Breakout Session 1: Track A

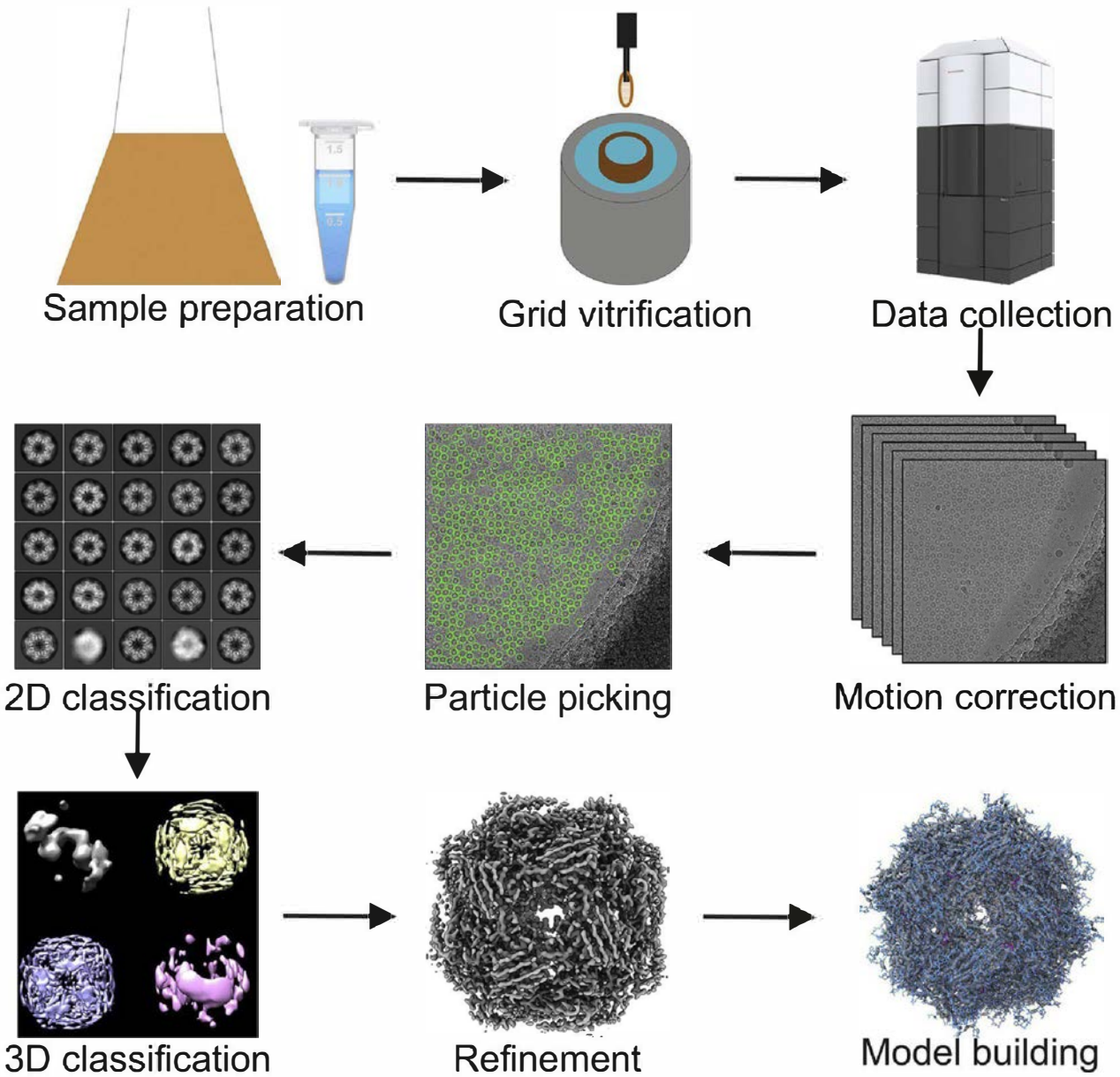
Implementation of AWS Cloud Computing for cryoEM Data Processing

Dr. Joseph Marcotrigiano (Moderator)
Senior Investigator, NIH/NIAID

Implementation of AWS Cloud Computing for cryoEM Data Processing

Joseph Marcotrigiano

The pipeline of cryo-EM structure determination



- Homogenous, highly pure protein sample is applied to cryo-EM grids.
- The sample is rapidly frozen in liquid ethane in a thin layer of vitreous ice.
- Images are recorded as movies on a transmission electron microscope.
- Movie frames are aligned to reduce effects of drift.
- Particles are picked from each micrograph with those representing the same view grouped together to increase the signal-to-noise (2D class).
- 2D classes are then computationally aligned to generate a 3D map.
- 3D classification can identify different conformational states of the protein.

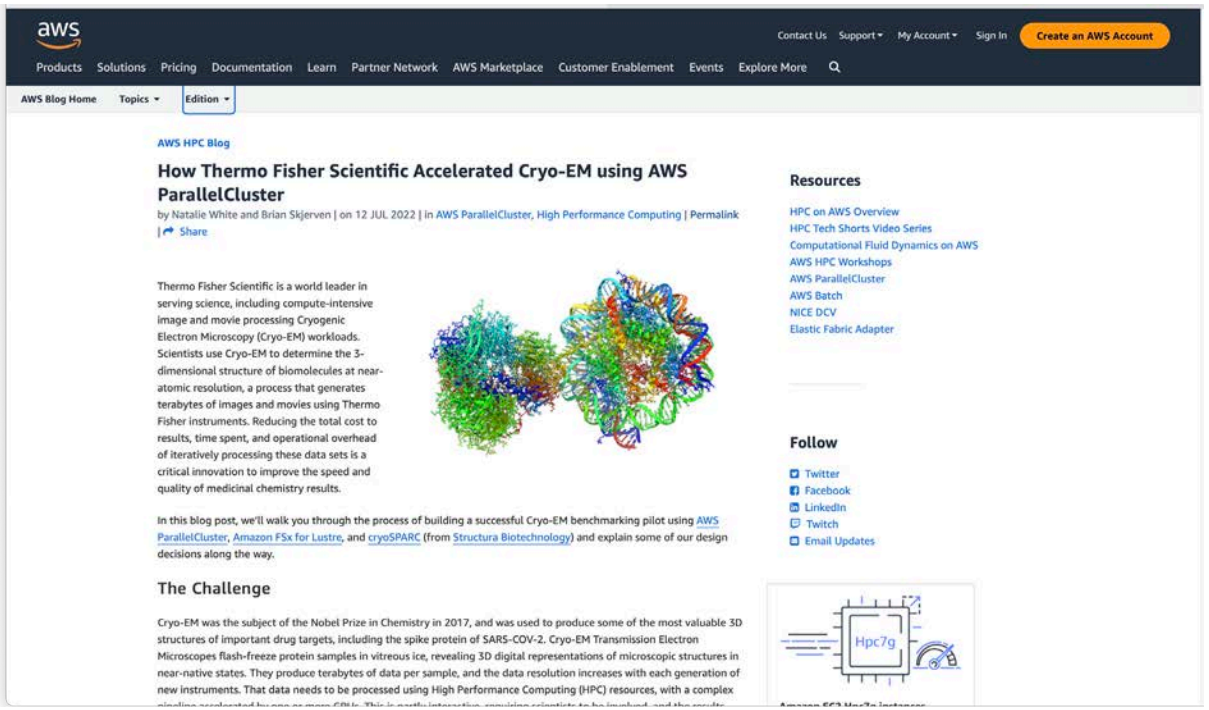
Data Amounts

- Each cryo-EM data set consists of about 5,000-10,000 movies
- A typical movie is about 0.5Gb in size, resulting in 2.5-5Tb per dataset.
- Data processing of the images increases the size of the data by 3-5x.
- In addition to data storage, cryo-EM data processing is computationally intensive.
- The structure of a yeast spliceosomal complex required more than half a million CPU hours of classification and high-resolution refinement (Kimanius et al. eLife, 2016).
- The implementation of graphics processors (GPUs) to alleviate the computational bottleneck has transformed the cryo-EM field.
- Many of the common cryo-EM software packages have been redesigned to take advantage of recent advances in GPU technology and can implement many independent tasks simultaneously.

Local High Performance Computing

- Biowulf – Well set up for cryo-EM, however wait times for GPUs can be long (V100 wait times 1-3hours)
- Locus – Has GPUs but I/O is not configured properly. Motion correction of one dataset took 8 days. Upgrades ongoing (Skyline)
- BigSky – Well set up for cryo-EM but it is for RML only
- Workstation – Have one workstation with 4 GPUs and 50Tb of storage. Data has to go through Locus and storage is limiting

Cryo-EM on AWS



aws

Contact Us Support My Account Sign In Create an AWS Account

Products Solutions Pricing Documentation Learn Partner Network AWS Marketplace Customer Enablement Events Explore More

AWS Blog Home Topics Edition

AWS HPC Blog

How Thermo Fisher Scientific Accelerated Cryo-EM using AWS ParallelCluster

by Natalie White and Brian Skjerven | on 12 JUL 2022 | in AWS ParallelCluster, High Performance Computing | Permalink | Share

Thermo Fisher Scientific is a world leader in serving science, including compute-intensive image and movie processing Cryogenic Electron Microscopy (Cryo-EM) workloads. Scientists use Cryo-EM to determine the 3-dimensional structure of biomolecules at near-atomic resolution, a process that generates terabytes of images and movies using Thermo Fisher instruments. Reducing the total cost to results, time spent, and operational overhead of iteratively processing these data sets is a critical innovation to improve the speed and quality of medicinal chemistry results.

In this blog post, we'll walk you through the process of building a successful Cryo-EM benchmarking pilot using AWS ParallelCluster, Amazon FSx for Lustre, and cryoSPARC (from Structura Biotechnology) and explain some of our design decisions along the way.

The Challenge

Cryo-EM was the subject of the Nobel Prize in Chemistry in 2017, and was used to produce some of the most valuable 3D structures of important drug targets, including the spike protein of SARS-COV-2. Cryo-EM Transmission Electron Microscopes flash-freeze protein samples in vitreous ice, revealing 3D digital representations of microscopic structures in near-native states. They produce terabytes of data per sample, and the data resolution increases with each generation of new instruments. That data needs to be processed using High Performance Computing (HPC) resources, with a complex

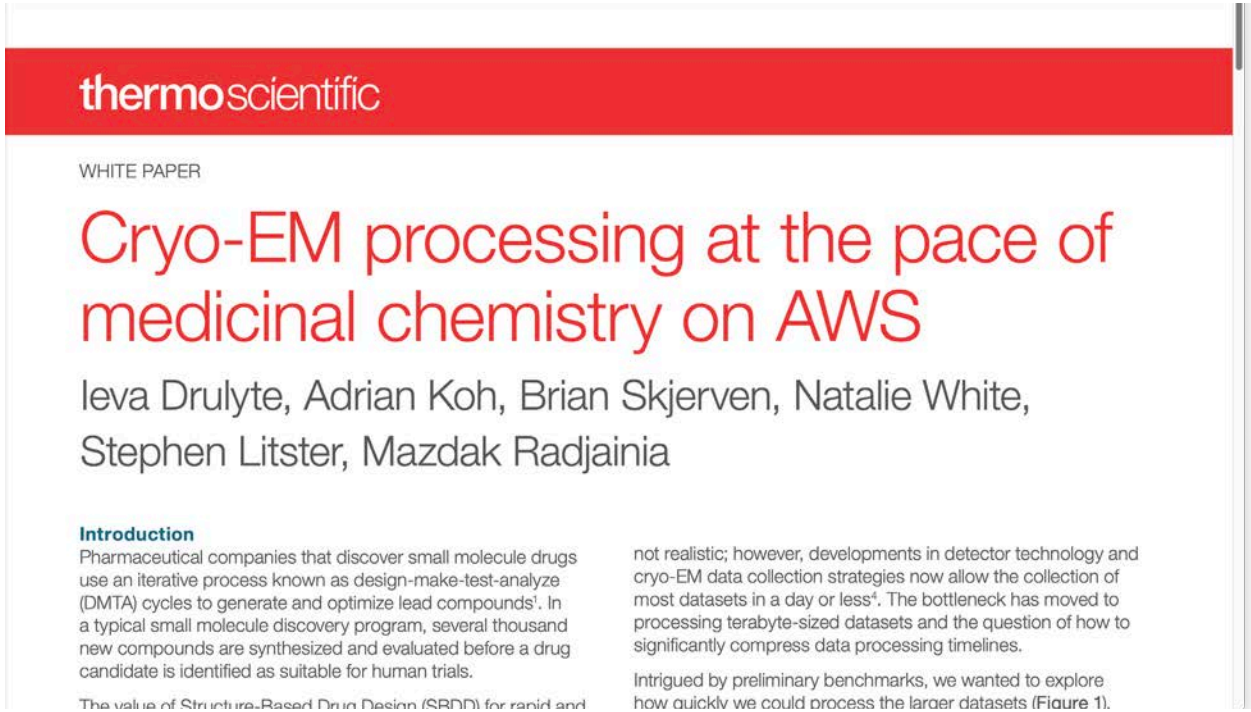
Resources

HPC on AWS Overview
HPC Tech Shorts Video Series
Computational Fluid Dynamics on AWS
AWS HPC Workshops
AWS ParallelCluster
AWS Batch
NICE DCV
Elastic Fabric Adapter

Follow

Twitter
Facebook
LinkedIn
Twitch
Email Updates

Hpc7g



thermoscientific

WHITE PAPER

Cryo-EM processing at the pace of medicinal chemistry on AWS

Ieva Drulyte, Adrian Koh, Brian Skjerven, Natalie White, Stephen Litster, Mazdak Radjainia

Introduction

Pharmaceutical companies that discover small molecule drugs use an iterative process known as design-make-test-analyze (DMTA) cycles to generate and optimize lead compounds[1]. In a typical small molecule discovery program, several thousand new compounds are synthesized and evaluated before a drug candidate is identified as suitable for human trials.

The value of Structure-Based Drug Design (SBDD) for rapid and

not realistic; however, developments in detector technology and cryo-EM data collection strategies now allow the collection of most datasets in a day or less[4]. The bottleneck has moved to processing terabyte-sized datasets and the question of how to significantly compress data processing timelines.

Intrigued by preliminary benchmarks, we wanted to explore how quickly we could process the larger datasets (Figure 1).

https://assets.thermofisher.com/TFS-Assets/MSD/Reference-Materials/pharma-cryosparc-wp0028.pdf
https://aws.amazon.com/blogs/hpc/how-thermo-fisher-scientific-accelerated-cryo-em-using-aws-parallelcluster/

Timeline

- September 2022 – an NIH Cloud Lab account was created
  - Goal – to explore cloud computing for cryo-EM data processing
- $500 credit to work with Amazon Web Services (AWS) to load and test one popular cryo-EM software (cryoSPARC)
- After several attempts, cryoSPARC was loaded onto the NIH Cloud Lab account
  - Evan Bollig, Tom Fonseca, and Gargi Singh – AWS
- Successful structure determination of apoferritin test sample
- $25,000 credit on AWS from NIH STRIDES (Nick Weber)
  - Goal – to process datasets from experimental samples
- Early March 2023, cryoSPARC loaded and experimental data uploaded
- Spring 2023, received $100,000 grant from NIHCIT for further cloud development
- Summer 2023 joined SBGrid and created a complete structural biology platform in the cloud

SBGrid in AWS

SBGrid CONSORTIUM

Supported Software | Computing Resources | Get Help | About SBGrid

Home | Search | Latest Changes | Page History

AWS

SBGrid in AWS EC2

SBGrid is easy to use in on cloud resources. This is a short guide to using SBGrid on AWS.

- SBGrid in AWS EC2
  - Compatible AMIs
  - Prerequisites
  - Download the SBGrid command line interface
  - Activate your installation
  - Install programs
  - Run your software

Compatible AMIs

```
     __|  __|_  )
     _|  (     /   Amazon Linux AMI
    ___|\___|___|
```

We recommend CentOS / RHEL 7 AMIs for the SBGrid software stack. Amazon AMIs appear to work well, but are not routinely tested. Debian / Ubuntu AMIs may work but are untested. If you have problems let us know.

For the example below we are using an Amazon Linux AMI, 2018.03.0 (HVM), SSD Volume Type - ami-14c5486b

Prerequisites

First login to your running instance and install **tcsh**. It is not provided on linux by default but is required by several SBGrid titles.

```
sudo yum install tcsh
```

SBGrid Wiki

Installing SBGrid Software

- Overview
- Supported Platforms
- Required packages
- Site installations
- Graphical installation
- Command line installation
- macOS and Apple M1 Silicon

Using the SBGrid Environment

- Getting Started with SBGrid
- SBGrid Environment
- Managing Software Versions
- Modulefiles for SBGrid

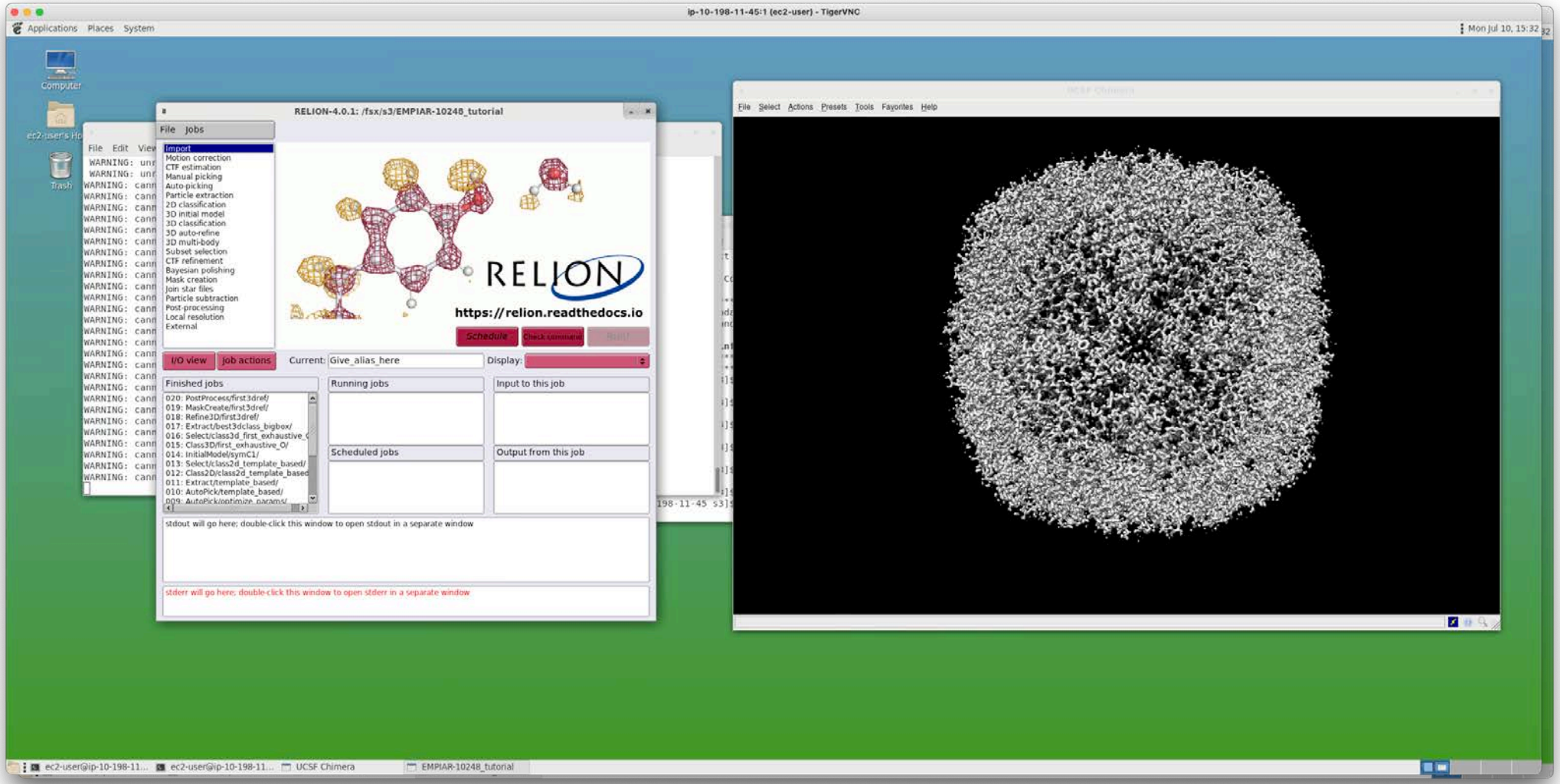
Support for Site Administrators

- Managing your Installation
- Administrator Software Version Overrides

Hardware Support Notes

We are working with Jason Key at SBGrid to implement and test packages on AWS

Virtual Desktop on the AWS



Virtual Desktop on AWS using TigerVNC

Alphafold, cisTEM, Chimera, cryoSPARC, Model Angelo, DeepEMhancer,Relion (4 and 5), Topaz, loaded onto AWS using SBGrid

Thanks to Tee Gobezie

Cloud vs. Biowulf vs. Workstations

"The data processing worked well and I got 3 maps at 3.1~3.5 A resolution after NUR refinement last week. Compared to Biowulf, it did take much less time in the queue before starting the job, and also finishes the job faster." – Jingyu Zhan (postdoc with Di Xia)

Job	Biowulf	Cloud	Micrographs	Classes	Particles	Box size	Customized settings
Motion correction	23h11m	20h30m	9,685				
Patch CTF	2h24min	2h02m	9,685				
2D classification	15h	5h		150	1.8M	360	Uncertainty factor 4, 2 final iterations, 50 online-EM iterations, batchsize 400, 4 gpus
Hetero Refinement	30h	6h11min		6	600,000	360	3 final iterations,1 gpu
NUR refinement	15h	3h07m			~132,000	360	2 extra final passes

"One week on AWS equals 4-6 weeks on a workstation" Sarah Nyenhuis (postdoc with Jenny Hinshaw)

Job	Workstation	Cloud	Micrographs	Classes	Particles	Box size	Customized settings
Motion correction	18h33m	2h39m	2,435				Fcrop 1/2
Patch CTF	3h21min	30m	2,435				
2D classification	3h42min	18min		50	149,452	744 (fcrop 500)	Uncertainty factor 10, align filament classes vertically, 17 online-EM iterations, 2 gpu
2D classification	memory error	19min		50	165,469	870 (fcrop 580)	Uncertainty factor 10, align filament classes vertically, 17 online-EM iterations, 2 gpu
Helical Refinement	1h51min	1h28min			47,349	744 (fcrop 500)	symmetry imposed, symmetry search, 15 iterations,1 gpu
Helical Refinement NU	87h16min	13h20min			128,270	744 (fcrop 500)	symmetry imposed, symmetry search, non-uniform refinement, 15 iterations,1 gpu
Helical Symmetry Search	4min3sec	1min1sec				744 (fcrop 500)	symmetry search, rise, 1 gpu

Acknowledgements

NIH CIT

Nick Weber

Tee Gobezie

Thad Carlson

Gavin Brennan

AWS

Evan Bollig

Gargi Singh

Tom Fonseca

NIDDK

Jenny Hinshaw

Sarah Nyenhuis

NCI

Di Xia

Jingyu Zhan

Rick Huang

NIAID

Structural Virology Section

Ashish Kumar

Altaira Dearborn

Brandon Schweibenz

Khurts Shilagardi

RTB

Haotian Lei