Machine learning & Deep learning at ALCF

Elise Jennings

Argonne Leadership Computing Facility

Data Sciences Group

Argonne National Laboratory

ejennings@anl.gov

Argonne NATIONAL LABORATORY

OUTLINE

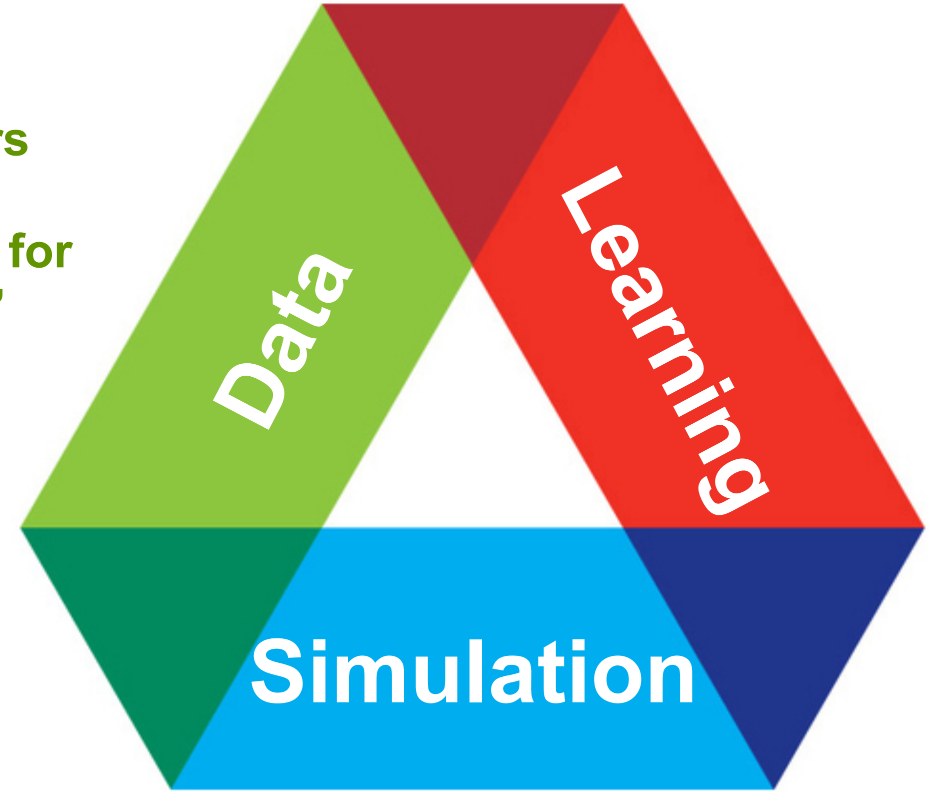
- **Data Science Program**
- **Highlights of current ADSP projects**
- **ML, DL & workflow software**
- **Other research projects**

ALCF DATA SCIENCE PROGRAM (ADSP)

- Targets “big data” science problems that require the scale and performance of leadership computing resources: Mira and Theta
- Techniques like uncertainty quantification, statistics, machine learning, deep learning, databases, pattern recognition, image processing, graph analytics, data mining, real-time data analysis, and complex and interactive workflows…more in the talk tomorrow!

https://www.alcf.anl.gov/alcf-data-science-program

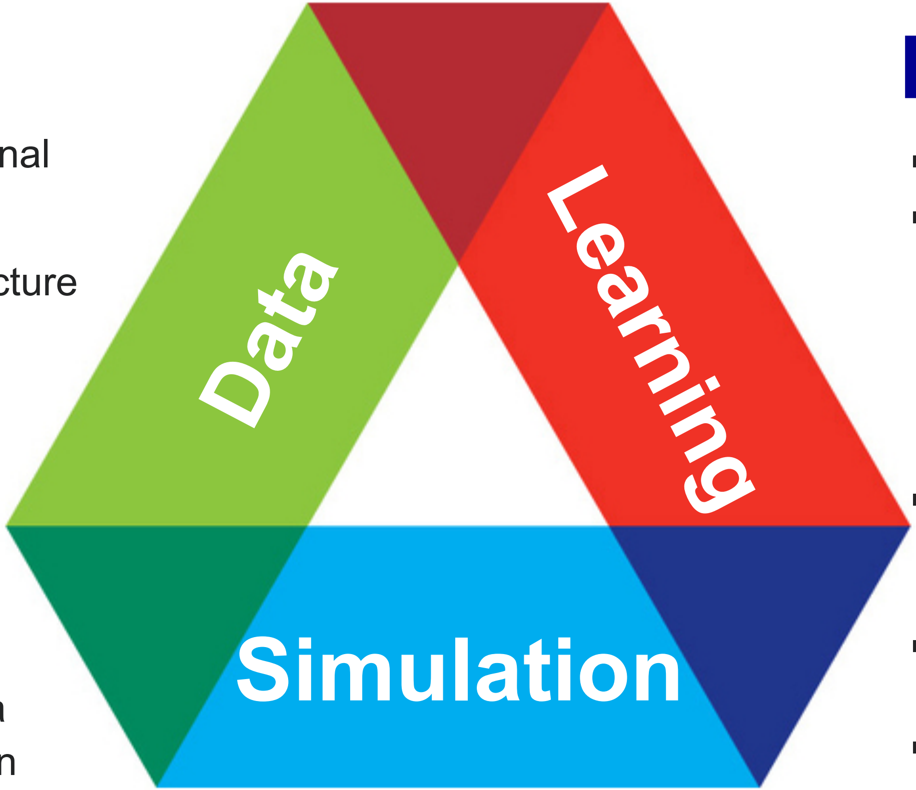
CORAL Supercomputers and Exascale Systems support for "Three Pillars"



Cross-cutting proposals targeting the convergence of simulation, data & learning

Data

- Experimental/observational data
  - Image analysis
  - Multidimensional structure discovery
- Complex and interactive workflows
- On-demand HPC
- Persistent data techniques
  - Object store
  - Databases
- Streaming/real-time data
- Uncertainty quantification
- Statistical methods
- Graph analytics



Learning

- Deep learning
- Machine learning steering simulations
  - Parameter scans
  - Materials design
  - Observational signatures
- Data-driven models and refinement for science using ML/DL
- Hyperparameter optimization
- Pattern recognition
- Bridging gaps in theory

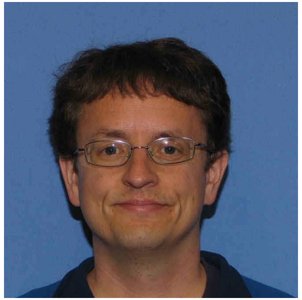
Venkat Vishwanath

Tom Uram

Murat Keceli

Elise Jennings

Alvaro Vazquez Mayagoitia



Adrian Pope

Misha Salim

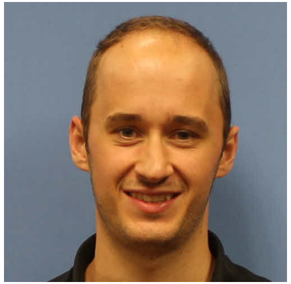
Preeti Malakar

Prasanna Balaprakash

Jing Li

William Scullin

Taylor Childers



Samuel Flender

Richard Zamora

Francois Tessier

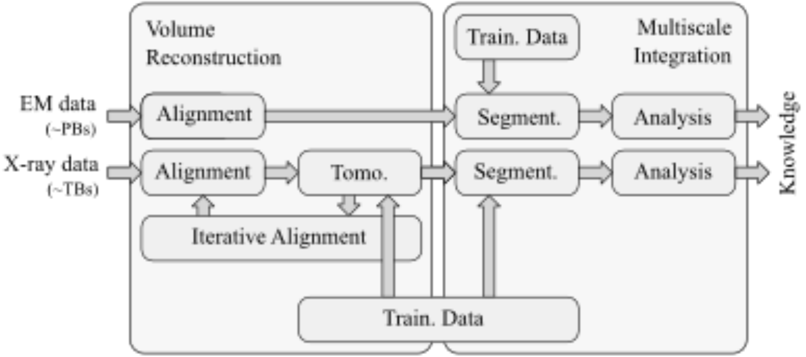
Xiao-Yong Jin

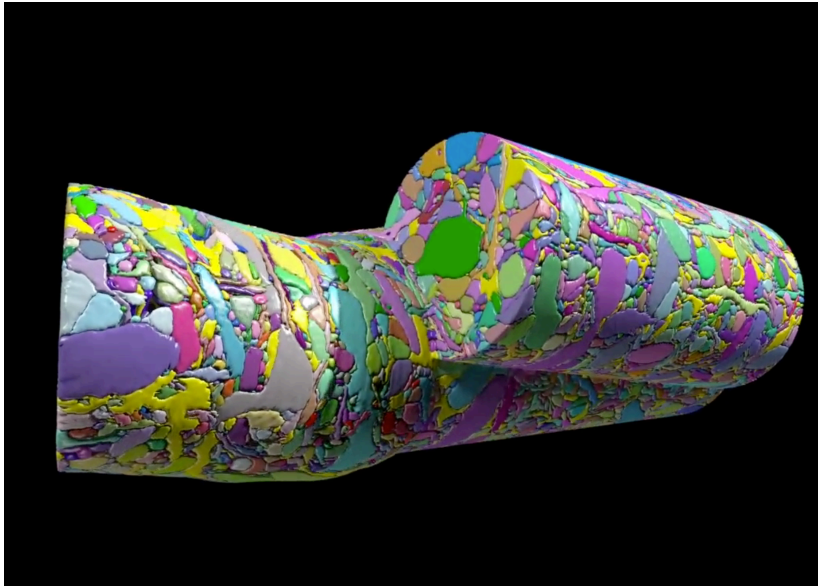
Ganesh Sivaraman

- **Highlights of current ADSP projects**

Large-scale computing and visualization on the connectomes of the brain

- **PI**: Doga Gursoy, Argonne National Laboratory, 25M core hrs
- **Objectives:** Comprehensive maps of connections within the brain - > Connectomics. Extreme scale data-centric computational pipelines for brain science.
- **Imaging:** X-Ray extended tomography with 1micron resolution done an the Advanced Photon Source. EM based imaging at nm resolution
- **Segmentation:** State of the art segmentation algorithms (U-Net, Google's FFN). Extract features: cell bodies, myelinated axons, blood vessels…
- **Impact:**Scalable workflows will help extract valuable knowledge about disease models such as Alzheimer's, autism spectrum disorder, etc., and enable advances in neuromorphic computing.



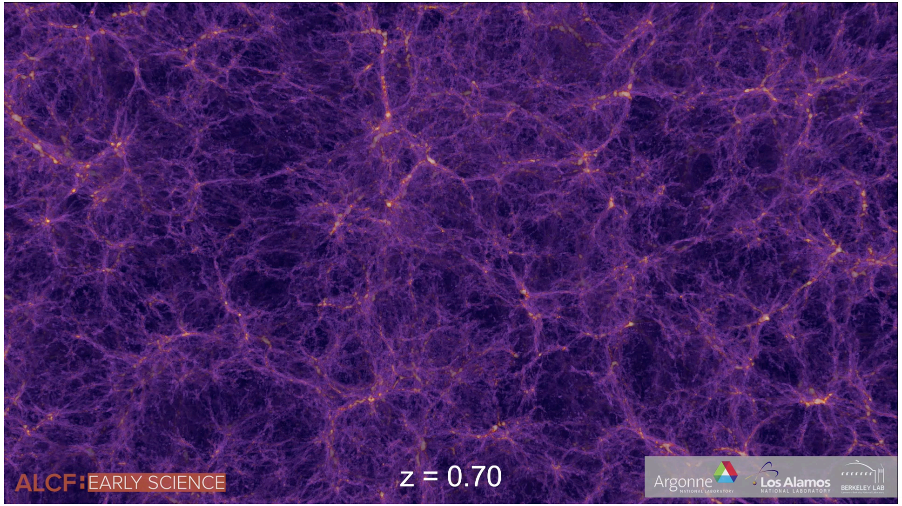


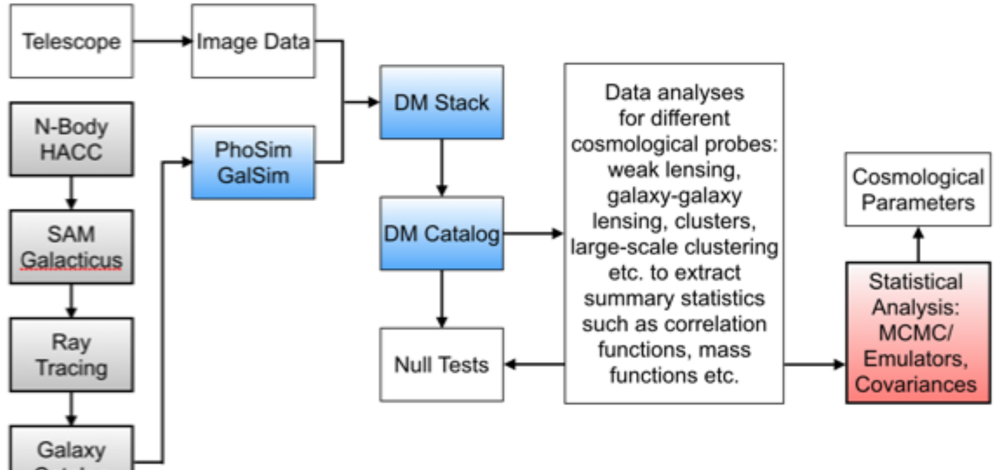
ADSP Category: Tier-1

Machine Learning and Science

Realistic Simulations of the LSST Survey at Scale

- **PI**: Katrin Heitmann, Argonne National Laboratory. 70M core hours
- **Objectives:** Development & execution of end-to-end workflow starting from simulation to the creation of sky maps with realistic galaxies.
- **Impact:** Deliver largest & most detailed synthetic sky maps ever created ready for the first data from LSST.
- **Approach:** Create a virtual survey with images almost indistinguishable from real LSST observations, develop an end-to-end pipeline for LSST data processing and analysis on ALCF supercomputers

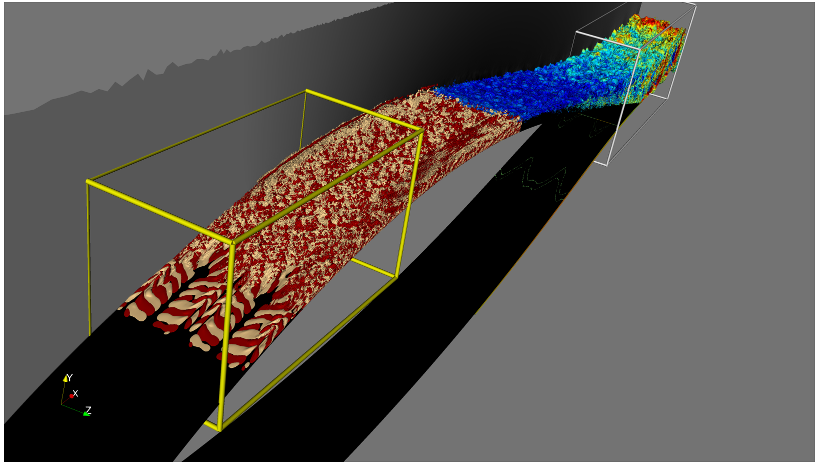




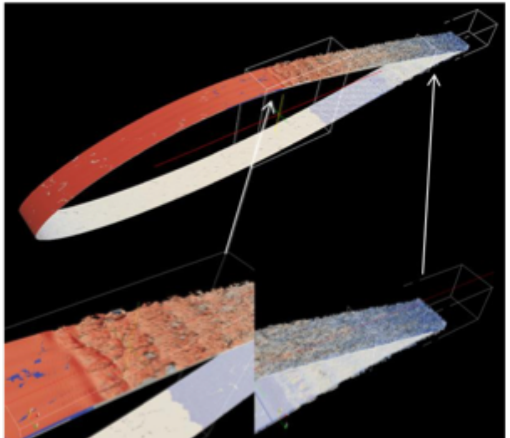
DM: Data Management; SAM: Semi-analytic Modeling

ADSP Category: Tier-1 Workflows and Science

Enabling Multi-Scale Physics for Industrial Design using Deep Learning Networks



- **PI**: Dr. Rathakrishnan Bhaskaran, GE Global Research. 8M core hours
- **Objectives:** Develop data driven turbulence models for improved predictive accuracy using machine learning and large data sets
- **Impact:** Significant stepping stone towards extensions to complex aerodynamic flows in turbomachinery, as well as to the larger computational fluid dynamics community in general. The outcomes will impact industries including aerospace and power generation
- **Approach:** Data from wall-resolved LES simulations - > DL models -> drive improvements to turbulence models within the RANS framework

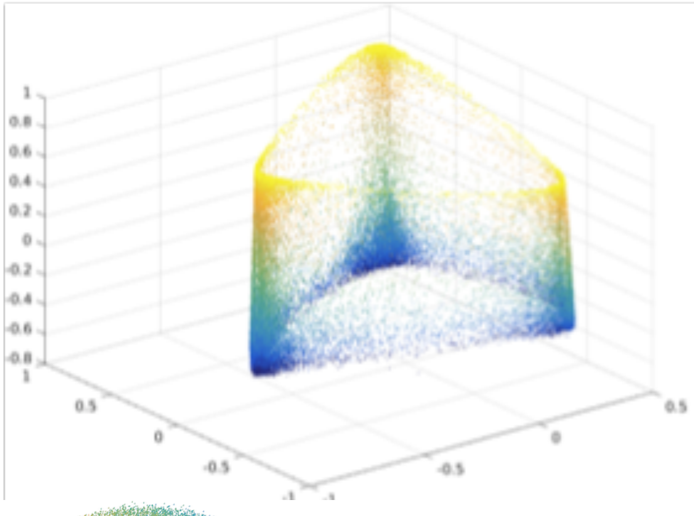


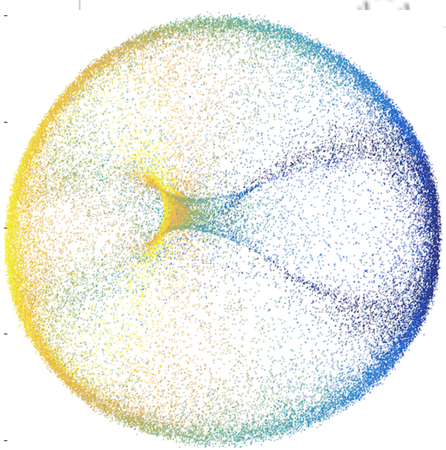
ADSP Category: Tier-1

Machine learning in Engineering

Constructing and Navigating Polymorphic Landscapes of Molecular Crystals

- **PI:** Alexandre Tkachenko, Univ. of Luxembourg, 60M core hours
- **Objectives:** Combine atomistic quantum simulations and data science methods -> accurate predictions of novel molecular crystals. Data-driven knowledge for structures, relative energies, and properties for many polymorphic systems.
- **Impact:** Alternative energy materials, novel molecular electronics, and disease-fighting pharmaceutical agents
- **Approach:** Explore polymorphic energy landscape for ~O(100) molecular crystals of interest. Manifold learning methods for crystal structure prediction (CSP) on supercomputing systems for organic molecular crystals





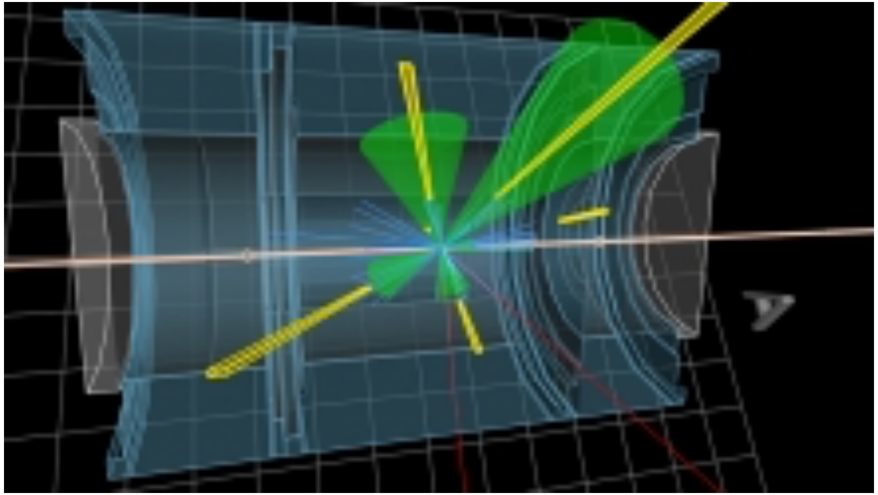
Low-dimensional representation of the collective coordinates of ab initio molecular dynamics trajectories of ethanol (above) and malonaldehyde (left)

ADSP Category: Tier-2

Machine Learning and Science

Massive hyperparameter searches on deep neural networks using leadership systems

- **PI:** Prof. Pierre Baldi, Univ. of California - Irvine, 10M core hours
- **Objectives:** Design and development of massive hyperparameter searches on deep neural networks in order to investigate both the fundamentals of deep learning algorithms. Detection of exotic particles at the LHC.
- **Impact:** Improve the use of leadership resources for deep learning in science. It will also increase use of LCF resources for LHC scientists in the discovery of new particle physics. Improve detection of Higgs boson particles.
- **Approach:** Scale the Sherpa hyper-parameter optimization to run on leadership systems and develop novel search algorithms



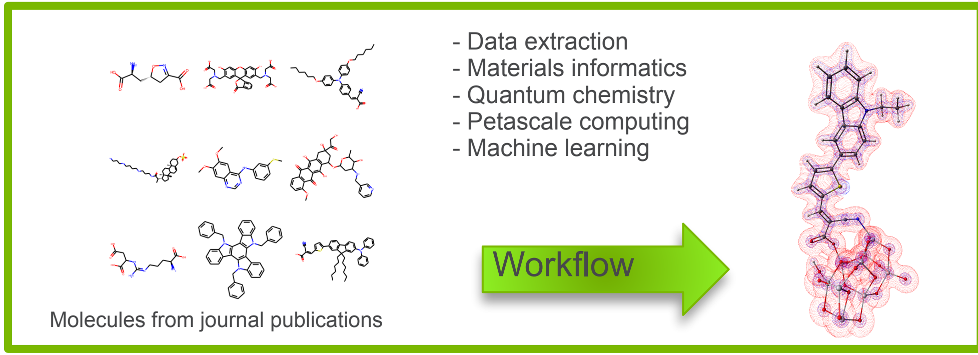
ADSP Category: Tier-2

Machine Learning and Science

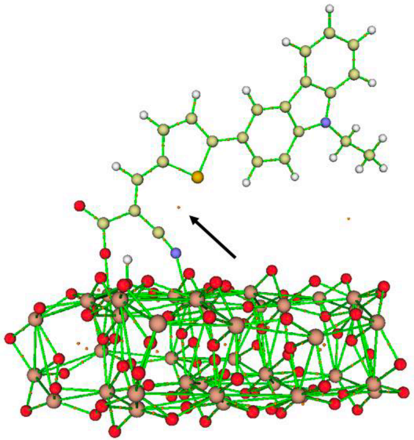
Data-Driven Molecular Engineering of Solar-Powered Windows

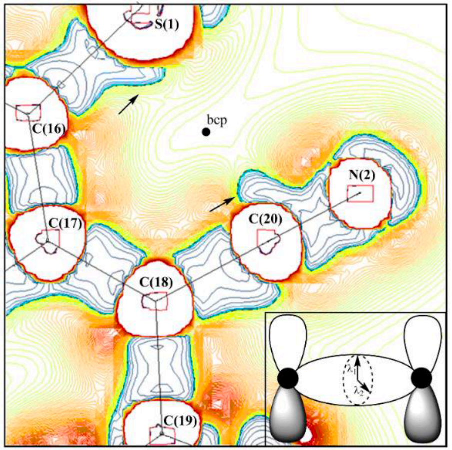
- **PI:** Jacqueline Cole,Cambridge University, Argonne National Laboratory. 117M core hours
- **Objective:** Discovery of better performing light-absorbing dye molecules
- **Impact:** Dye-sensitized solar cells (DSCs) are an alternative to organic metal cells with a desirable cost-efficiency tradeoff in the design of energy-efficient buildings. 40% of total energy usage in the USA comes from buildings.
- **Approach:**
  - Extracting data from 300,000 publications and computing properties of 80,000 molecules require leadership computing resources.Uses quantum chemistry using TDDFT.
  - Explored viable molecular descriptors to evaluate candidate molecules using ML

ADSP Category: Tier-1 Workflows and ML



J. M. Cole *et al*, *ACS Appl. Mater. Interfaces*, **2017**, *9* (31),25952





- **Machine Learning, Deep Learning & Workflow software**

Software

- **ML/DL:**
  - TensorFlow, Keras, Neon, MXNet, Caffe2, Theano, CNTK, PyTorch, Sci-kit Learn, Graph Analytics (Cray Graph Engine), Horovod…
  - With performance libraries e.g. Intel MKL, MKL-DNN, LibXSMM etc enabled
  - Intel optimized Tensorflow
    - Conda package on Theta
    - Intel Distribution for Python's optimized numpy

```
# Python 2.7
pip install https://anaconda.org/intel/tensorflow/1.4.0/download/tensorflow-1.4.0-cp27-cp27mu-linux_x86_64.whl

# Python 3.5
pip install https://anaconda.org/intel/tensorflow/1.3.0/download/tensorflow-1.3.0-cp35-cp35m-linux_x86_64.whl

# Python 3.6
pip install https://anaconda.org/intel/tensorflow/1.4.0/download/tensorflow-1.4.0-cp36-cp36m-linux_x86_64.whl
```

Keras

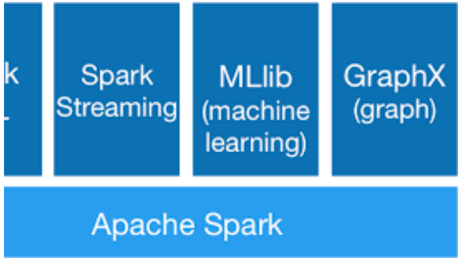
mxnet

TensorFlow

HOROVOD

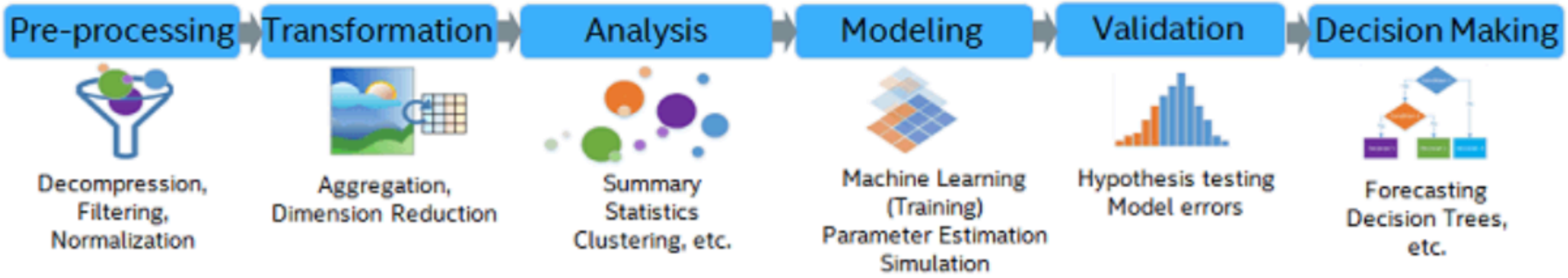
Software

- **Workflow/Data analysis:**
  - Containers
    - Singularity container solution for application science workloads
    - Environment imported into container
    - mount additional directories into the container with the -B flag
    - aprun -n $RANKS -N 1 singularity exec my_image.img ./my_binary
    - **- >Taylor Childers' talk**
  - Jupyter Hub, MongoDB, Apache Spark, R
  - Balsam
  - Python
    - Intel and Cray modules on Theta
    - ALCF alcfpython/2.7.14-20180131
    - **-> William Scullin's talk**
- **Visualization:** Paraview on Theta



Argonne NATIONAL LABORATORY

Intel® Data Analytics Acceleration Library (Intel® DAAL)



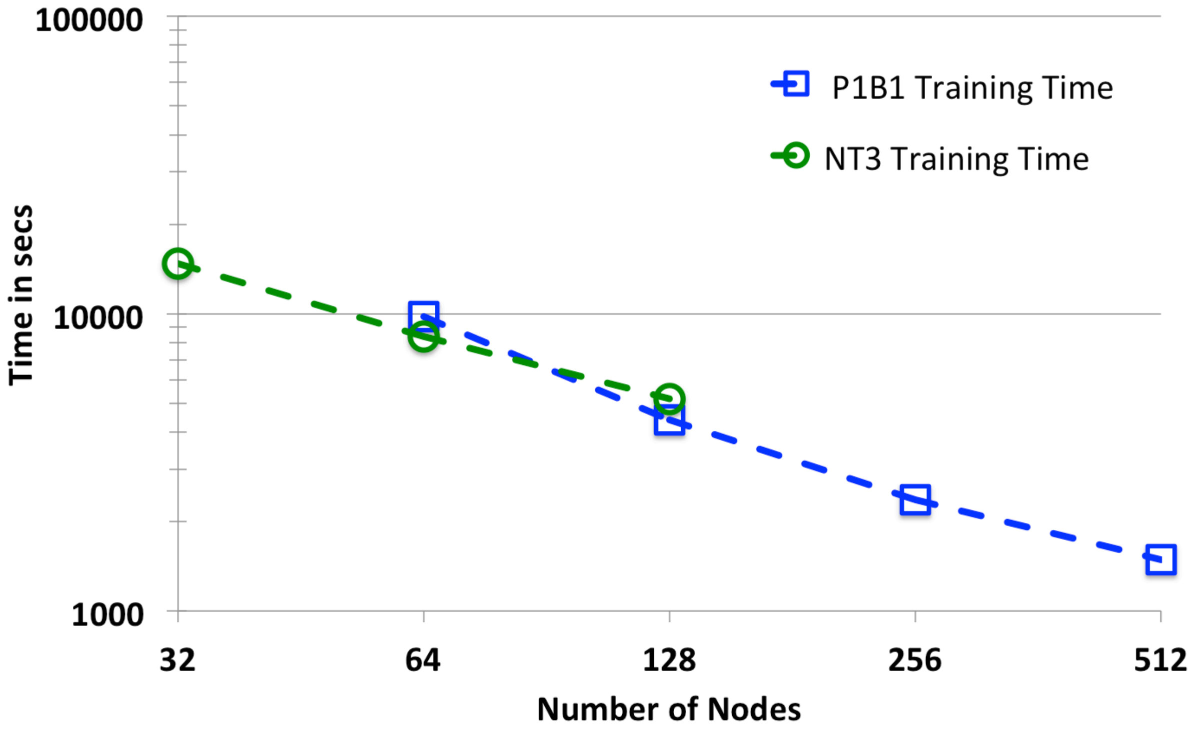
Modules

Here is a list of all modules:

▼Algorithms	
▼Analysis	
▼Association Rules	Contains classes for the association ru
Batch	
▼Cholesky Decomposition	Contains classes for computing Choles
Batch	
▼Correlation Distance Matrix	Contains classes for computing the coı
Batch	
▼Correlation and Variance-Covariance Matrices	Contains classes for computing the coı
Batch	
Distributed	
Online	
▼Cosine Distance Matrix	Contains classes for computing the co:
Batch	
▼Expectation-Maximization	Contains classes for the EM for GMM a
▶Computation	Contains classes for the EM for GMM a
▶Initialization	Contains classes for the EM for GMM i
▼K-means Clustering	Contains classes of the K-Means algor

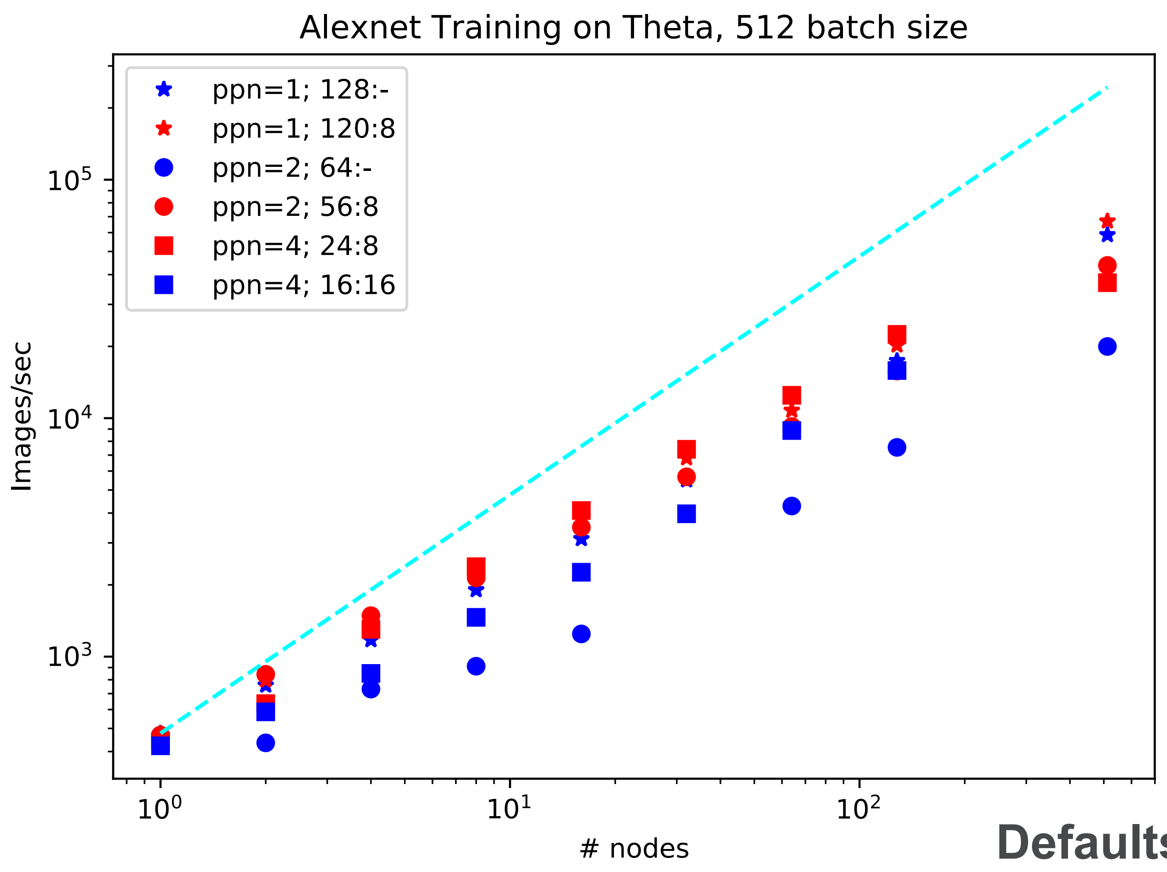
Argonne NATIONAL LABORATORY

Data-Parallel Deep Learning on Theta



- Tensorflow and Horovod for data-parallel training
- Scaled data-parallel training for two Candle benchmarks to 512 nodes on Theta

TensorFlow Optimizations on Theta



- Data format: NCHW
- Inter-op / intra-op: impact parallelism within one layer as well as across layers.
- OMP_NUM_THREADS
- KMP_BLOCKTIME
- ALPS affinity settings
  - - cc none versus - - cc depth -d -j

Defaults not optimal

```
int32 inter_op_parallelism_threads = 5;
int32 intra_op_parallelism_threads = 2;
```

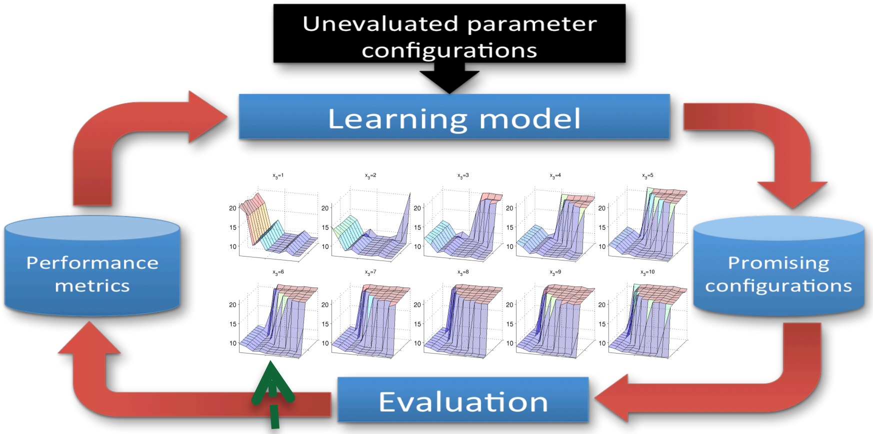
- **Other research projects**

Scalable Hyperparameter Search for Deep Learning

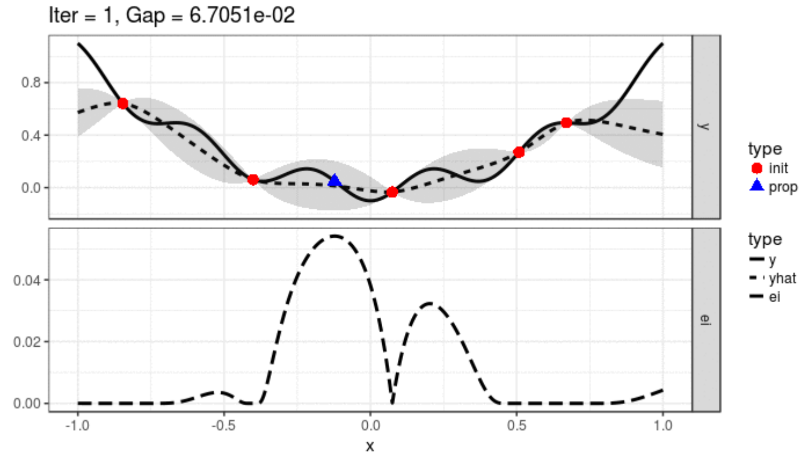
- Hyperparameter optimization is of paramount importance for deep learning for science. This is expected to be a **key workload on exascale systems**
- Model-based search iteratively refines the model in promising input region by obtaining new outputs at unevaluated input configurations

- General framework:
  - Initialization phase using Random or Latin hypercube sampling
  - Iterative phase wherein we fit a model and sample using this model

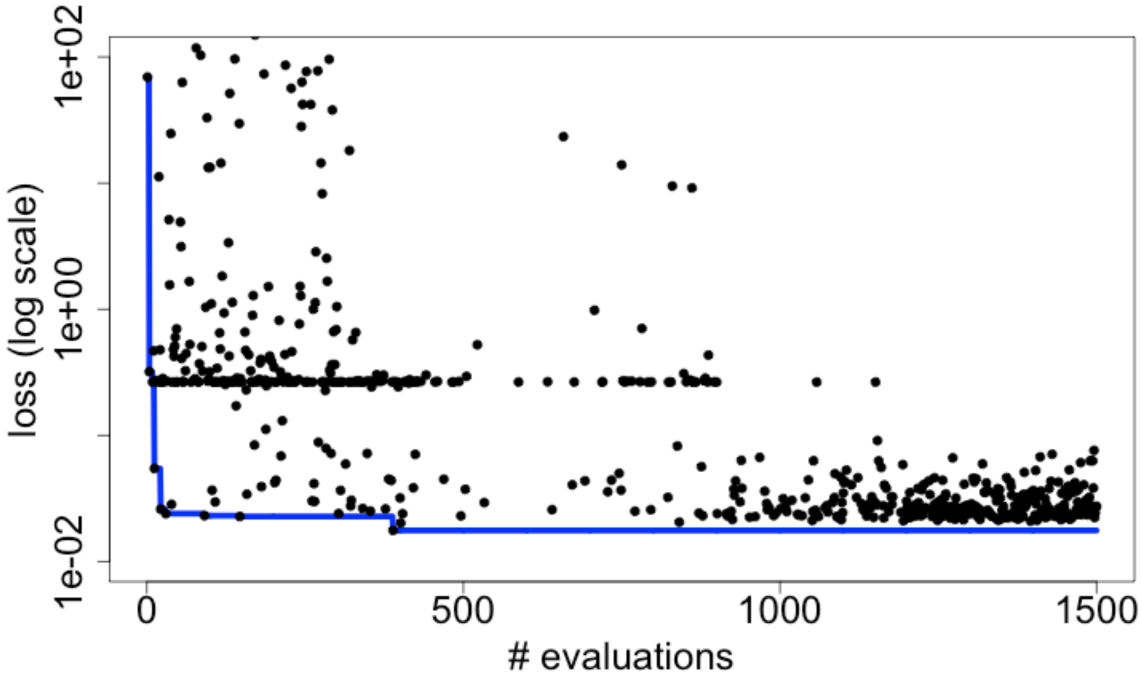
- Random forest and xgboost algorithms are used for building models because of their ability to handle the integer and categorical parameters



Example Surrogate Model Fitted to Sampled Performance (iterative refinement improves the learning model)



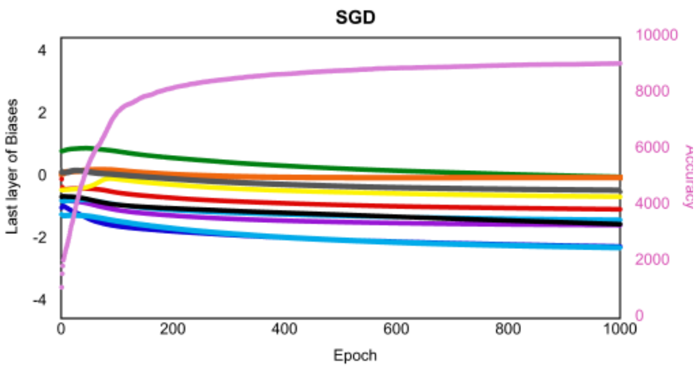
Scalable Hyperparameter Search for Deep Learning

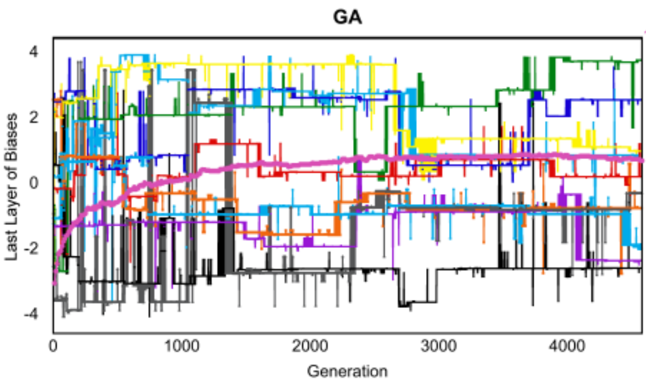


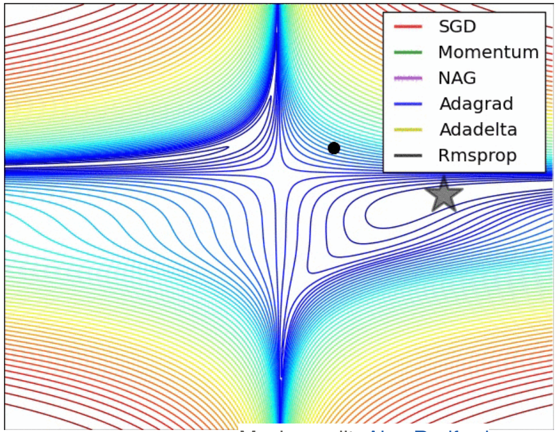
Scaled the hyperparameter search framework for CANDLE benchmarks on **300 Theta nodes.** Each node evaluates a deep neural network for ~1 hour. The figure depicts 5 iterations with 1500 evaluations (~5 hours)

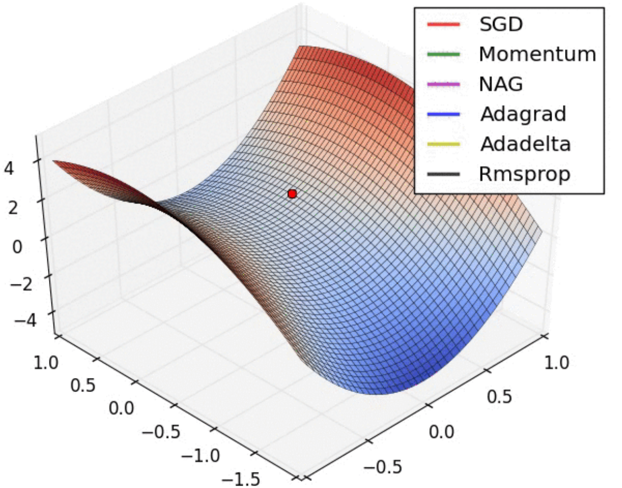
Optimizers in Deep Learning

- **Genetic Algorithms:**
  - Alternative to SGD
    - Mutation(crossover(P_i, P_j))
  - Hyperparameter estimation
    - Promising for optimal model structure









Movie credit: Alec Radford

Balsam

- Maintain a job database for your computational campaign
- **Unlimited job submission** and DAG workflows
- **Auto-package and execute** jobs on ALCF resources
- Python+Django API enables **dynamic workflows** with runtime job creation/killing/monitoring
- Resilient workflows that **gracefully handle job failures/ timeouts;** optional user-defined timeout/error handlers

Command line tools for workflow management

Manipulate job database, start job launcher, …

```
(testmpi) misha@alcfwl115 in ~/workflow/argobalsam/balsam on develop*$  balsam ls
                              job_id | name |     workflow |              application |       state
--------------------------------------------------------------------------------------------------
dd9119b3-3e79-4673-81f4-fa1603006191 | sim1 | my_campaign | test_balsam_1__runjobs | CREATED
56f2d7f1-f05a-4ab4-ab10-f24c5fc115c9 | sim2 | my_campaign | test_balsam_1__runjobs | RUN_DONE
991fb79e-9302-45f2-9456-d7f78af200b2 | sim3 | my_campaign | test_balsam_1__runjobs | RUN_TIMEOUT
```

Python+Django API for dynamic job control

Monitor job output, create & kill jobs at runtime, run queries on job database

```python
# Read in new results
new_jobs = BalsamJob.objects.filter(job_id__in=my_jobs)
new_jobs = new_jobs.filter(state="JOB_FINISHED")
new_jobs = new_jobs.exclude(job_id__in=finished_jobs)
for job in new_jobs:
    result = json.loads(job.read_file_in_workdir('result.dat'))
```

Balsam in action

```
import balsam.launcher.dag as dag
from balsam.service.models import BalsamJob, END_STATES

def create_job(x, eval_counter, cfg):
    '''Add a new evaluatePoint job to the Balsam DB'''



child = dag.spawn_child(name=jname, workflow="dl-hps",
            application="eval_point", wall_time_minutes=2,
            num_nodes=1, ranks_per_node=1,
            input_files=f"{jname}.dat",
            application_args=f"{jname}.dat",
            wait_for_parents=False
        )
```

-> Tom Uram's talk

Thank you !

Upcoming Program Deadlines

Aurora Early Science Program for
Learning and Data
Call for Proposals in January 2018

ADSP Program
Call for Proposals in April 2018