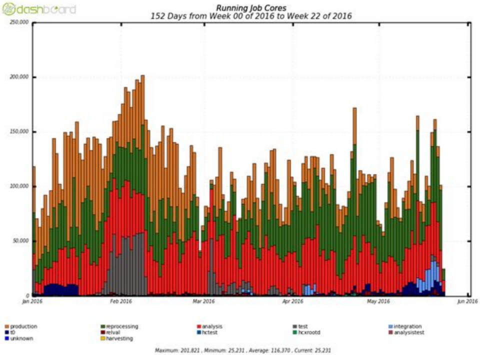
Opportunistic resource usage at Argonne Facility for CMS

J.R. Vlimant, J. Balcas, **Josh Bendavid** (Caltech)
On behalf of the CMS collaboration

Oct. 11, 2016
CHEP
San Francisco, United States

Introduction

- CMS computing resources mainly consist of WLCG grid resources
- Typically 150k cores in simultaneous use



- Largest scale computing facilities require more power-efficient architectures compared to traditional x86 server processors (IBM Power 7, Intel Xeon Phi, GPU's, etc)

ALCF

- Targeted ALCF production systems:
  - **Mira:** IBM Blue Gene/Q (Power 7), 49152 nodes, 786432 cores, 786 TB memory
  - **Cooley:**: Intel Haswell + Nvidia Tesla K80 GPU's, 126 nodes, 1512 Cores, 126 GPU's, 48TB system memory, 3 TB GPU memory
- ALCF future systems of interest:
  - **Theta:** (2017) Intel Xeon Phi Knight's Landing, 3240 nodes, 207360 cores, 50.6 TB high speed MCDRAM, 607.5 TB conventional DDR4 system memory, 414.7 TB in locally attached SSD storage
  - **Aurora:** (2019) Intel Xeon Phi Knight's Corner, >50k nodes, > 3.2M cores

Goals for CMS workflows at ALCF

- Enable relevant CMS software to run on Mira and/or Cooley (and eventually Theta/Aurora)
- Allow CMS production infrastructure to submit jobs
- Efficient transfer of required input/output data
- Efficient running of Monte Carlo generators

Challenges

- Main architecture for CMSSW and associated packaging is x86-64. Porting CMSSW and all dependencies to ppc64 for Mira is a non-trivial effort.
- CMS production workflows require CVMFS for software and dependencies
- WMAgent infrastructure used to manage CMS production jobs is very resource intensive, not conducive to a light-weight installation locally at ALCF
- Restrictive network security environment at ALCF
- Constraints on the number and frequency of job submissions

Current Progress - Job Submission

- Since WMAgent used to manage production jobs is difficult to install locally at ALCF, using Crab3 for testing (submission tool normally used for CMS analysis jobs)
- HTCondor and Crab3 taskworker setup on Cooley with functional job submission (setup on Mira currently not functional because of missing CVMFS support)
- Some modifications to Crab3 needed to allow submission of MPI jobs
- Stripped down preliminary port of CMSSW to ppc64 with some dependencies removed

Current Progress - Data Transfers

- Tested authorization using myproxy from the remote nodes and myproxy.alcf.anl.gov as the server.
- Tested transfers using globusurlcopy from CERN and Caltech to and from Mira using the gsiftp protocol.
- Tested changing the block size and using a different number of TCP streams for the transfer until reaching a saturation of the remote node NICs.
- Tested using the single DTNs instead of the load balancer. More work is needed to understand how to ensure the load balancer stripes among more than one DTN. It does this in principle using the transfer file size, but not clear how to pass that information to it when transferring data into ALCF.
- Registered CMS Site (T3_US_ANL) which is required to have in CMS Infrastructure for job matching and data transfers.

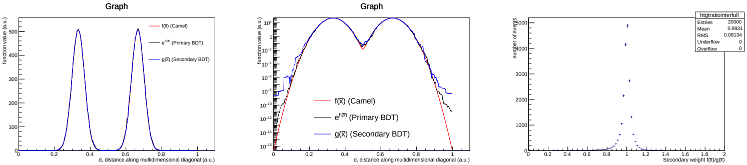
Monte Carlo Generators:

- CMS relies on detailed and large scale Monte Carlo production for modeling of the detector and underlying physics
- At LHC significant level of additional QCD radiation
- Many measurements or searches explicitly select or veto on the presence of extra jets
- Strong motivation for NLO and/or multi-leg/merged-multiplicity Monte Carlo generators in order to achieve highest possible accuracy for final states with additional jets
- Main multi-leg NLO generators of interest are Madgraph_aMC@NLO and Sherpa

Monte Carlo Generators:

- Madgraph_aMC@NLO does not presently support MPI parallelization → poor balancing of work between jobs, not straightforward or efficient to run on large MPI-oriented clusters
- Sherpa currently supports mainly MPI-level parallelization → not possible to run complex phase space integrations on conventional batch resources widely available to the collaboration
- Work-in-progress to exploit MC generators on large parallel computing facilities
- First test case is Sherpa on ALCF Mira system → already significant improvements to efficiency/parallelization in 2.2 series coming out of previous/ongoing work from ATLAS
- Interest together with authors to extend range of parallelization options in Madgraph_aMC@NLO as well
- Ongoing work to improve the efficiency of underlying phase-space integration/generation using machine learning techniques

Phase Space Integration with Boosted Decision Trees



(a) linear

(b) log

(c) weights

Algorithm	# of Func. Evals (4d Camel)	$\sigma_w / < w >$	σ_I / I (2e6 add. evts)
VEGAS	300,000	2.820	$\pm 2.0 \times 10^{-3}$
Foam	3,855,289	0.319	$\pm 2.3 \times 10^{-4}$
BDT-based	300,000	0.0819	$\pm 5.8 \times 10^{-5}$

- Multidimensional integration is a critical computational step in MC generators, Boosted Decision Trees can be used to extend existing FOAM-type algorithms
- Orders of magnitude improvements possible on non-factorisable integrands, tests for more realistic cases for MC Generators in progress

Outlook/Next Steps

- Enabling security model with ALCF team to balance local vs remotely installed CMS job-submission infrastructure
- Follow-up on interplay of MPI with CMSSW and CMS job-submission infrastructure and balance against constraints on number/frequency of job submission
- Follow-up on PPC porting of CMS software and assess cost-benefit vs expected prevalence of Power-based supercomputers in the future
- Monte Carlo Generators: Exploit the use of current Sherpa MPI-based parallelization on existing Mira system, follow/contribute to better parallelization in Madgraph_aMC@NLO, further developments on underlying phase space integration/sampling with machine learning
- Prepare for the deployment of Theta (x86-based, but still challenging to fully exploit)