, the option of using a Level-3 trigger based on machine learning and neural networks is being considered. This will filter out the background and reduce the amount of data recorded. The data from the first level trigger will be sent to a heterogeneous system consisting of a computer farm, a GPU and an FPGA. Evaluation of an ML-FPGA based prototype for track reconstruction in FDC and calorimeter clusters has 5-10 μ s latency. This should allow us to pre-process data from Level 1 trigger with rates up to 100 kHz. We can reduce the amount of data to tape, which will allow faster reprocessing/ reconstruction. Or we can lower the L1 trigger thresholds to have more physics.

“Development of ML FPGA Filter for Particle Identification and Tracking in Real Time”

IEEE Transactions on Nuclear Science, 2023 | Journal article

DOI: 10.1109/TNS.2023.3259436

“ML on FPGA for tracking and PID”

Streaming readout Workshop SRO-XI

https://indico.bnl.gov/event/20010/contributions/79168/attachments/51353/87814/MLFPGA_Streaming_XI_2023_12_03.pdf

3. Calibration (Sean)

Prompt and efficient data calibration and validation is crucial in reducing the time between collecting the data and its availability for physics analysis. There has been a continuous focus on improving the stability of the readout firmware and in improving and automating calibration procedures, which has dramatically decreased the required calibration time. Generally, calibrations for well-understood experimental conditions, such as those for the nominal GlueX experiment and the PrimEx- η , are stable and allow for the reconstruction of important physics signals online or within hours of recording the data. Procedures for new experiments, such as the Charged/Neutral Pion Polarizability experiment, can take longer to develop depending on the number of changes to the experimental setup.

There have been several recent improvements in calibration procedures recently. These include the full automation of the energy calibration procedures for both calorimeters, and recent developments in improved web-based data quality monitoring tools. The AI-based control of the CDC reduced the processing time required for the calibration of that detector by roughly a factor 4. Additional processing has recently been added to the online computing farm in anticipation of

the 2024-25 GlueX-II/JEF run that is planned to be sufficient for the creation of all skims required for calibration work. This will minimize our offline processing requirements while the experiment is operating, and will allow calibration processes that require these skims to run sooner. Overall, we are making small and steady improvements in our procedures to promptly calibrate detectors and produce production-ready data for this next run, once the new FCAL insert is fully commissioned.

4. Reconstruction (Igal)

As is often the case, reconstruction of the raw data is by far the most compute intensive activity that we perform. For the past few years we have performed reconstruction not only on the JLab farm, but also at several High-Performance Computing (HPC) sites. Successful applications for allocations have been made to NERSC and the Pittsburgh Supercomputing Center. Access to the Indiana University BigRed3 and BigRed200 supercomputers is a collaboration contribution and no application is needed. Table 1 gives statistics for each of these facilities.

At JLab the Scientific Computing group has developed a workflow manager SWIF that has been in operation for many years. It allows users to deal with job submission en masse, including efficient retrieval of input data from the Tape Library, with a built-in database keeping track of individual job progress. Processing campaigns or “launches” have used a set of by-now-legacy Python scripts to manage submissions to the SWIF2 system (both for “Reconstruction” and “Analysis” launches).

For the HPC sites, job submission and management of input and output data is handled by a SWIF2 and Globus

Run period	Length of processing	Fraction of events at each site	CPU used at each site in millions of core-hours	Number of jobs
2017-01	1 Month	JLab (100%)	6	42165
2018-01	4 Months 1 Month	NERSC (81%) JLab (19%)	20.4 2.3	77603 16279
2018-08	2 Months 1 Month 1 Month	NERSC (52%) PSC (Bridges) (26%) JLab (22%)	9.25 0.81 2.1	24669 6990 13358
2019-01	2 Weeks	NERSC (100%)	0.12	1220

2019-11	4 Months 2 Months 2 Months 4 Months 4 Months	NERSC PSC (Bridges) PSC (Bridges2) BigRed3 JLab (57%)	12.4 2.33 4.03 3.56 23.9	45236 17752 19694 16392 119397
2021-08	1 Month	NERSC (100%)	0.15	13838
2021-11	2 Months	NERSC (49.5%) JLab (50.5%)	0.32 0.72	28021 28672
2023-01	1 Month	NERSC (100%)	0.04	7033

Table 1: Reconstruction processing.

5. Analysis (Alex)

The full set of reconstructed data files (Reconstructed Event Summary Tape or REST files) is stored on tape and too large to be easily handled by individual analyzers. See Table 2. In order to reduce the size of the data sets accessed by analyzers, a central system was developed to process the REST data at JLab and extract reaction-specific ROOT trees.

Run period	# REST files	Size REST files [TB]	# Analysis Launches	# Channels	Σ Tree Size [TB]
Spring 2017	42k	117	51	1955	700
Spring 2018	84k	377	19	360	700
Fall 2018	48k	217	17	399	500
Spring 2020	210k	1,112	2	31	

Table 2: GlueX run periods and Analysis Launches

Users can request ROOT trees for reactions of interest via a web interface, shown in Fig. 2. Periodically, the submitted reactions are collected into a configuration file, which controls a workflow that produces all of the trees, an “Analysis Launch.” For each reaction, the GlueX

analysis library inside the JANA framework creates possible particle combinations from the reconstructed particle tracks and showers saved in the REST files. Standard selection criteria are applied for exclusivity and particle identification before performing a kinematic fit, which imposes vertex and four-momentum constraints. Displaced vertices and inclusive reactions are also supported. Objects representing successful particle combinations (e.g. $\pi^0 \rightarrow \gamma\gamma$) and other objects are managed in memory pools, and can be reused by different channels to reduce the overall memory footprint of the process. With this scheme, up to one hundred different reactions can be combined into one analysis launch, processing the reconstructed data on multiple cores in parallel without large memory overhead.

Please fill out your reaction below:

Use add/remove particle to add/remove a particle from the products side of the reaction.
Each product comes as a set of three objects:
1) the main selector where you can select the product.
2) a tri-state button to let you flag the particle as "m" (missing) or "M" (NOT Mass constrained) as desired.
3) a checkbox to indicate the product decays

B (Beam Bunches): 3 T (Extra Charged Tracks): 3 F (Fit Type): P4 and Vertex U (unused tracks): ☐

Initial Particles ----> Final State Particles

γ p $\rightarrow$ η ☒ M π^0 ☐ p ☐

LEVEL 1

$\eta \rightarrow \pi^+$ ☐ π^- ☐ π^0 ☐

```
Reaction1: 1 14 7 17 14
Reaction1: Decay1 17 7 8 9
Reaction1: Flags 83_M17
```

Figure 2: Web Interface for Analysis Launches.

If the kinematic fit converges for one combination of tracks and showers, the event is stored into a reaction-specific but generic ROOT tree. The size of the resulting ROOT trees strongly depends on the selected reaction. ROOT trees (about 200 per run) are merged into a single file whose size is suitable for copying a user's home institution for a physics analysis. For a given run period, a new version of REST production or global changes to the selection criteria require that Analysis Launches be repeated for the new conditions. The total number of channels in Table 2 may therefore include multiple versions of the same reaction.

With nominal availability of JLab farm nodes, a typical analysis launch can be completed in one to two weeks. The elapsed time is limited by the latency due to retrieval of the REST data from the Tape Library. In the future, staging of files on SSD or on a distributed file system will help throughput.

6. Simulation (Peter)

Simulations of the detector response are required in order to study the feasibility of measurements or apply corrections to data. The simulation of a typical reaction is split up into

independent steps. The general flow consists of (1) event generation, (2) detector simulation where interaction of produced particles with detector elements is simulated, (3) smearing or addition of detector resolution and efficiency, (4) reconstruction, and (5) data analysis. The latter two steps are performed with the same reconstruction and analysis code as that used on real data.

6.1. Simulation Components

Event generation. A variety of event generators have been developed for different needs.

- **bggen** produces minimum-bias hadronic photoproduction events. It is based on a custom version of Pythia for high-energy photons and a compilation of known reactions for photon energies below 3 GeV.
- **genr8** produces events from a user-defined decay tree of hadronic resonances according to 2-body and 3-body phase space for a fixed photon beam energy.
- **gen_amp** is a collection of reaction-specific t-channel photoproduction generators. Samples are weighted by a user-selected set of partial-wave amplitudes.
- The **photon beam source** models the coherent bremsstrahlung process at the diamond radiator.
- The **beam conversion source** models pair+triplet production in the polarimeter target.
- The **Bethe-Heitler source** models e^+e^- and $\mu^+\mu^-$ pair conversion in various types of GlueX targets.

Detector Simulation. The original hdgeant simulation based on the CERNLIB GEANT3 library has been superseded by hdgeant4 based on the Geant4 toolkit. Both programs utilize the same abstract geometry description and magnetic field maps, can read events from the same generators, and produce output events in the same format. The ability to directly compare the outputs from the two simulations has been very helpful throughout the transition period.

Smearing. The mcsmeas program reads in lists of raw hits from hdgeant4, and applies a set of transformations to them, designed to imitate the detector response when passing particles deposit energy in sensitive elements. Run-dependent parameters describing the resolution functions and the efficiencies are stored in a database. Significant progress has been made in improving and calibrating these parameterizations over the past years. mcsmeas also overlays accidental detector hits on top of the pattern of hits from hdgeant4. This is done by including hits from a set of random triggers obtained with each run. Simulated data is produced on a run-by-run basis so that the prevalence of accidental hits matches that of the real data.

6.2. Simulation with MCwrapper

To help users perform these five steps in an efficient and accurate manner, the tool MCwrapper[4] was developed to manage the entire chain. It is controlled with a configuration file in which the user specifies software parameters such as package versions, or experimental parameters such as beam energy or polarization. MCwrapper can also query the run condition database to pull information that may vary on a run-by-run basis. MCwrapper can be invoked from the command line, but the recommended method is to use a web-based submission form. Fig. 3 shows a screenshot of the form. Here the user only has to choose default settings for a given beam time and provide a path to a configuration file used by the event generator. This minimizes the room for errors even further. Alternatively, users can make choices from dropdown menus and configure a MCwrapper project that way. Projects submitted via the webform are first checked against a database to make users aware of other projects using the same configurations to avoid duplication of effort. Projects undergo automated small-scale testing and upon passing, jobs are submitted to the Open Science Grid (OSG). As of January 18, 2024, 77 unique users have submitted 2,958 projects, which ran more than 5.1 million jobs and produced about 95 billion events. The total CPU time used for successful projects is about 24 million core hours.

Experiment

GlueX
CPP
JEF

Beam time

2017-01 (ver03)

2017-01 (ver04)

2018-01

2018-08

2018-08 lowE

2019-11
custom

Name
Email

halld_recon version:
halld_sim version:
version Set:

Run Number of Events

Output Directory Name

Generator

Full Path to Generator Config

Flux to Generate: ☒ ccdB ☐ cohrens

Min Photon E: Max Photon E:

Post-Processing:

Geant Version: ☐ Geant3 ☒ Geant4

Geant Secondaries? ☒

recon-2019_11-ver01_10.xml

- created: 2023-10-13
- description: Recon-launch-compatible, based on version set 5.12.0.

Package	Version	Directory Tag	Debug Level
amptools	0.15.1	root60806	
ccdb	1.06.07		
cernlib	2005		
diracxx	2.0.1		
evio	4.4.6		
evtigen	01.07.00		
geant4	10.04.p02		
glueX_MCwrapper	v2.7.0		
glueX_root_analysis	1.25.0	rec191113	
halld_recon	recon-2019_11-ver01.3		
halld_sim	4.46.0	rec191113	
hdds	4.10.0	x323	
hdgeant4	2.36.0	rec191113	
hd_utilities	1.46		
hepmc	2.06.10		
jana	0.7.9p1	x323	
lapack	3.6.0		
photos	3.61		
rcdb	0.06.00		
root	6.08.06	bs221	
sqlitecpp	2.2.0	bs130	
sqlite	3.13.0	bs130	
xerces-c	3.2.3		

Figure 3: Screenshot of MCwrapper submission web interface.

7. Artificial Intelligence and Machine Learning (Naomi)

Hall D, the JLab Data Science Department, the Artificial Intelligence for the Electron Ion Collider Group, and Universities (Indiana & Massachusetts) have pursued several projects in the area of artificial intelligence (AI) and machine learning (ML).

7.1. Online systems

RoboCDC (EPSCI + Naomi Jarvis, CMU)

RoboCDC paper <https://10.1088/1742-6596/2438/1/012132>

AI/ML has been incorporated into detector control for the GlueX Central Drift Chamber (CDC), using a model trained using previous calibration data and environmental measurements from EPICS to predict the CDC gain at its traditional operating HV, and altering the HV autonomously at the start of each run to compensate for changes in the environment and thereby keep the detector gain constant. This system is known as RoboCDC and was developed during 2021-2023; it is now in routine use.

Hydra (EPSCI + GlueX collaboration)

Hydra paper <https://doi.org/10.1051/epjconf/202125104010>

The AI/ML software Hydra has been used in Hall D during data collection since 2019, using image classification models for data quality monitoring to alert the shift-takers promptly when a potential issue has been identified. Hydra is able to check the online monitoring data more efficiently than the shift-takers because it can inspect images many times faster, and can detect intermittent problems. Hydra has also been deployed in Halls A, B and C more recently.

On-board processing with FPGAs (Sergey + JLab EIC)

FPGA ML paper: https://misportal.jlab.org/ul/publications/view_pub.cfm?pub_id=16832

Work is ongoing using prototype EIC detectors to refine ML algorithms implemented on FPGA chips for particle identification and shower clustering, aiming to identify the useful detector hits and reduce the volume of data recorded.

AI Optimized Polarization (EPSCI lead, JLab, W&M, CMU)

Staff have been hired already, we are literally waiting for the funding.

<https://science.osti.gov/-/media/funding/pdf/Awards-Lists/2875-NP-AI-ML-Awards-List.pdf>

The proposal: https://wiki.jlab.org/epsciwiki/images/4/45/FOA_002875_AIEC_v0.3.pdf

A multi-institutional team led by the EPSCI group has been awarded DOE funding to develop AI/ML models and control software to optimize polarization of beams and targets used across the

experimental Halls. In Hall D, the software will adjust the position and orientation of the diamond radiator for optimal photon beam polarization. Work on this will start as soon as the funds have arrived.

7.2. Offline systems

FCAL shower classification

FCAL Shower Classification ML paper: <https://doi.org/10.1088/1748-0221/15/05/P05021>

A ML algorithm has been implemented in the GlueX reconstruction software to differentiate between low energy photons and split-offs occurring from hadronic interactions in the detector.

Pion identification for CPP

https://halldweb.jlab.org/DocDB/0060/006060/002/BetheHeitlerPhysics_AndrewSchick_GlueXColab_May2023.pdf

https://halldweb.jlab.org/DocDB/0060/006061/001/identifying_muon_pairs.pdf

Analysis of data from Hall D's charged pion polarizability (CPP) experiment requires the ability to separate pion tracks from Bethe-Heitler (BH) muon and electron/positron tracks. Two ML neural nets (one for each charge state) have been developed and incorporated in the GlueX reconstruction software to separate pions from BH electrons/positrons, using data from FCAL and the FDC. A third neural net is under development to separate pions from BH muons, using data from the MWPCs (multi-wire proportional counters), FDC and CTOF.

7.5. Charged-pion polarizability experiment and Bethe-Heitler pair identification ()

The charged-pion polarizability experiment in Hall D is a precision measurement of low-energy QCD using pion pairs produced via the Primakoff effect. Making such a measurement relies on the ability to distinguish pions produced at threshold from Bethe-Heitler muon and electron pairs. Separating muons from pions is significantly more difficult than separating pions from electrons. The primary difficulty is that muon, like pions, generally do not produce electromagnetic showers forward calorimeter (FCAL). Construction of six multi-wire proportional chambers (MWPCs), to be deployed between several layers of steel absorbers, has been completed to assist in π/μ separation, and a neural net that combines information from the forward drift chambers, calorimeters, and the MWPCs is under development.

The simpler problem of identifying Bethe-Heitler electron pairs has been studied with already-taken GlueX data. A multi-layer perceptron neural net was trained using ROOT's TMVA package using three features from the forward calorimeter and drift chambers—the ratio of FCAL energy to charged track momentum, the E9E25 shower ratio (the ratio of energy deposited in a 3x3 block grid to that for a 5x5 block grid), and the distance between the charged track projection

and the centroid of the calorimeter shower. The electrons were trained on simulated data and pions on real data from the ρ^0 peak. Fig. 7 shows results from the study.

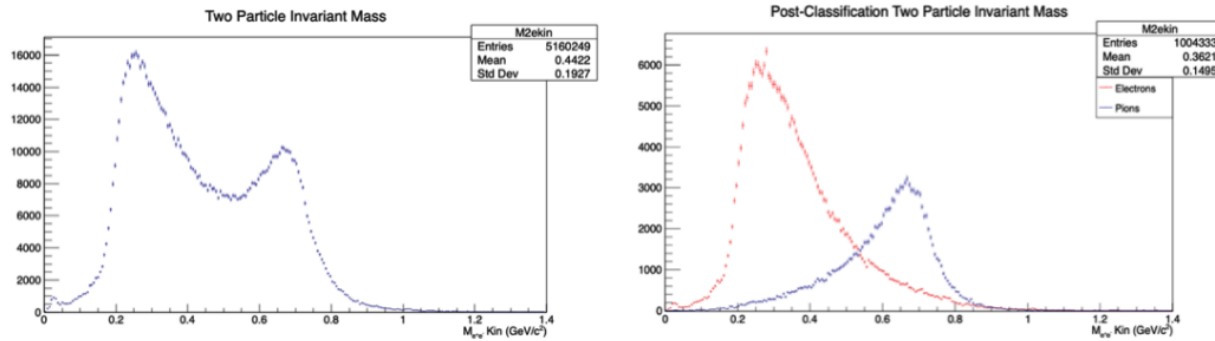


Figure 8: Left: input GlueX data containing both electrons and pions prior to classification. Right: the neural net applied to the input data twice, first selecting for pions (blue), then electrons (red).

7.7. Physics-Informed Neural Network (Daniel/Igal)

A model has been developed for Hall D physics analysis, specifically for experiments like PrimEx-eta and JEF. It includes a custom loss function and corresponding machinery, enabling a detailed study of the model and facilitating the determination of systematic errors associated with its usage.

8. GPU resources for amplitude analysis (Alex)

The final step in the GlueX analysis chain often involves a single analyst performing an amplitude analysis on a data set. Amplitude analysis involves an unbinned multi-dimensional likelihood fit to the data set and has, for decades, been the standard technique for extracting resonance properties from data. Very roughly the computing cost of a fit is given by the product of the number of events being fit and the complexity of the model. The large GlueX data set and sophisticated phenomenological models developed by the Joint Physics Analysis Center (JPAC) drive both terms in this product. The problem is ideal for parallel computing on GPUs, and the collaboration is currently using the AmpTools library, which initially supported NVIDIA GPU-accelerated fitting about ten years ago and has undergone many iterations of improvement and optimization in the past decade. While AmpTools has methods to optimize memory use and

also distribute a single fit across multiple GPUs (even on different nodes via MPI), the limitation one often runs into is memory. If all of the data needed to perform the unbinned fit can't be loaded into GPU memory, then GPU acceleration is not a viable option. The new NVIDIA A100 and V100 GPUs, which are also effectively deployed for machine learning applications, provide up to an order of magnitude more memory than previous generations of GPU and are ideal for using computationally complex models to fit the large GlueX data sets.

Our experience is that it is relatively easy for a single analyst to saturate the available GPU resources on the GPU enabled nodes on the JLab SciComp cluster (3 with TitanRTX cards and 3 with Tesla T4 cards = 44 GPU cards in total). For a typical analysis, a standard workflow requires multi-dimensional binning resulting in ~100 independent fits, each running for several hours on a single GPU to fit a given model. With hundreds of possible models to fit and many analyses being performed in parallel the existing resources will soon be oversubscribed, given that their usage for machine learning applications are also growing rapidly. In addition the cost of the high-memory GPUs that are ideal for amplitude analysis prohibits many institutions from making an investment in this hardware. The collaboration would benefit from an enhanced pool of state-of-the-art high-memory NVIDIA GPUs that could be shared with other activities at the lab that can exploit this computing architecture.

9. Reconstruction on the Open Science Grid (move to 4)

A demonstration system has been developed, deployed and tested to do GlueX event reconstruction on the OSG. Each 20 GB raw data file is split into 60 to 70 small files and a single OSG reconstruction job is run against each one. This allows us to run single-threaded jobs taking 2 to 3 hours, opening access to opportunistic resources. Results are copied back to a local host and merged to produce one output file per input raw data file. A PostgreSQL database is used to keep track of all the partial files. We hope to roll out the system, at scale, in the coming year.

10. Areas for Improvement (All)

There are several areas in which we would like to do better.

- **Data Catalog.** Our workflows, in total, produce millions of files. We would benefit from a global data catalog not only to keep track of what files we have and where they are, but also what files that we expect to be produced have in fact *been* produced. Another desirable feature would be to validate files, according to some user-defined criteria, as they are produced and record results of the validation. Many of the workflow managers that we use have databases underpinning their work, but those are not instrumented for direct user access to facilitate custom queries and are generally aimed at tracking jobs

and not files. We are particularly interested in pursuing solutions that leverage work by other collaborations/labs and adapting them to our needs.

- **Work Flow Management.** We need a mechanism to couple work flow management systems more tightly to any future data catalog. Lack of coupling defeats many of the advantages of a data catalog.
- **Continuous Integration (CI).** We have a system for CI but the tests are limited in scope.
- **Comprehensive Testing.** Global testing of reconstruction and simulation is done, but there is not a good way to track changes in performance over time.
- **Unit Testing.** We do very little unit testing and have not developed a paradigm for implementation.
- **Documentation.** We have recently focused some resources on this area. New collaborators have complained that documentation is hard to find and often out of date, among other age-old problems. We have recently formed a documentation task force to take a comprehensive look at how we can improve in this area.

11. Conclusions

It has been a busy period for GlueX since the last Computing Review three years ago. All stages of the scientific enterprise from data taking, through data analysis, and publication of results are now in full flight. Many lessons have been learned and areas of improvement have been identified. There are also many ideas for future developments, some of which have been pioneered by other experiments, others that are more speculative. All of these endeavors would benefit from more human resources deployed at the interface of physics data analysis and software engineering.

12. Summary of Computing Requirements

References

1. HOSS webpage: <https://halldweb.jlab.org/hoss>
2. HOSS paper: <https://doi.org/10.1051/epjconf/202125104005>
3. RoboCDC paper <https://10.1088/1742-6596/2438/1/012132>
4. FCAL Shower Classification ML paper: <https://doi.org/10.1088/1748-0221/15/05/P05021>
5. FPGA ML paper: https://misportal.jlab.org/ul/publications/view_pub.cfm?pub_id=16832
6. MCwrapper paper: <https://doi.org/10.1051/epjconf/202024503028>

7. Hydra paper: <https://doi.org/10.1051/epjconf/202125104010>