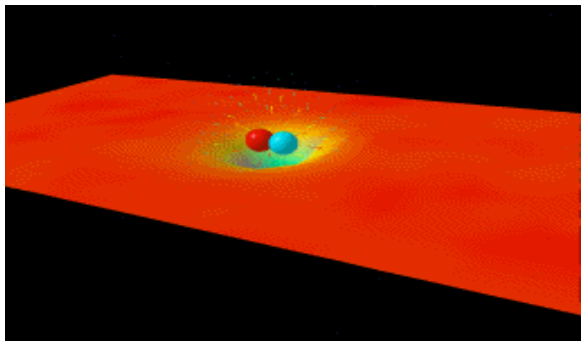
Present and Future Computing Requirements for Jlab@12GeV Physics Experiments]

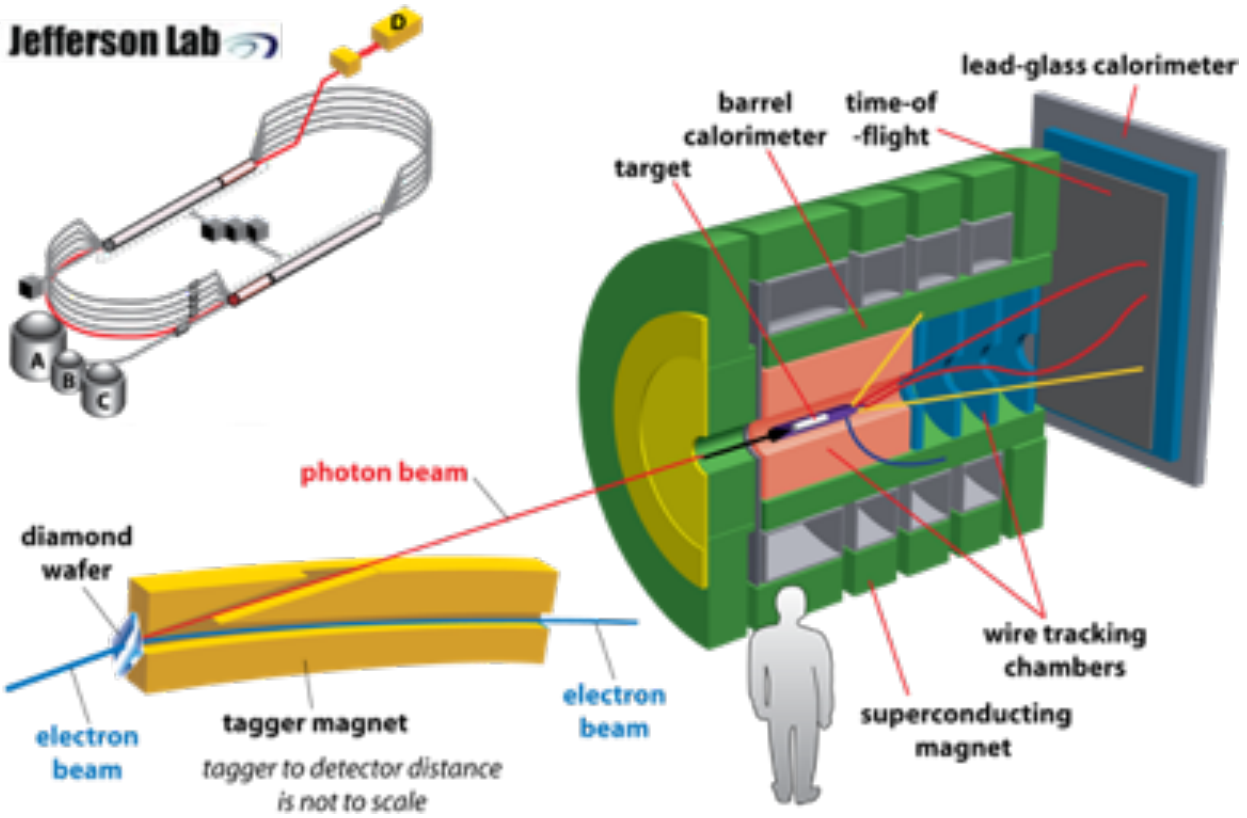
Richard Jones
GlueX

1. JLab@12GeV Experiments Case Study: Overview

Halls A, B (CLAS12), C, D (GlueX)

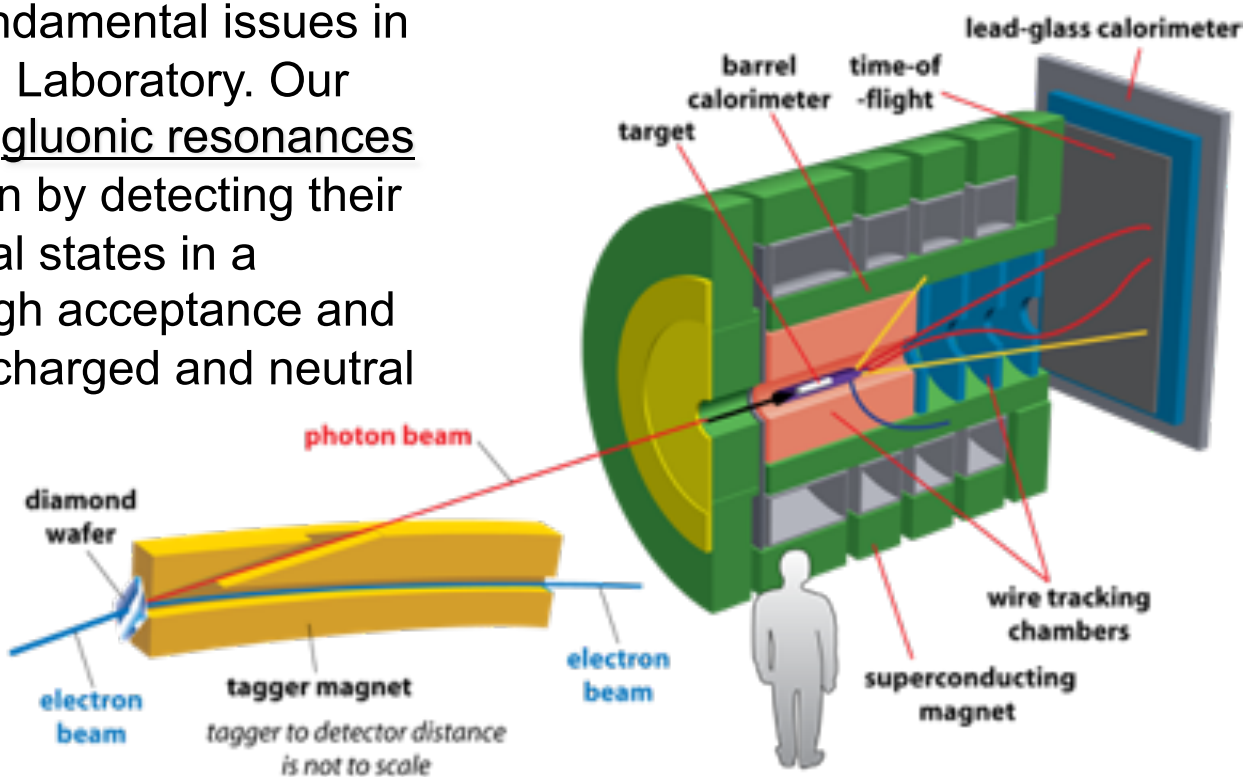
- Hall A – nucleon structure and form-factors, New Physics
- CLAS12 – parton distributions, hadronization, excited mesons
- Hall C – form factors, nucleon structure
- **GlueX – spectrum of mesons with gluonic excitations**
- Present focus is
  - — build the detectors
  - — debug the analysis
- In the next 3 years
  - — install
  - — commission
  - — be ready for data





1. GlueX – the science

The GlueX Collaboration is building a 12 GeV photon beam line and a dedicated solenoidal spectrometer to study fundamental issues in **strong QCD** at Jefferson Laboratory. Our primary aim is to identify gluonic resonances in meson photoproduction by detecting their decays into exclusive final states in a hermetic detector with high acceptance and good resolution for both charged and neutral particles.



Unambiguous discovery of a multiplet of hybrid mesons will provide answers to long-standing questions regarding how gluonic degrees of freedom are expressed in hadrons.

1. GlueX – the collaboration

15 institutions + Jlab
~60 members

Collab. Board (6)
Executive Committee
Current spokesperson
Curtis Meyer, CMU

Schedule:

- Sept. 2008: **CD3**
  start of construction
- Dec. 2012:
  end of 6 GeV Ops.
- 2015: **CD4**
  start of operations

- University of Athens
- Carnegie Mellon University
- Catholic University
- Christopher Newport University
- University of Connecticut
- Florida International University
- Florida State University
- University of Glasgow
- IHEP Protvino
- Indiana University
- Jefferson Lab
- U. of Massachusetts, Amherst
- North Carolina A&T State
- U. of North Carolina, Wilmington
- Santa Maria University
- University of Regina

2. Current HPC Methods

- Algorithms used, Codes, etc.
  - **Simulation**
  - **Reconstruction**
  - **Analysis**

2. Current HPC Methods

- Algorithms used, Codes, etc.

  - **Simulation**

    based on Geant3/4

    events are independent

    code is volatile during initial years => like a glass of beer

    results quickly go stale

    lots of re-generation

    limited need to archive simulations

    code more stable as time passes => like a bottle of wine

    volume demand increases

    more need to archive to satisfy demand

2. Current HPC Methods

- Algorithms used, Codes, etc.

  - **Reconstruction**

    custom code developed for each experiment

    reconstructs the path of particles through the detector

    events are independent

    some parallelism is helpful to conserve memory

      static information must be in memory

      shared memory can be used

      Gluex uses pthreads library

    CLAS12 reconstruction components are web services

      demanding portions of code hosted separately

      services can by dynamically provisioned

2. Current HPC Methods

- Algorithms used, Codes, etc.

  - **Analysis**

    reduces the reconstructed sample to a selected subset

    performs some useful transformation on the subset

    custom code developed by each group

    mostly based on ROOT

    single-threaded algorithms

    reduced data samples

    can be computationally demanding in special cases

    model fits to multi-dimensional distributions

    partial-wave analysis resonance decays

2. Current HPC Methods

- Quantities that affect the problem size or scale of the simulations (grid? particles? basis sets? Other?)

  - **Simulation**

    typical event size 10-100 kB

    production rate 2-3 events/s on a 2 GHz Nehalem core

    60% of total CPU time used for simulation

    scale set by how many real events collected

    6000 cores needed in steady state to keep up

    at maturity, simulation needs 2.5 PB/yr of storage

2. Current HPC Methods

- Quantities that affect the problem size or scale of the simulations (grid? particles? basis sets? Other?)

  - **Reconstruction**

    same code is used for both raw and simulated events

    multiple passes are needed – calibration

    only one pass through the entire sample

    cpu limited, mostly because of track finding, fitting

    30% of CPU time is devoted to reconstruction

    reconstructed event record 50% larger than raw

      option to save “cooked” data + raw (space factor 2.5)

      option to discard “uninteresting” events (space factor 1.2)

2. Current HPC Methods

- Quantities that affect the problem size or scale of the simulations (grid? particles? basis sets? Other?)

  - **Analysis** – *special case of partial-wave analysis*

    fits on a single set of data can take many hours on a single processor

    involves evaluation of many identical matrix operations

    literally begs to be parallelized

    several implementations of parallel PWA exist

      using the MPI library – speed-up factors of 100

      using the CUDA platform – speed-up factors of 1000's

      running on commodity NVIDIA hardware

    GlueX plans incorporate GPU's for PWA into analysis infrastructure

2. Current HPC Requirements

- Requirements are not really “HPC”, more like “HTC”
- Currently no data – no real requirements, except **get ready!**
- **Where we are in mid-2011:**

Facilities Used or Using	JLAB OLCF ACLF NSF Centers Other: Open Science Grid
Architectures Used or Using	Cray XT IBM Power BlueGene Linux Cluster GPUs Other:
Total Computational Hours Used per Year	1,000,000 Core-Hours in 2011, estimated
OSG,JLab ~~**NERSC**~~ **Hours Used in 2010**	50,000 Core-Hours **(OSG, not NERSC)**
Number of Cores Used in Typical Production Run	1000
Wallclock Hours of Single Typical Production Run	50 hours
Total Memory Used per Run	500 GB
Minimum Memory Required per Core	1 GB
Total Data Read & Written per Run	10,000 GB
Size of Checkpoint File(s)	0 GB **(not used)**
Amount of Data Moved In/Out of ~~**NERSC**~~ **OSG**	10 GB per job **(interpreted as grid storage)**
On-Line File Storage Required (For I/O from a Running Job)	10 GB and 10,000 Files
Off-Line Archival Storage Required	0 TB and 0 Files **(no need, currently)**

3. HPC Usage and Methods for the Next 3-5 Years

- Part of commissioning in 2014 is shaking down the Monte Carlo
- Physics Data taking expected to begin in 2015
- **Where we need to be in 2016:**

Computational Hours Required per Year	80 million
Anticipated Number of Cores to be Used in a Typical Production Run	10,000
Anticipated Wallclock to be Used in a Typical Production Run Using the Number of Cores Given Above	15 hours
Anticipated Total Memory Used per Run	5 million GB
Anticipated Minimum Memory Required per Core	2 GB
Anticipated total data read & written per run	100,000 GB
Anticipated size of checkpoint file(s)	0 GB
Anticipated Amount of Data Moved In/Out of ~~NERSC~~ SRM	10 GB per job
Anticipated On-Line File Storage Required (For I/O from a Running Job)	100 GB and 10,000 Files
Anticipated Off-Line Archival Storage Required	2.5 PB per year

3. HPC Usage and Methods for the Next 3-5 Years

- Upcoming changes to codes/methods/approaches to satisfy science goals

  - ❑ Simulation

    move from Geant3 (f77) to Geant4 (c++)

    algorithms remain fundamentally the same

    speculation about future porting of Geant to GPU's

      would have significant impact on GlueX

      serious impediments are seen

      more than 5 years away

    Present algorithms are adequate for the lifetime of the experiment, fit within a reasonable scope, especially when resources from member institutions are counted.

3. HPC Usage and Methods for the Next 3-5 Years

- Upcoming changes to codes/methods/approaches to satisfy science goals

  - Reconstruction

    algorithms are still evolving

      techniques for tracking, clustering are well established

      search is for mix of algorithms that works best

    compared to other solenoid experiments, tracking is slow

      exploit of SIMD instructions tried, results limited

      new super-scalar extensions (AVX) could improve it

      bigger gains probably in algorithm refinements

    pthreads-based parallelization framework is stable

    development is proceeding based on simulated events

3. HPC Usage and Methods for the Next 3-5 Years

- Upcoming changes to codes/methods/approaches to satisfy science goals

  - ❑ Analysis
    active area of current effort by several groups
    progress impeded by issues with reconstruction

This is where the development of novel algorithms and working modes must occur – hopefully within the next 5 years – if the experiment is to achieve its scientific goals.

Use of GPU's for partial-wave analysis has demonstrated results.
Must be scaled up into a full analysis application
application runs interactively on a desktop
application is integrated with remote data services
remote compute services offload intensive aspects to maintain interactive response

3. HPC Usage and Methods for the Next 3-5 Years

- Upcoming changes to codes/methods/approaches to satisfy science goals

PWA is just one example of interactive access services to large scientific data sets being filtered through a user-supplied filter.

data sets are smaller than the indexable internet

~5 XB (2 years ago)

but

queries are more complex than "find *Will* next to *Rogers*"

however

latency tolerance for "interactive use" >> 0.2 s

3. HPC Usage and Methods for the Next 3-5 Years

- Upcoming changes to codes/methods/approaches to satisfy science goals

Such services have started to appear – for physics analysis

PROOF service at CERN (AliProof service, ALICE experiment)

large data store tightly coupled to interactive analysis cluster

users connect through standard root application

user code is same as used for local analysis

commands ("queries") are run in parallel in real time

results are returned in real time to user

Provision of such a service marks a significant shift from traditional scheduling paradigms for shared compute resources.

Introduces a QOS component to resource allocation and accounting, goes beyond the total cpu use and queue priorities of batch systems.

Strategy for New Architectures

- Further advances in the common HPC benchmarks (more FLOPS/processor, network throughput and latency) do not impact the major production workflows of GlueX very much – but they may affect cost of hardware.
- PWA is one outstanding exception, GPU's are an excellent solution.
- *Assembling all of the pieces into a functional analysis platform for an interactive user doing a PWA study has further to go.*
- *Implementation of PWA algorithms on GPU hardware is still in its infancy.*
- *All work up to this moment has been done on the CUDA platform*
- *All of the usual bottlenecks in on GPU's affect us*

memory and GPU memory

limited bandwidth between main

double precision

cost of doing computations in

Strategy for New Architectures

- **The grid platform provides excellent opportunities to better harness resources at member institutions**
  - the Virtual Data Toolkit (VDT)
  - the grid user community and support infrastructure
  - naturally fits with collaborative framework
  - provides tools for effectively managing a shared resource
  - increases the usefulness of the resource for science
  - leverages times of reduced local demand for outside users
  - takes advantage of reciprocal behavior
  - leverages the investment of HEP in grids
  - new opportunities for collaboration
- Two+ years of experience with operating UConn site on the OSG

4. Summary

- Sufficient resources are included in the JLab computing plan to satisfy the minimum requirements for all of the 12GeV experiments, including GlueX.
- What more could we do if we had 5X as many resources?

the experiment is expected to be systematics-limited
systematic errors are often limited by Monte Carlo simulation

more Monte Carlo studies

- Ease of use and response time is more likely to be the limiting factor in terms of Monte Carlo studies than available hardware.
- Significant compute facilities at user institutions organized into a grid offers significant flexibility in designing analysis solutions.

Backup Slides

(from Matt Shepherd, IU, GlueX Collaboration, Jan. 2010)

A very very rough estimate of the scale of the problem

Experiment	N_{data}	N_{amps}
E852	1×10^6	30
CLEO	1×10^5	10
BES III	1×10^7	20
GlueX	1×10^7	30

10,000 CPU hours/analysis simple theoretical model

doable with approx. 100 2003 CPUs

- Fit time scales linearly with statistics and like the square of N_{amps}
- Floating parameters in amplitudes add 1-2 orders of magnitude to time
- More realistic theoretical models add an additional 1-n orders of magnitude to fit time
- *To do a first GlueX analysis, we need 10x faster machinery than E852*
  (done now! ...faster multi-core CPUs)
- *To do pioneering, high-statistics meson spectroscopy we want 3-4+ orders of magnitude speed gain*
  - Grid model: Can we really use 1,000-10,000 CPUs at one time in a fit?
  - GPUs: Inexpensive way to enhance speed of single box by two orders of magnitude

Parallel Computing

- This type of problem is perfect for parallel computing since all of the large sums over can be done in parts

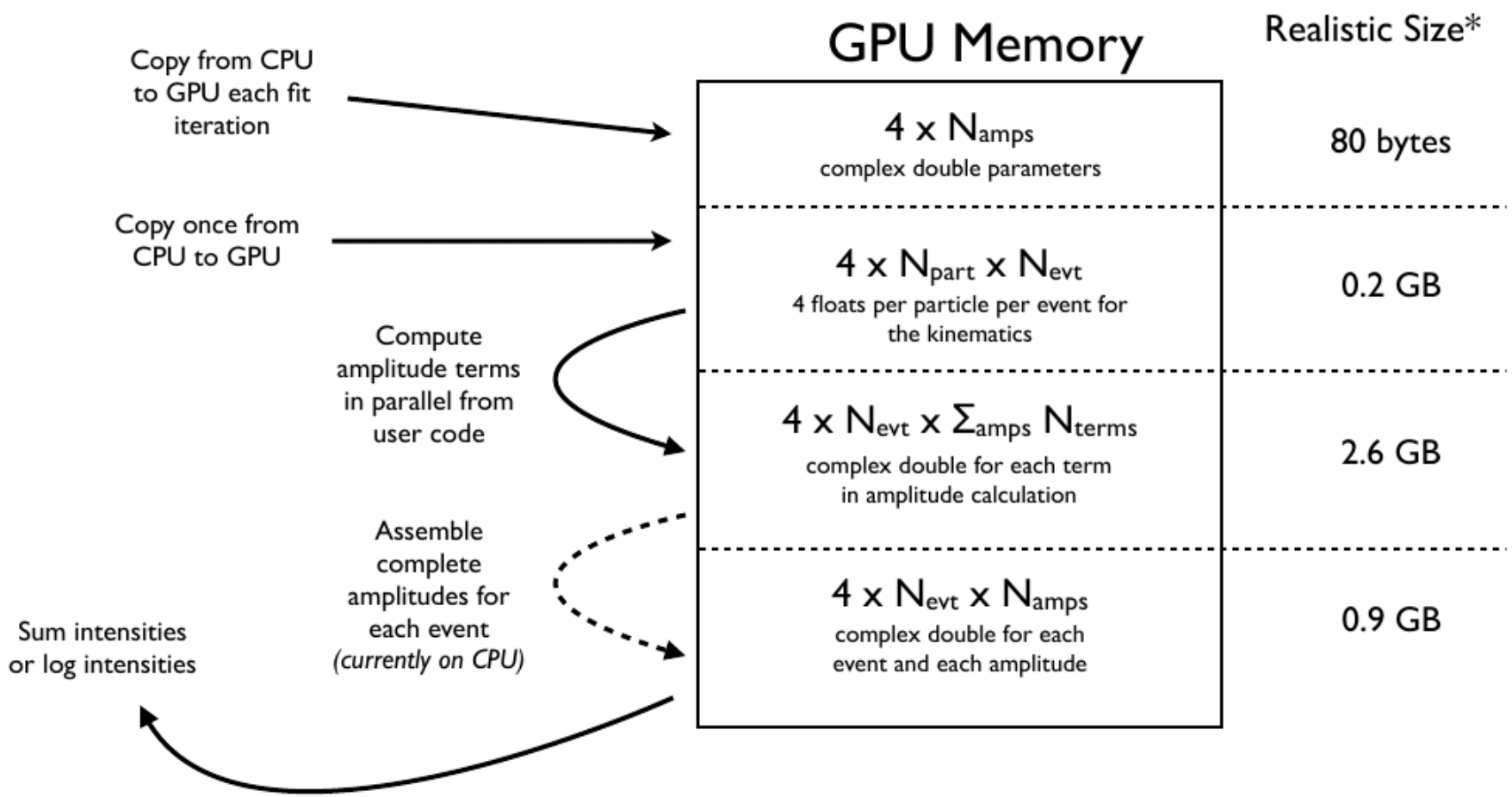
$$\ln \mathcal{L} = \sum_{i=1}^{N} \ln \left(\sum_{\alpha,\beta}^{N_{\text{amps}}} V_\alpha V_\beta^* A_\alpha(\vec{x}_i) A_\beta(\vec{x}_i)^* \right) - \sum_{\alpha,\beta}^{N_{\text{amps}}} V_\alpha V_\beta^* \int \eta(\vec{x}) A_\alpha(\vec{x}) A_\beta(\vec{x})^* d\vec{x}$$

$$\int \eta(\vec{x}) A_\alpha(\vec{x}) A_\beta(\vec{x})^* d\vec{x} \to \frac{1}{N_{\text{MC}}^{\text{gen}}} \sum_{i=1}^{N_{\text{MC}}^{\text{acc}}} A_\alpha(\vec{x}_i) A_\beta(\vec{x}_i)$$

- Initially each node needs a sub-collection of data or MC and an algorithm for computing the *A*
- With each fit iteration the node just needs to know the new values of the fit parameters and it returns its contribution to the log likelihood
- Many successful implementations exist... *the problem is scaleability*

GPU Memory Handling

(current design does a lot of caching on GPU)



*Assumptions:
1M signal events, 10M MC events, 5 particles, 20 amplitudes, 3 terms in each

Some First Results

- With most fits we are doing now, using the GPU is like cutting butter with a chainsaw
  - current CLEO analysis uses theoretically complex amplitudes and will soon benefit from being ported to GPU
  - plan to generate high statistics GlueX MC
- Total fit time is not a meaningful benchmark now, compare what we know to be limiting parts of calculation

100,000 $\gamma p \rightarrow \eta \pi^0 p$ events

Calculation	CPU Intel Core i7	GPU nVidia GTX 285
Breit Wigner	90 ms	0.87 ms
Angular Distributions	82 ms	3.9 ms
$\Sigma \log(I_i)$ (GPU includes CPU→GPU memcpy)	371 ms	33.4 ms

(double precision computation on both CPU and GPU)

Very easy factor 10-100 for a single graphics card!

No serious GPU optimizations yet -- use global memory on GPU which is "slow"

Outlook

- Scalability:
  - New Femi architecture from nVidia (not released yet) plans 448 cores and 6 GB of memory per card
  - Four cards per CPU possible -- cooling an issue (1 kW from GPUs alone)
  - IU fitter already has parallel implementation (via MPI) for cluster running -- will work immediately on multiple GPUs in the same box (1 process per GPU)
  - *May soon reach 1000 - 2000 cores and ~20 GB of GPU RAM in a single box*
  - We plan build a small GPU cluster at IU this spring -- size depends on price, but expect a few machines with a few GPUs... maybe in the neighborhood of 5000 - 10000 cores
- Issues:
  - Need double precision (not a problem with newest hardware), but also need to control precision in calculations to avoid problems with MINUIT
  - Debugging is challenging: the classic "cout" debugging method doesn't work with GPU (but emulation is available)
  - Eventually we will need to tune/optimize GPU algorithm, but to do so, we need a big enough problem first!

No foreseeable show stoppers -- a realistic route to 3-4+ orders of magnitude speed needed to maximize physics output from GlueX!