Modeling Data Transfers: Change Point and Anomaly Detection

Cecilia Dao (Yale University)
Xinyu Liu (University of California, Berkeley)
Alex Sim, Craig E. Tull, K John Wu (Lawrence Berkeley National Laboratory)

Outline

- Goal
- Data
- Methods
- Results
- Conclusions

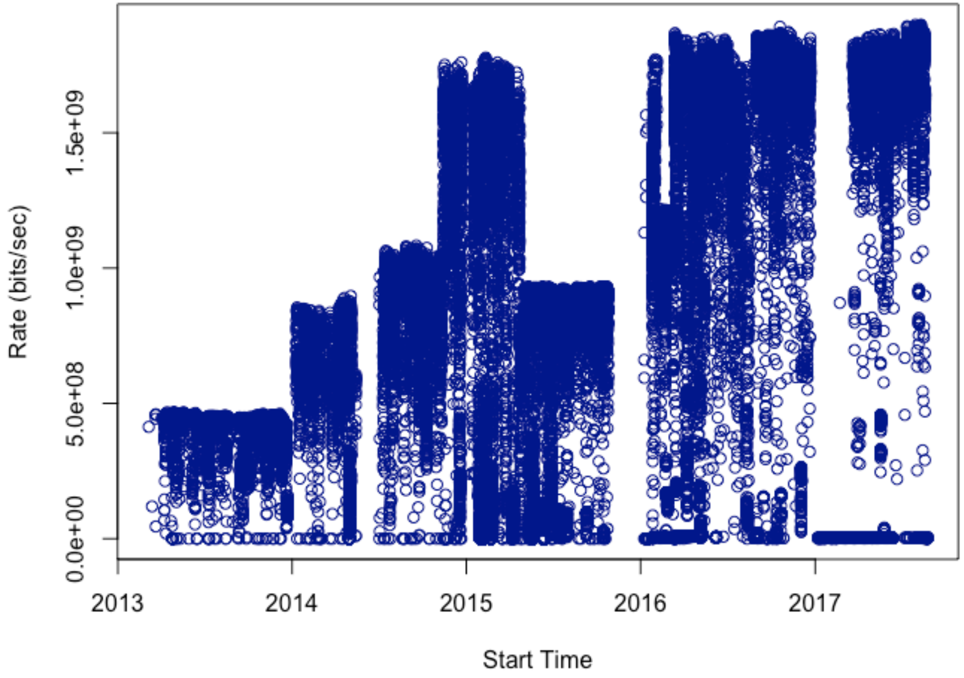
Goal

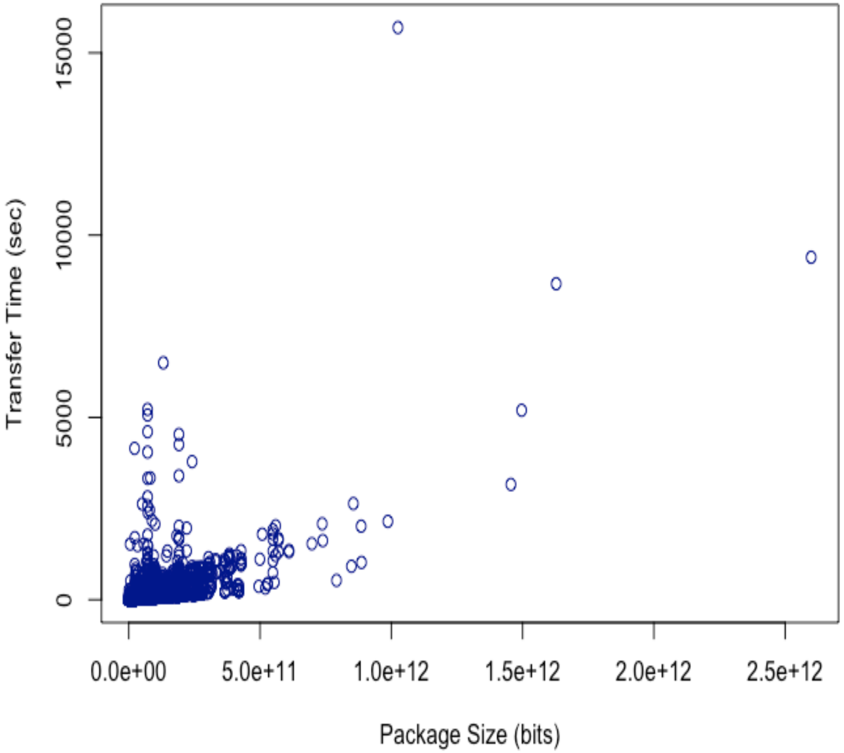
- Alert administrators when unusually slow transfers are detected in real-time
  - Model the time needed for transfering data files produced by the facility to a computer center
  - Predict a new file transfer time
  - Identify files with unusually slow transfers

DATA

- A single ALS beamline to NERSC(National Energy Research Scientific Computing Center) file transfer observations over from 2013 to 2017
- 39,761 observations with four features (Bundle (transfer file name), Size (bits), Start Time (of file transfer), and End Time (of file transfer))

bundle	packagedsize	whenstarted	whencompleted
20130127_111449_2nd_sample_no_heat_no_load	2815720176	2013-02-19 13:59:42.393-08	2013-02-19 14:00:31.582-08
20130127_114357_2nd_sample_T100C_no_load	2815721104	2013-02-20 12:00:49.163-08	2013-02-20 12:01:51.821-08
20130127_111449_2nd_sample_no_heat_no_load	2815720312	2013-03-06 14:36:17.64-08	2013-03-06 14:37:12.147-08
20130127_114357_2nd_sample_T100C_no_load	2815721240	2013-03-13 14:21:21.358-07	2013-03-13 14:24:30.1-07
20130127_121759_2nd_sample_T100C_L38lb	2815721296	2013-03-14 09:00:53.56-07	2013-03-14 09:01:44.329-07





Left plot shows rate has significant drop and rise
Right plot is size v.s. Transfer time plot and it is highly skewed

Test Data: SPOT Suite data transfer logs

Objective: identify anomalous data transfers

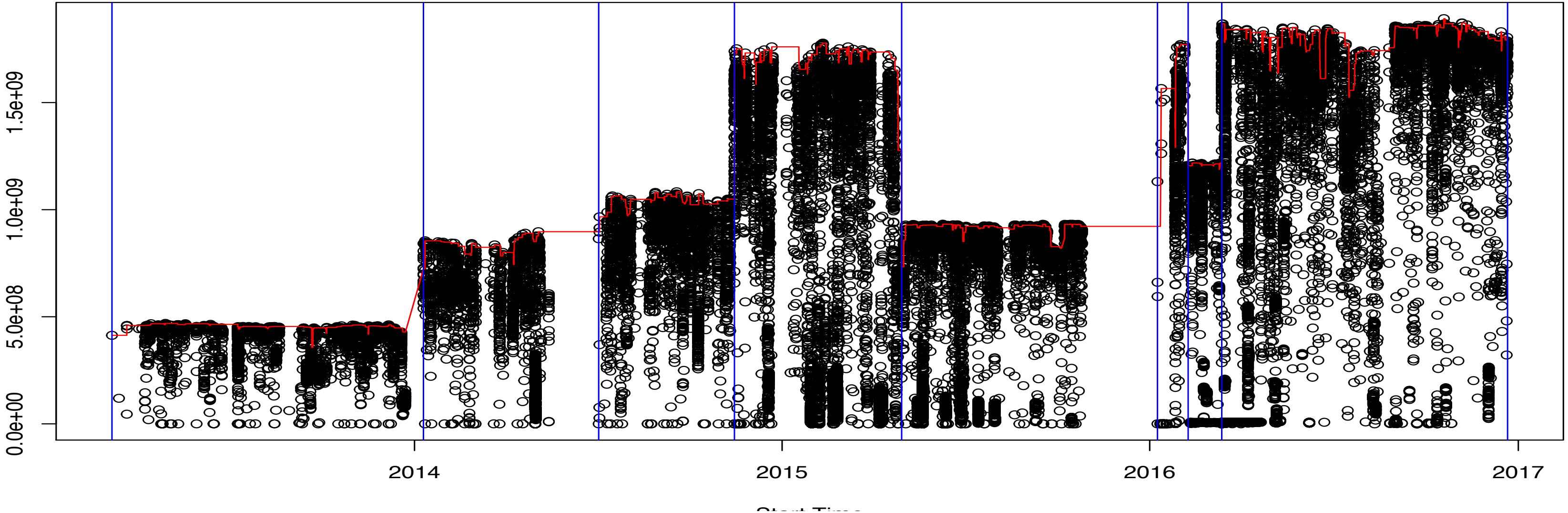
Approach: establish a statistical model of expected normal transfer time

Background Information

- SPOT Suite: (Craig)
- Data transfer from ALS to NERSC
- Information about file transfers: file size, start time, end time
- Logs from middle of 2013 to end of 2016
- ~35,000 file transfers

General approach

- During the time period, the network connection between ALS and NERSC has upgraded, and NERSC has moved: isolate these changes through change-point detection
- Within each relatively stable period between change points, develop statistical models for data transfer performance prediction



Methods

- Change Point Detection
- Prediction

Change Point Detection

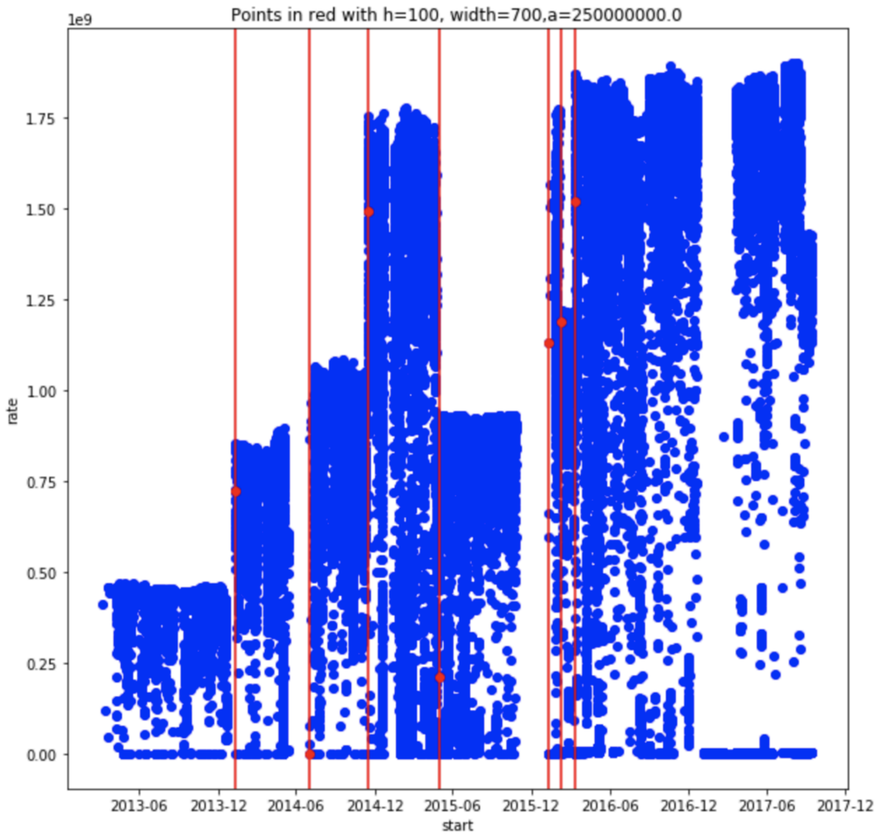
- Goal: Detect changes in the network connections, the storage systems or the computer systems
- Algorithm: Change Points are declared under two conditions
  - If the gap between the current file transfer i and the last file transfer i-1 has a gap larger than 21 days
  - With the moving maximum algorithm, the absolute change of two file transfers is greater than 2.5e+8 bits/s
- The moving maximum worked as follows:
  - Filter out files smaller than 8e+9 bits (1GB)
  - Keep the moving window of 100 files
  - Get the current maximum for this moving window-mmax_i
  - If abs (mmax_(i-1)-mmax_i)>2.5e+8: declare a change point
- Parameters are chosen by empirically applying different combinations to the dataset

Change Point Detection

```
for each file i do
    s_i ← transfer size for file i
    r_i ← transfer rate for file i
    if s_i ≤ s then
        Skip i
    end if
    if i ≤ h then
        Preparing for detection
    else
        M_i ← max(r_i, r_{i − 1}, ...r_{i − h})
        M_i − 1 ← max(r_{i − 1}, ..., r_{i − h − 1})
        if file i has more than break_day days apart from file
        i − 1 then
            declare i as a change point
        end if
        if abs(M_i − M_{i−1}) ≥ a and i ≥ ch_width then
            declare i as a change point
        end if
    end if
end for
```

Change Point Detection- Result

2014-01-09 23:47:11 - DTN OS upgrade
2014-07-03 01:12:12 - Resume ALS operation after network upgrade
2014-11-14 18:46:34 - Science DMZ network configuration upgrade
2015-04-29 20:34:42 - NFS-mounted file server shared with 2nd beamline DTN
2016-01-08 20:00:00 - 2nd beamline DTN disconnected from NFS file server
2016-02-08 07:02:14 - Unknown perturbation
2016-03-12 16:15:13 - Restore to baseline performance



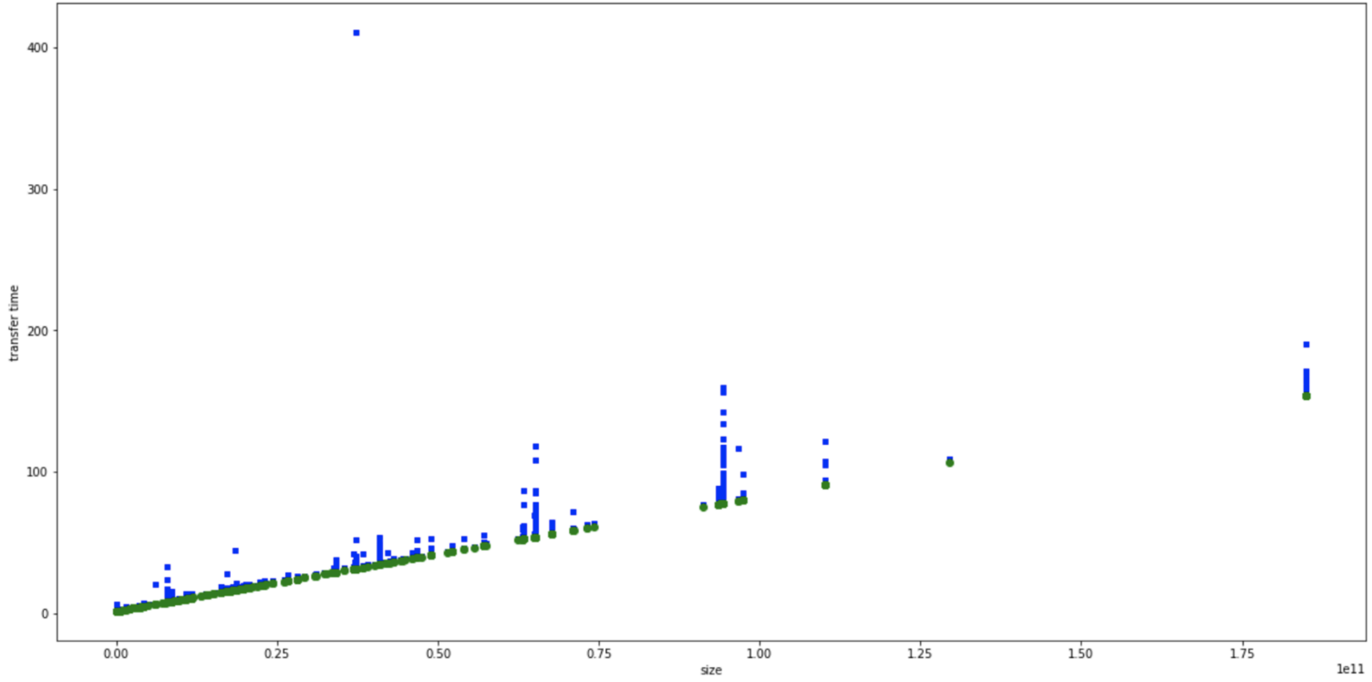
Prediction

- Goal: detect usually slow transfers within one segment identified by change point

- Breaking total file transfer time into two parts: base + congestion
  - Base time corresponds to the fastest transfer time, and is modeled with b-spline minimum quantile regression
  - Congestion time is the difference between base time and actual transfer time, and is modeled with non parametric kernel ridge regression with size as a covariate

Prediction-Baseline

Baseline prediction for segment 7(2016-02-08 to 2016-03-12)



Prediction-Congestion

Algorithm 1 Prediction

$y\{ij\} \leftarrow$ transfer time for file j in Segment i
$m\{ij\} \leftarrow$ baset time prediction for file j in Segment i
$\log_2(\alpha_{ij}) \leftarrow \log_2(y\{ij\}) - m\{ij\}$.
In non-parametric kernel regression, use covariates $\log_2$ of file size to predict $\log_2(\alpha_{ij})$
$\hat{y}_{ij} = \hat{m_i j} + \hat{\alpha}_{ij}(\log_2(\text{Size}_j))$
$\hat{\epsilon}_{ij} = y\{ij\} - y\{\hat{ij}\}$
if $\hat{\epsilon}_{ij} \geq 99$ *percent quantile of* $\hat{\epsilon}$ **then**
  file j is an anomaly
end if

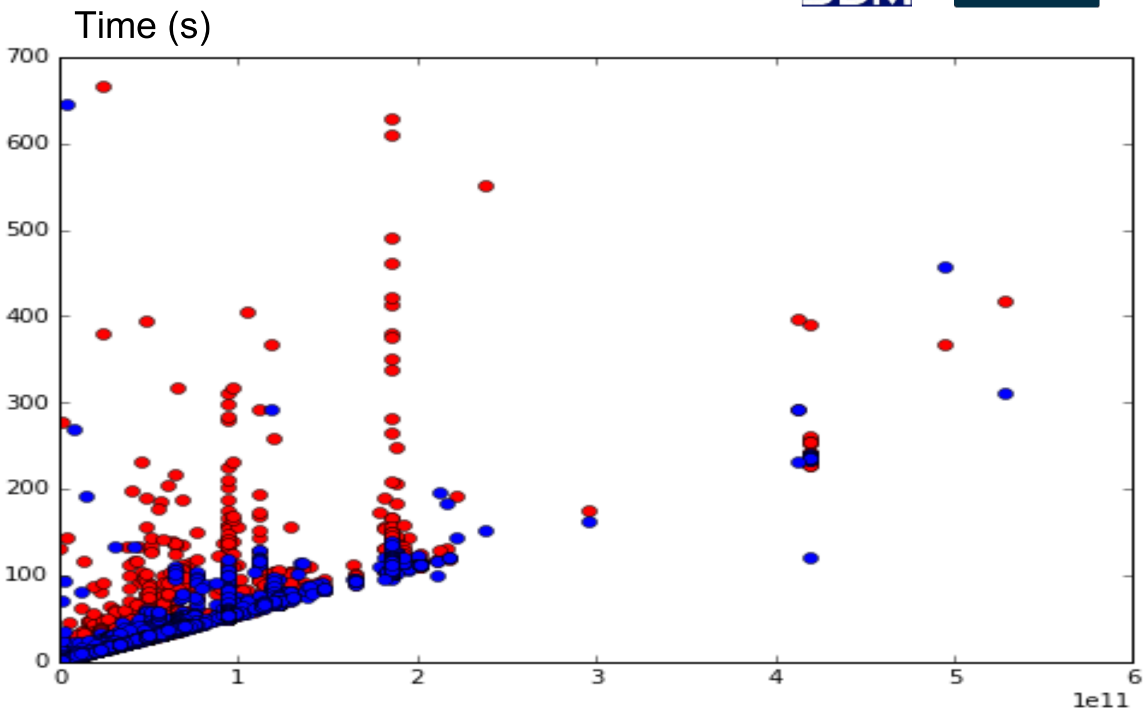
Predicting File Transfer Time: moving non-parametric kernel regression

Two classes are

- **Base**: minimum quantile of spline regression line for the best case transfer time (dependent variable file size)
- **Noise**: model the congestion and other delays as an additive noise on top of the base predictions (dependent variables time and file size)

Rationale

- The base line captures the expected data transfer time on a normal system condition (e.g. uncongested network link and storage link), typically regarded as a linear function of file size
- We model the deviation from the base line as a random noise that could capture the daily and weekly patterns as well as recent trends



Red dots: actual measurements
Blue dots: predicted time

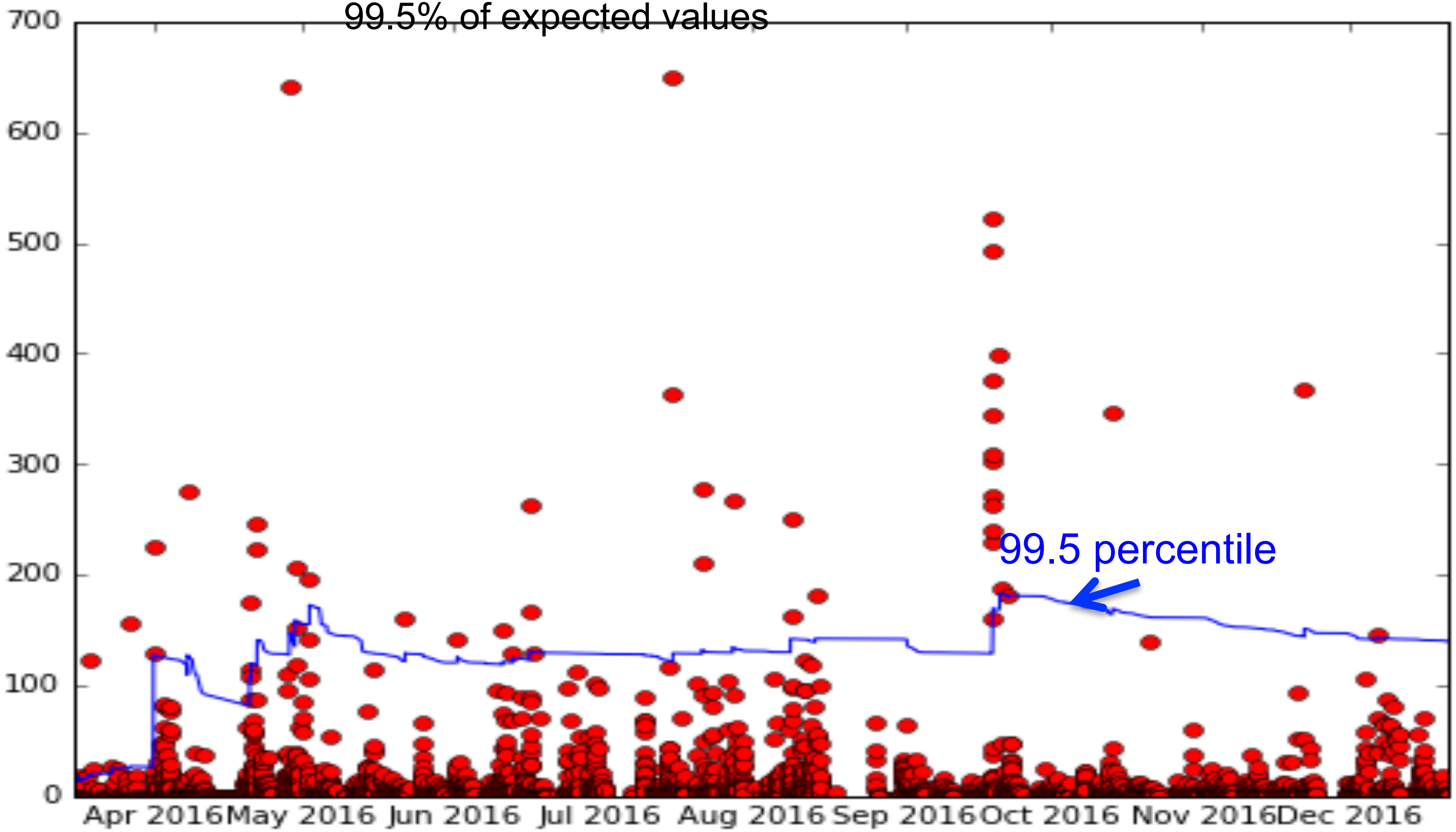
Combined two algorithm

1. Given a new file transfer, check if it is a change point first
2. if not, use prediction algorithm to check it if it is an anomaly point
3. If it is a change point, we will restart the prediction and refit models.

Anomaly Detection: counting points with large noise values

- Predict the expected noise values
- Predict the expected range of noise values
- Draw a threshold line for abnormal cases, say, when noise is above 99.5% of expected values

Noise (time in seconds)

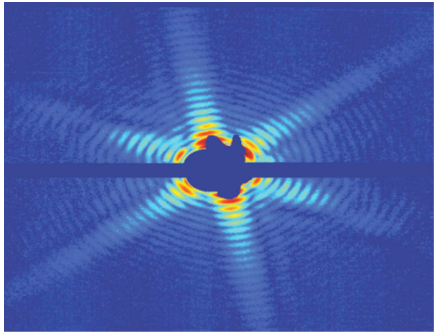


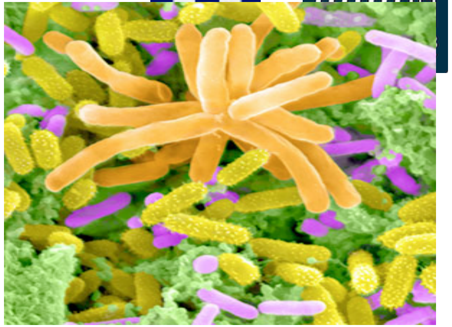
Start time

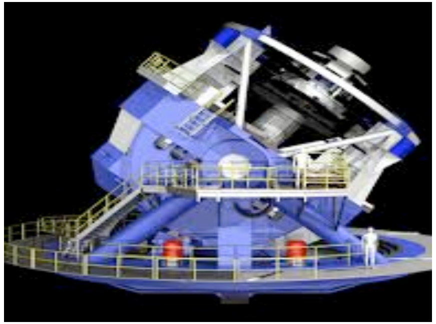
Summary

- Developed a new change point algorithm based on moving max
- Implemented many variations of this change point algorithms and verified change points detected
- Developed a statistical model for prediction data transfer time
- Combined the change point detection algorithm and the transfer time prediction algorithm into a single python code
- Working on turning the python code into an automated procedure

X-Swap
Extreme-Scale Scientific Workflow Analysis and Prediction







X-SWAP PROJECT

Erich Strohmaier,
Cecilia Dao, Kjiersten Fagnan, Devarshi Ghoshal, Wilko Kroeger,
Fritz Mueller, Peter Nugent, Amedeo Perazzo, Eric Pouyoul,
Vaikunth Thukral, Andrew Tritt, Craig E. Tull, John Wu

BERKELEY LAB
Lawrence Berkeley National Laboratory

SLAC
NATIONAL ACCELERATOR LABORATORY

Thank you!

Questions?