Fusing Intelligence into Big Data Transfer in High-performance Networks

Chase Wu
Department of Computer Science
New Jersey Institute Technology
Newark, NJ 07102, USA

The 18th International Conference on Wireless Networks and Mobile Systems (WINSYS)
July 8, 2021

YING WU COLLEGE OF COMPUTING

NJIT New Jersey Institute of Technology

Outline

- Introduction
  - Center for Big Data at NJIT
  - Big data transfer in high-performance networks and wireless networks
- Exploratory Analysis of Performance Measurements
- Performance Modeling Using Machine Learning
- Experimental Results of Performance Prediction
- Conclusion

Center for Big Data

Director, Chase Wu

https://centers.njit.edu/bigdata/

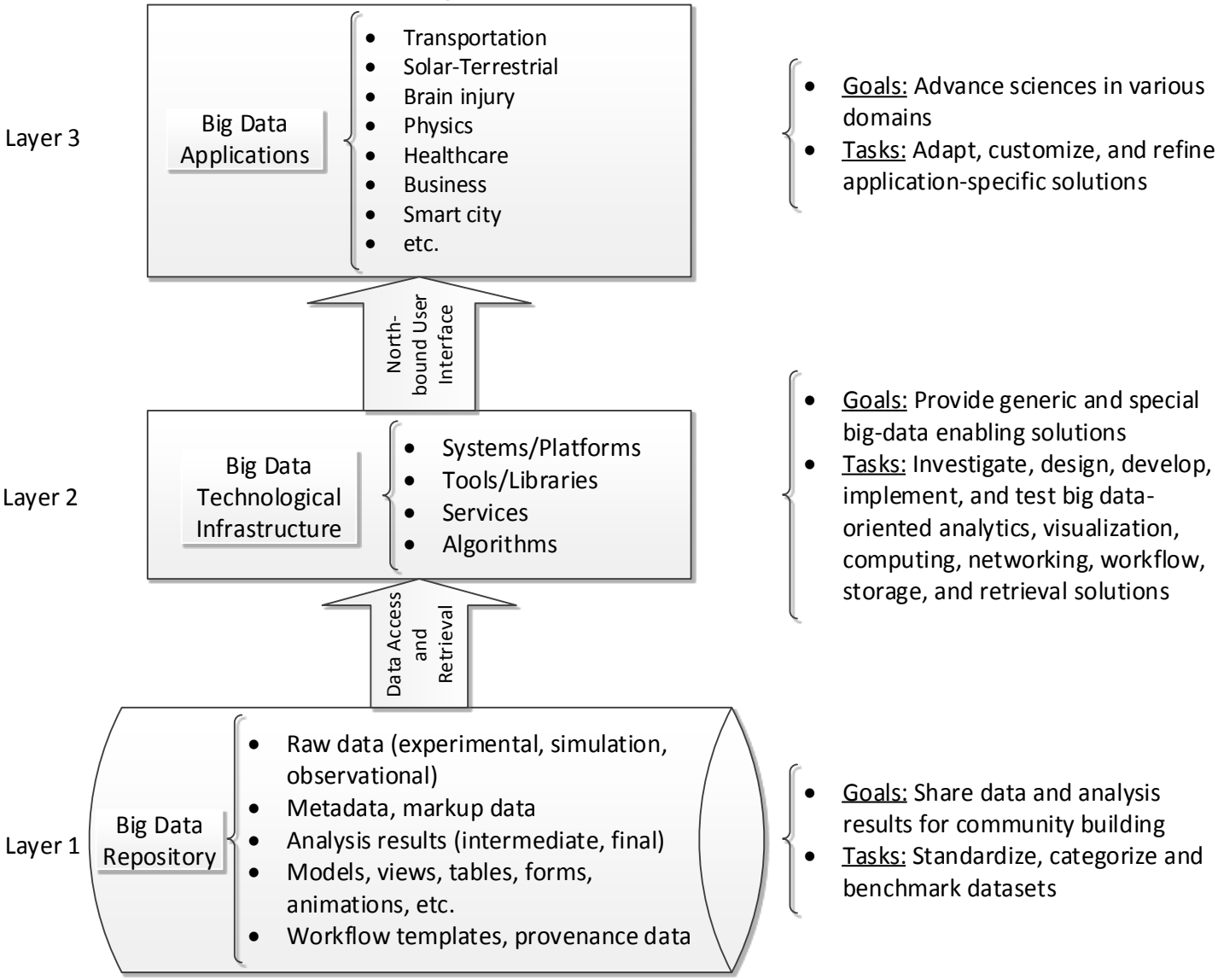
Location: GITC 4416, NJIT, Newark, NJ, USA

Mission Statement of CBD at NJIT

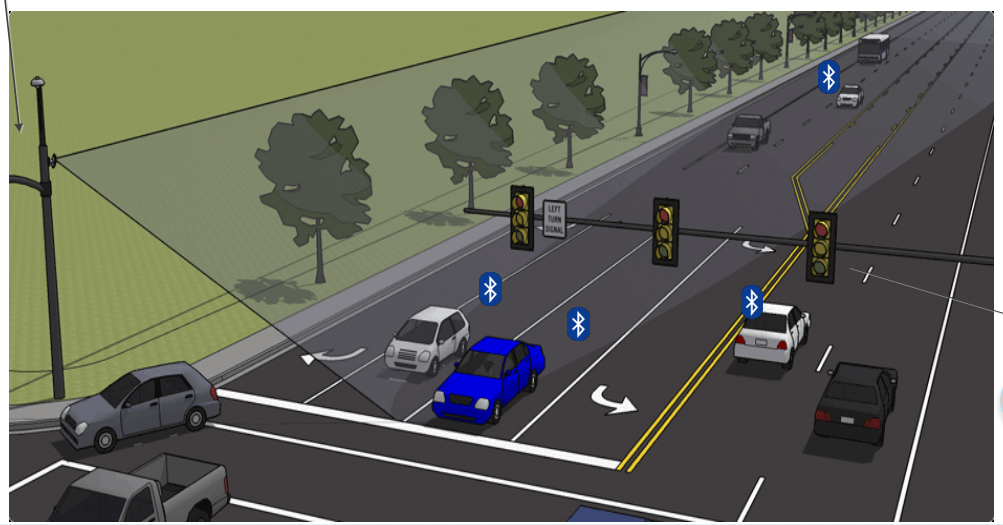
- Synergize the strong expertise in various disciplines across the NJIT campus
- Build a unified big data platform that embodies a rich set of big data enabling technologies and services with optimized performance to facilitate research collaboration and scientific discovery
- Investigate, develop, and apply cutting-edge technologies to address unprecedented challenges in big data with **high Volume, high Velocity, high Variety, and high Veracity**,

  in order to create **high VALUE!**

A Three-layer Structure of the CBD



Application 1: Transportation

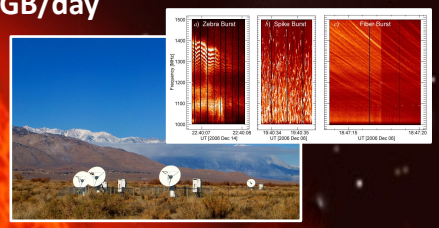


Big Data Challenges:

- Standardization of data format
- Accurate modeling
- Clustering and classifying
- Integrating data from independent sources
- Uncovering patterns, correlation, etc.
- Interpretation

Application 2: Solar Terrestrial Research

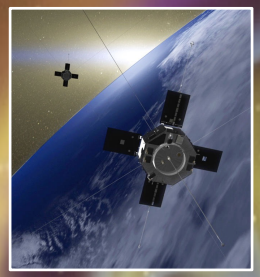
OVSA: 50 GB/day

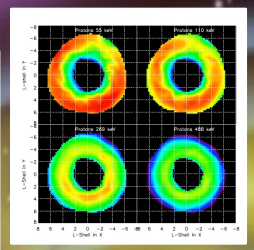


SWRL: 10 GB/day

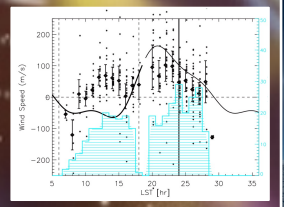
BBSO: 6000 GB/day

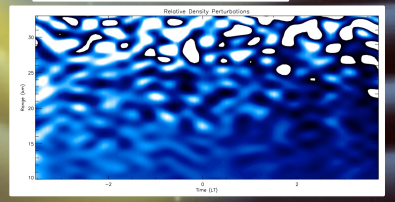
Van Allen Probes: 2GB/day





Jeffer Lidar





Other: 0.25 GB/day

PEDC/Antarctic: 0.5 GB/day

Big Data Challenges:

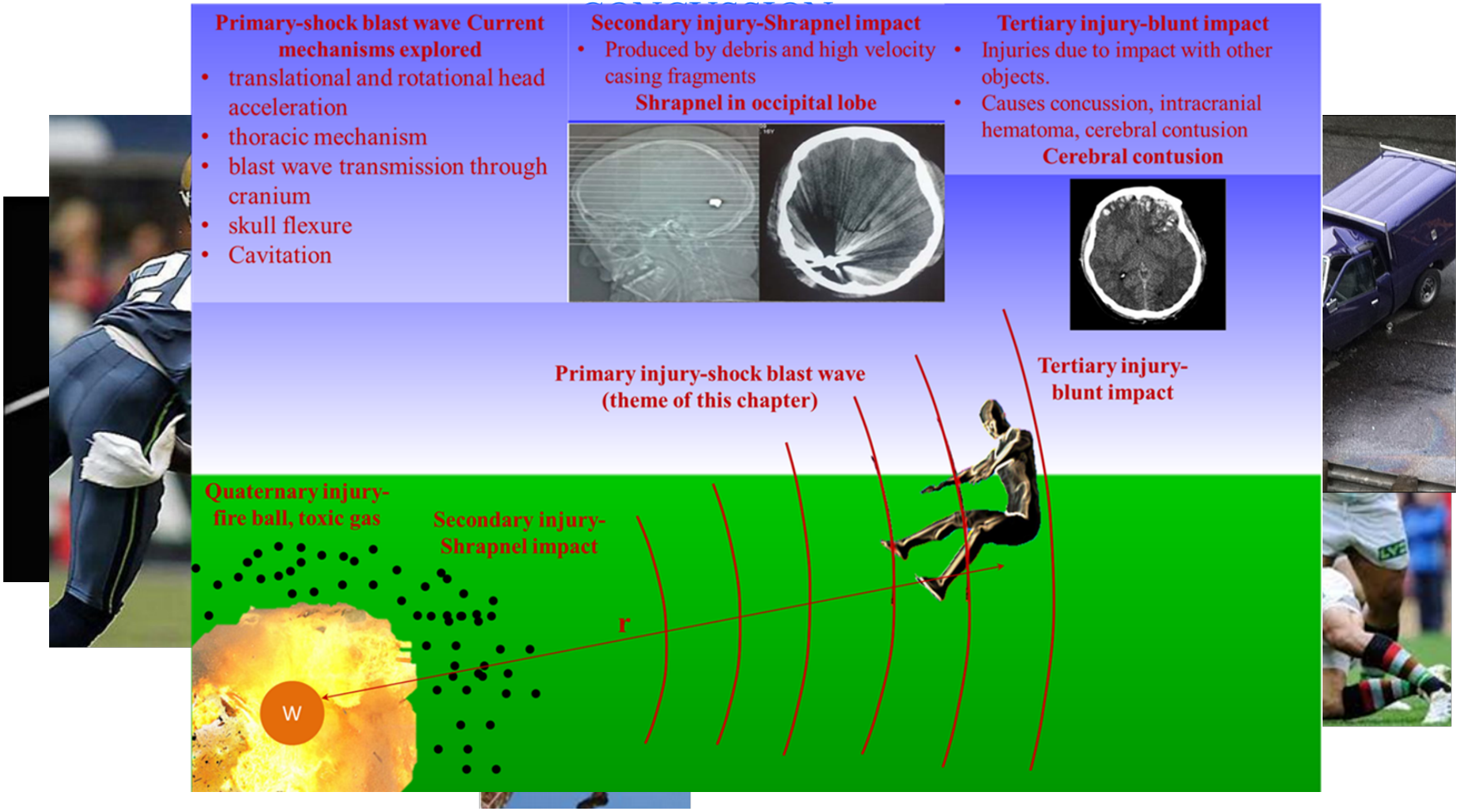
- Complex Process: Plasma Physics + Fluid Dynamics
- Expensive Equipment: Remote Sensing/Instrumentation
- Data Reduction and Inversion
- Modeling (?)

Application 3: Brain Injury Research

Ballistic (bullet)

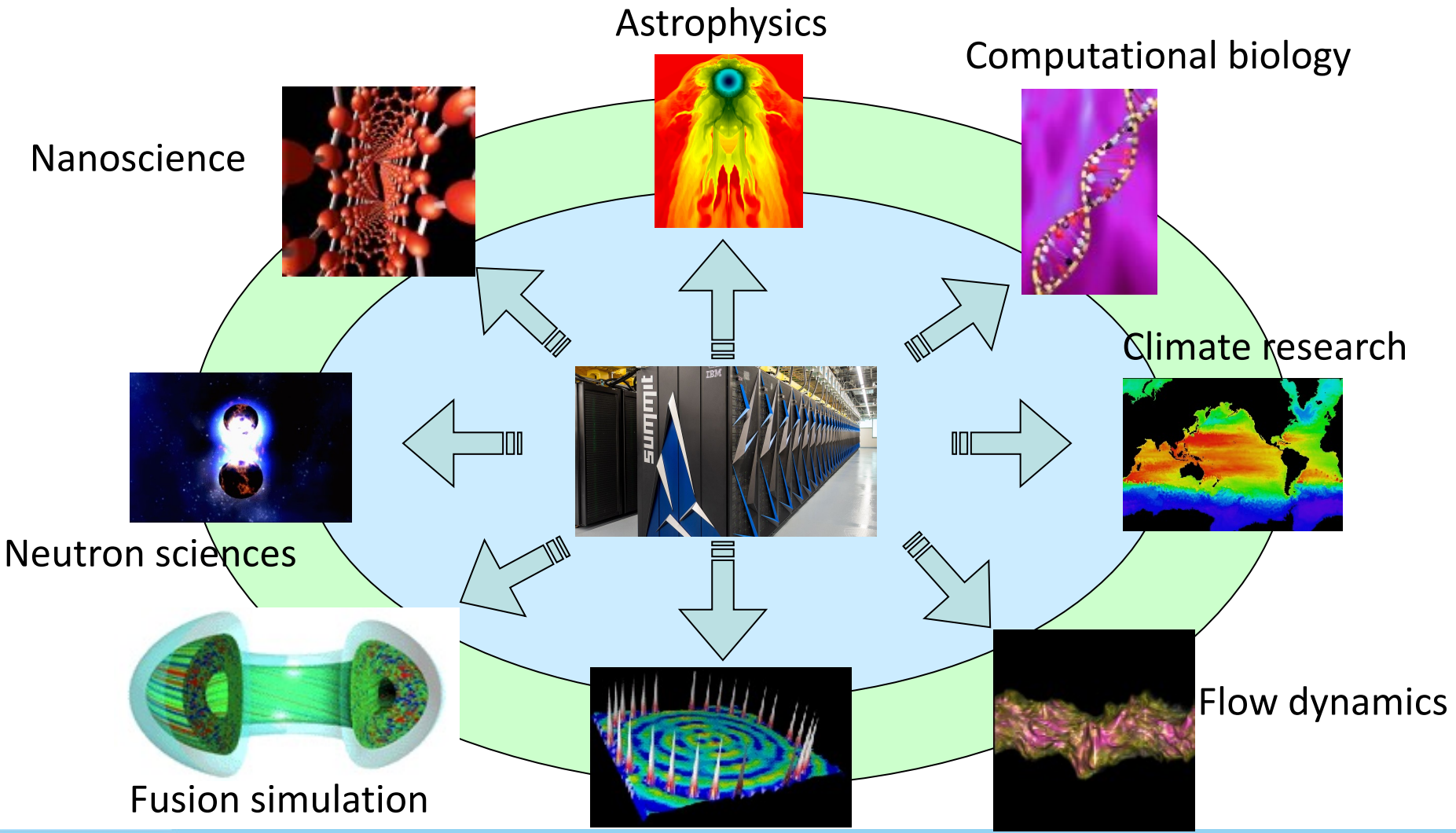
Blunt Injury-most prevalent
Blunt Impacts>> MVA, fall, sports injury

Blast (military)



Introduction

- Supercomputing for big-data science



Networking for Big-data Applications

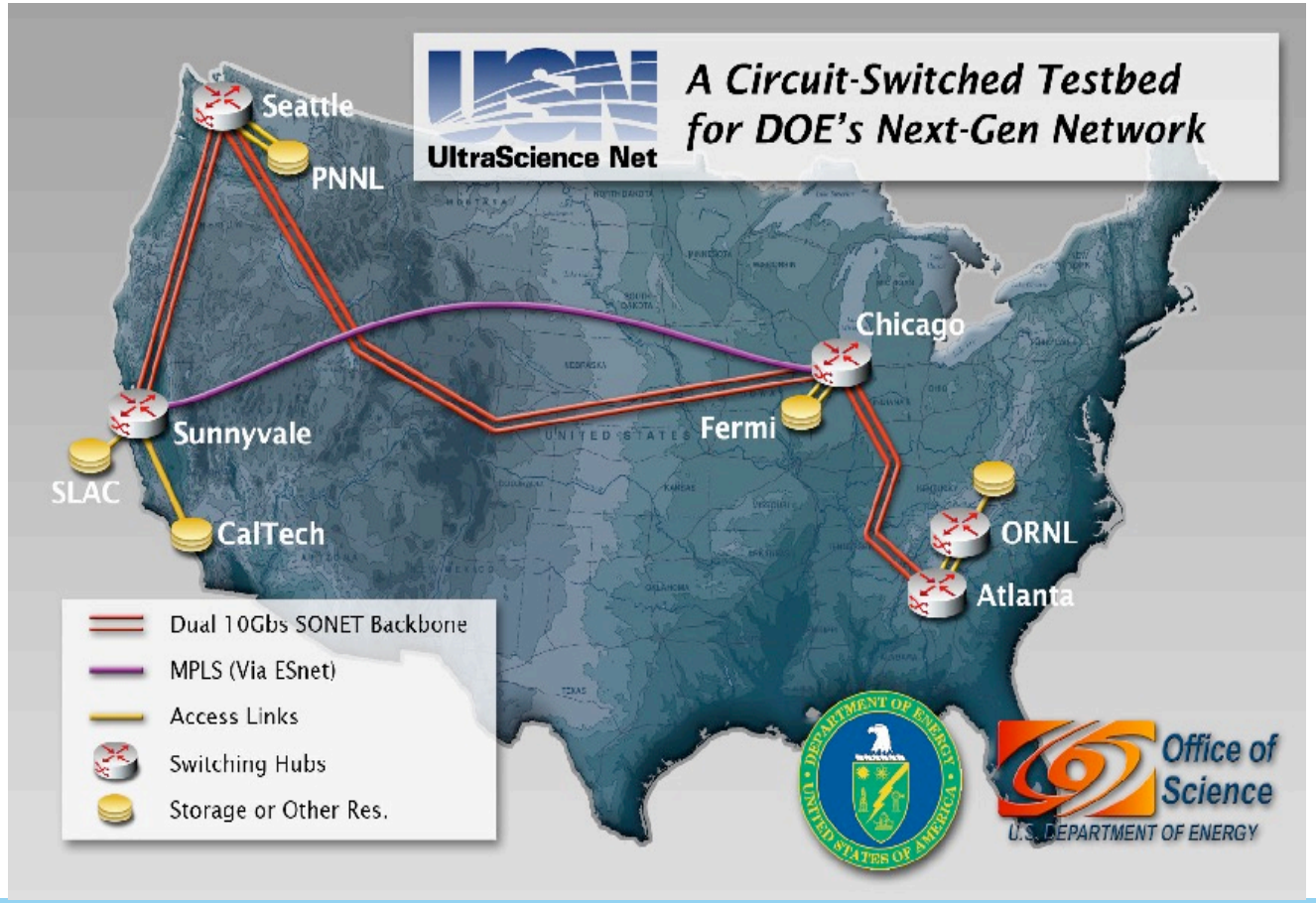
- Networking requirements
  - High bandwidth
    - Multiples of 10Gbps to terabits networking
    - Support bulk data transfer
  - Stable bandwidth
    - 100s of Mbps
    - Support interactive control and steering operations
- Limitations of the Internet
  - Only backbone has high bandwidths (last mile)
  - Complicated dynamics
    - Packet-level resource sharing
    - Best-effort IP routing
  - TCP: hard to sustain 10s Gbps or to stabilize

High-performance Networks

- Provision dedicated channels
  - UltraScience Net
  - ESnet OSCARS
    - Offers MPLS tunnels and VLAN virtual circuits
  - Internet2 ION
    - Offers MPLS tunnels and VLAN virtual circuits
  - UCLP
    - User Controlled Light Paths
  - CHEETAH
    - Circuit-switched High-speed End-to-End Transport Architecture
  - DRAGON
    - Dynamic Resource Allocation via GMPLS Optical Networks

UltraScience Net – In a Nutshell

- Experimental Network Research Testbed



Big Data Transfer in Wireless Networks

5G quotes 300 Mb/s of downlink, 50 Mb/s of uplink, an end-to-end latency of 10 milliseconds.

6G enables a peak rate of 1,000 Gb/s and air latency less than 1,000 microseconds.

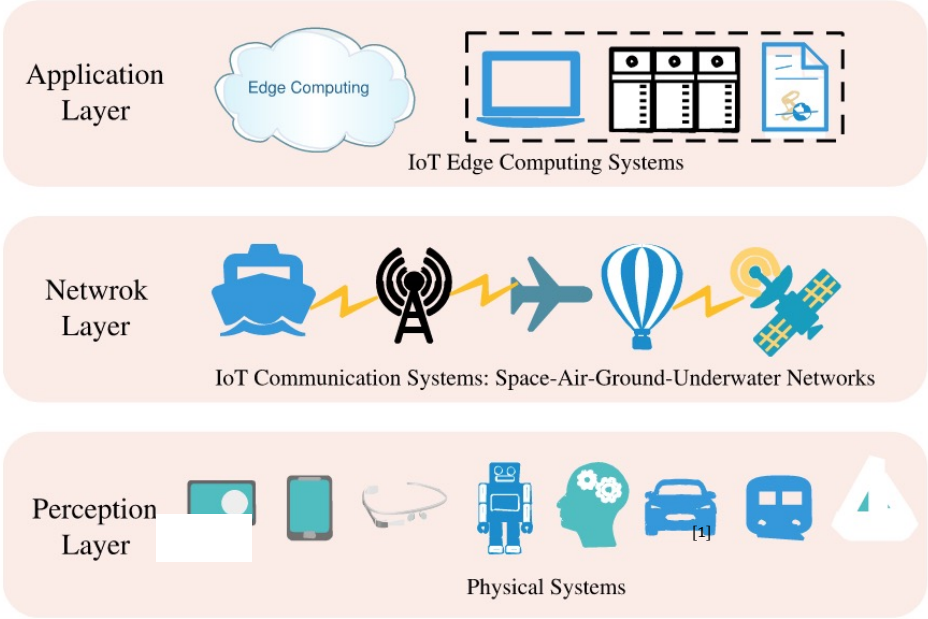
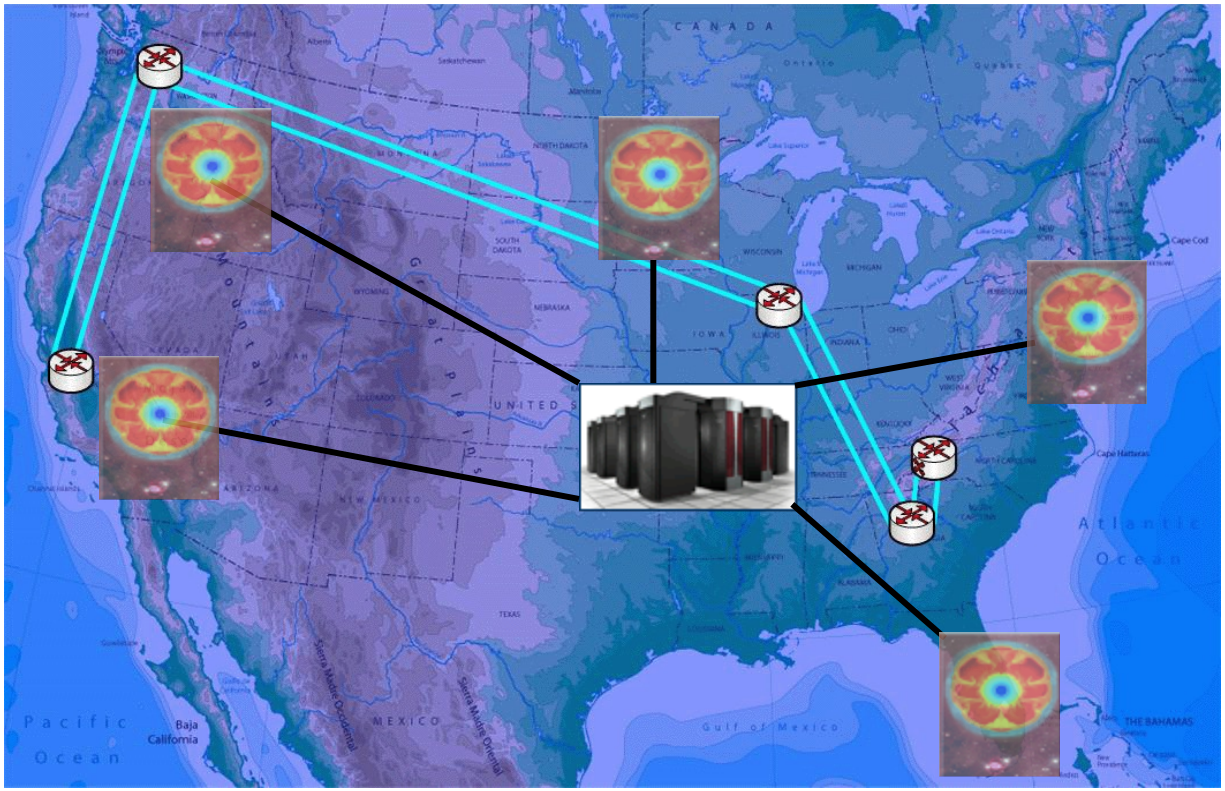


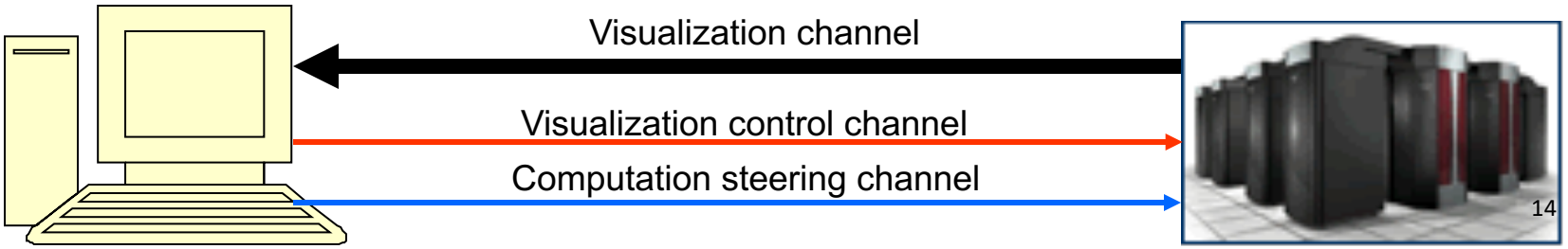
Fig. 5: A typical architecture of IoT systems.

[1] F. Guo and et al. 2021. Enabling Massive IoT Toward 6G: A Comprehensive Survey. Internet of Things Journal.

Terascale Supernova Initiative (TSI)

- Collaborative project
  - Supernova explosion
- TSI simulation
  - 1 terabyte a day with a small portion of parameters
  - From TSI to PSI to ESI
- Transfer to remote sites
  - Interactive distributed visualization
  - Collaborative data analysis
  - Computation monitoring
  - Computation steering





Reserve Ahead, but How Much?

Bandwidth reservation requires performance modeling and throughput prediction to avoid over-provisioning and under-provisioning!

Impact Factors in Big Data Transfer

Throughput performance of big data transfer is affected by many factors:

1. End host configuration
   - CPU frequency
   - Number of processors

2. Network connection properties
   - Bandwidth
   - RTT
   - Loss rate

3. Data transfer methods and corresponding control parameters
   - Packet size
   - Block size
   - Buffer size
   - Number of streams

Exploratory Analysis

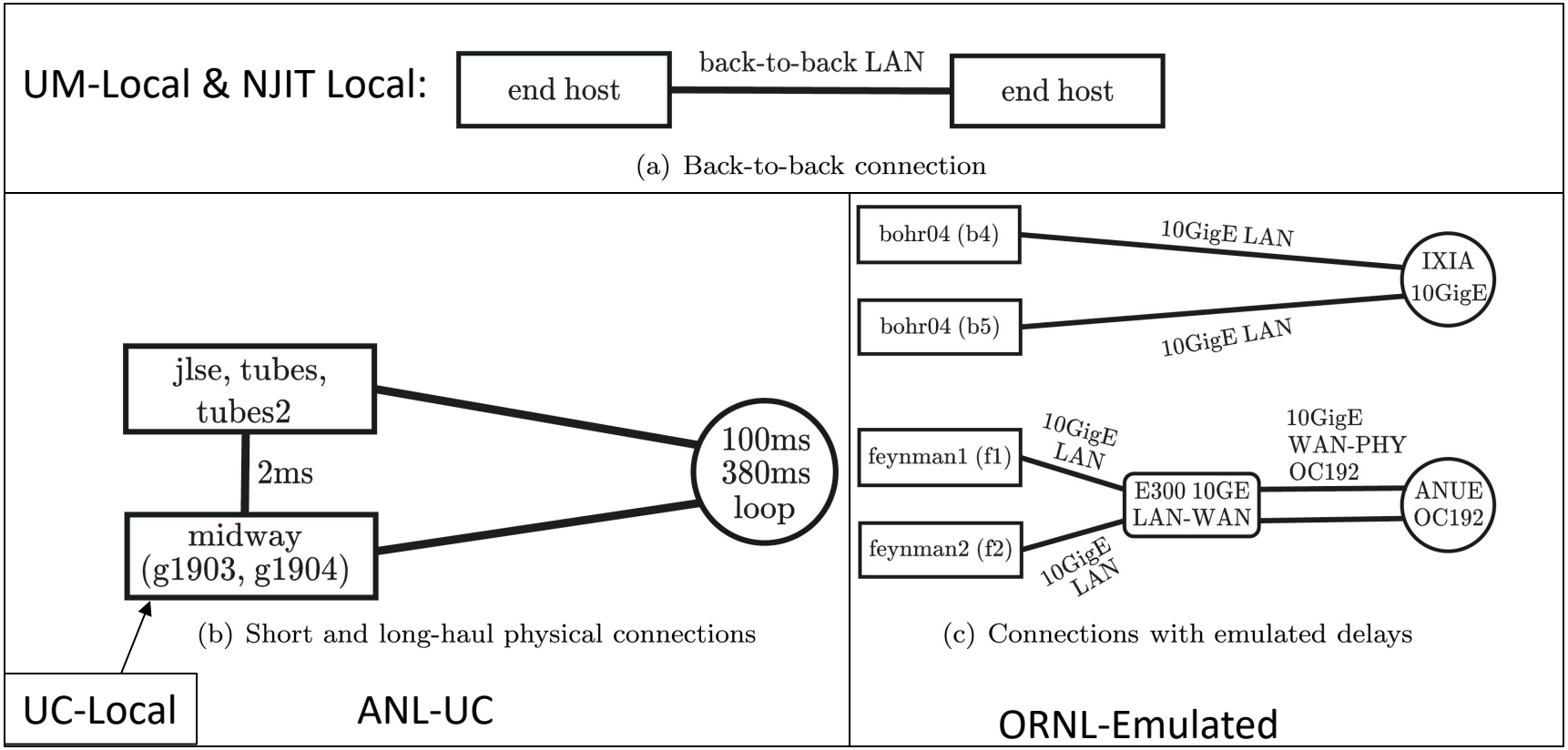
Conduct exploratory analysis of a subset of hyperparameters, including:

- Packet size
- Buffer size
- Block size
- Number of parallel streams
- Protocol type
- Connection delay
- End-host settings

TABLE I
LIST OF ATTRIBUTES OF THE THROUGHPUT PERFORMANCE DATASET.

Categories	Attributes	Remarks
Identifier	record ID	integer, unique
Testbed	testbed	string, nominal
End host	CPU frequency	double, hertz
	# of processors	integer
	# of cores per processor	integer
	memory size	integer, MB
	kernel buf size	integer, byte
Connection	bandwidth	double, Gbps
	RTT	double, ms
	loss rate	double, emulated
Toolkits & Protocols	data transfer protocol	string, nominal
	fata transfer toolkit	string, nominal
Control parameters	frame size	integer, byte
	packet size	integer, byte
	payload size	integer, byte
	block size	integer, byte
	TCP send buf size	double, MB
	TCP recv buf size	double, MB
	UDP send buf size	double, MB
	UDP recv buf size	double, MB
	UDT send buf size	double, MB
	UDT recv buf size	double, MB
	# of data streams	integer
	data size	double, MB
	time duration	integer, seconds
Performance	throughput/goodput	double, Mbps

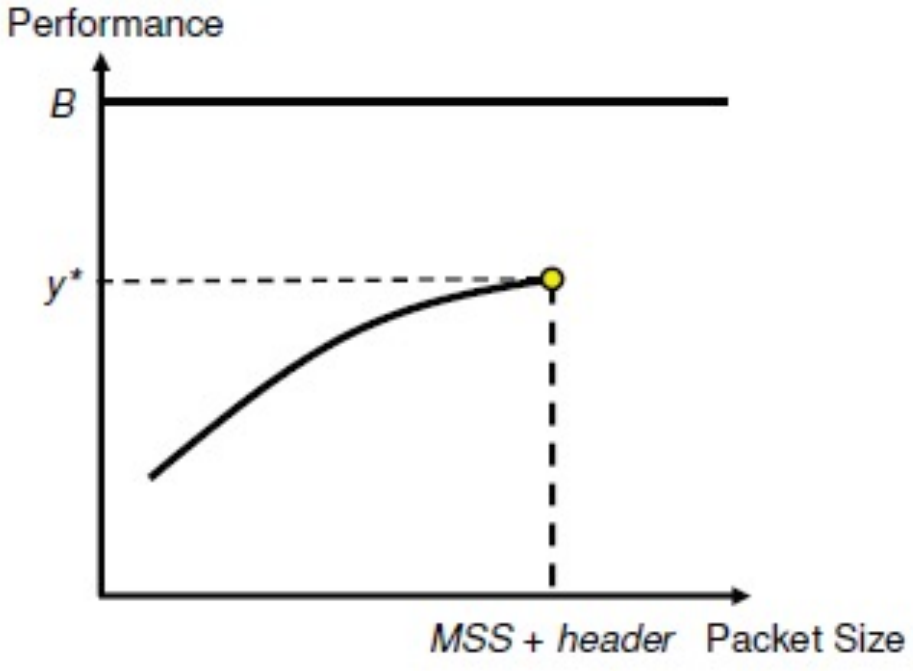
Network Infrastructures for Big Data Transfer Experiments



Effects of Packet Size on Throughput Performance

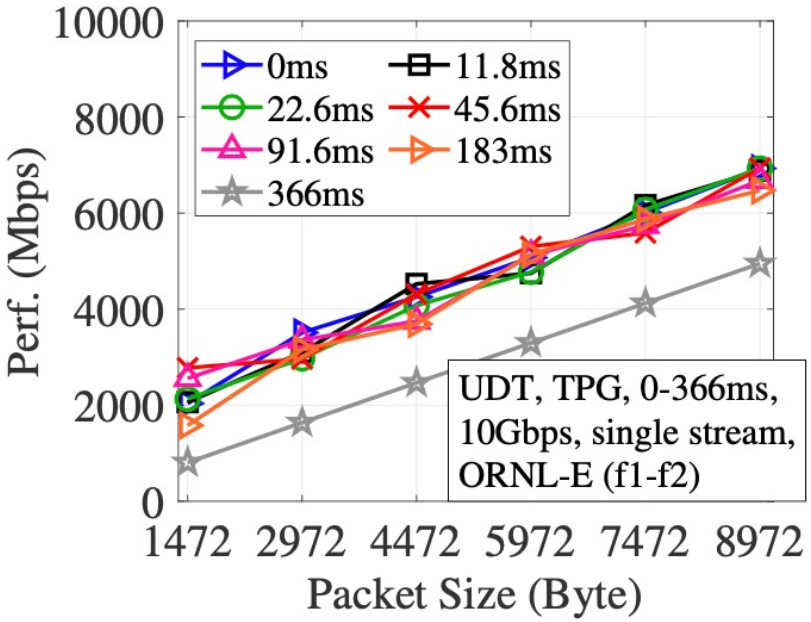
Intuitive analysis:

- A larger packet size benefits the performance since it carries more per-packet user payload and reduces per-packet processing overhead.

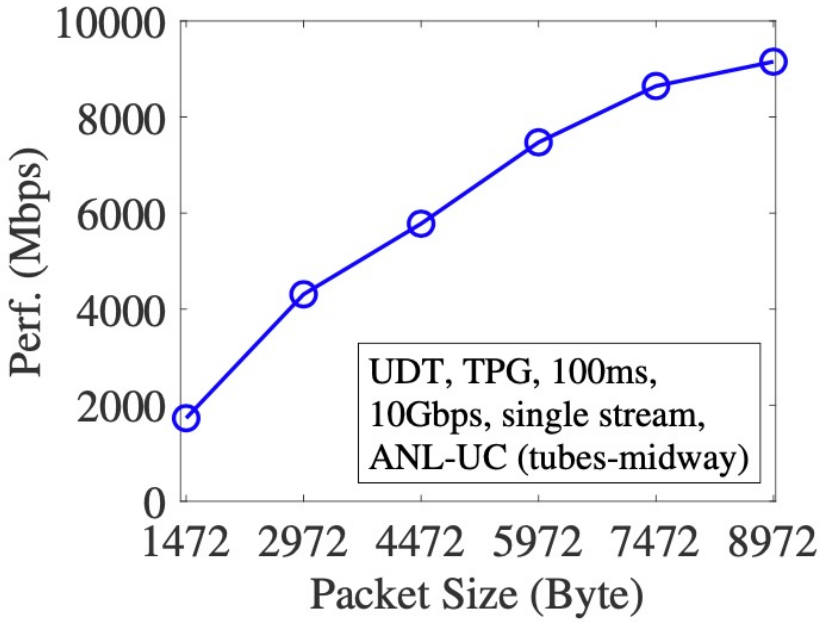


Expected Trace

Effects of Packet Size on UDT performance



(a) ORNL-E, 0–366ms, 10 Gbps

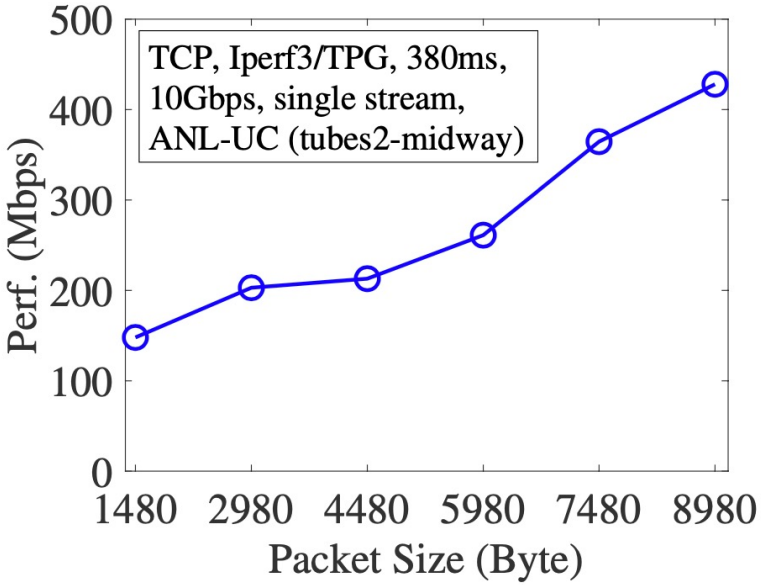


(b) ANL-UC, 100ms, 10 Gbps

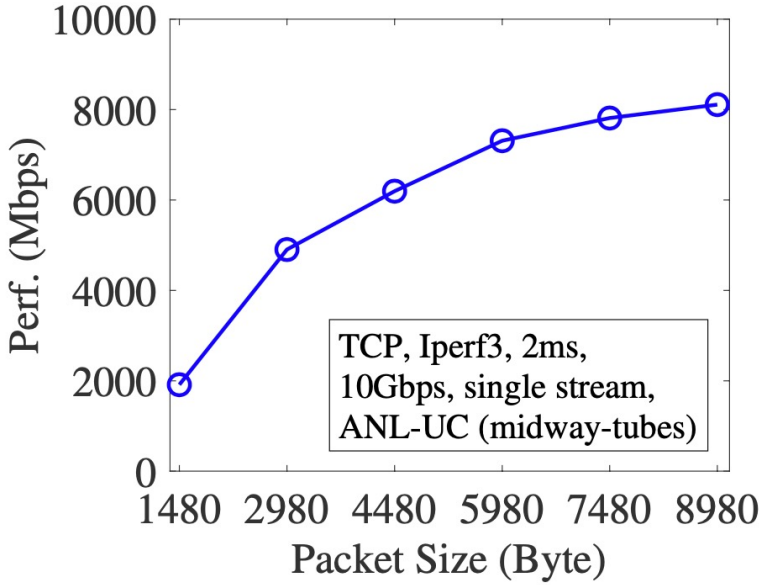
Observations:

- When other parameters such as buffer size are fixed, the performance almost linearly increases with packet size, but at a lower speed when buffer size is sufficiently large.

Effects of Packet Size on TCP performance



(a) ANL-UC, 380ms, 10 Gbps



(b) ANL-UC, 2ms, 10 Gbps

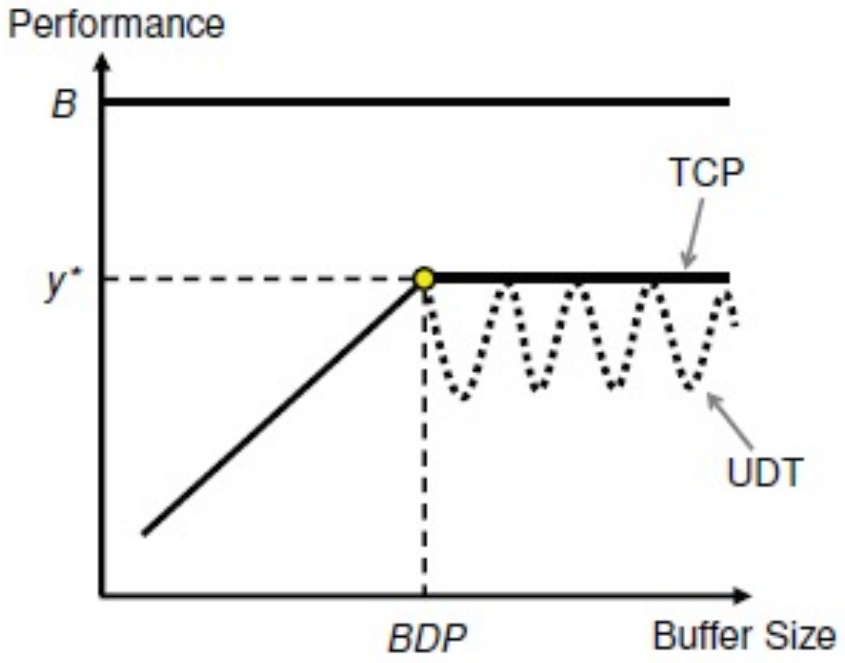
Observations:

- The increasing pattern of performance is also consistent when using other transfer protocols such as the widely used TCP.

Effects of Buffer Size on Throughput performance

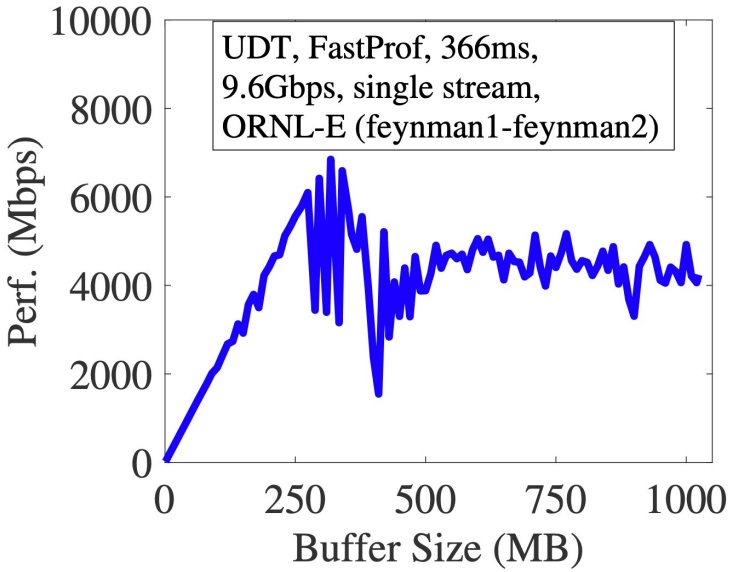
Intuitive analysis:

- A buffer size no less than the Bandwidth-Delay Product (BDP) is required to saturate the connection capacity.
- The peak achievable performance y^* is lower than the overall peak achievable performance and is limited by other factors.

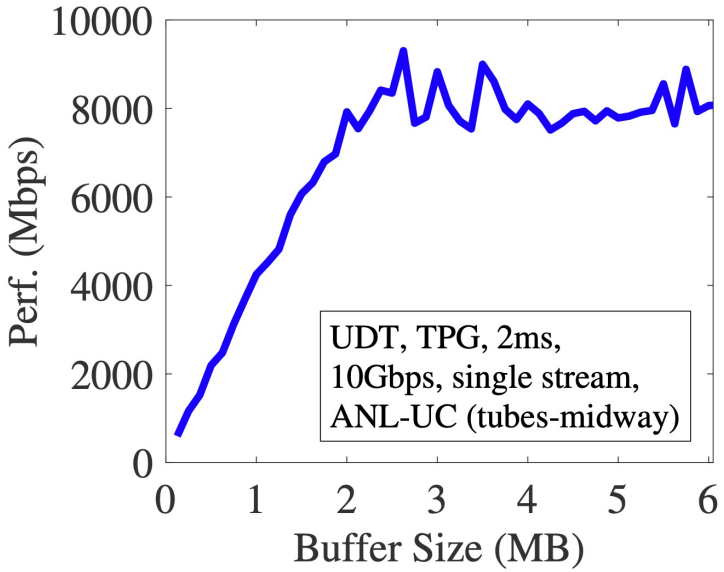


Expected Trace

Effects of Buffer Size on UDT performance



(a) ORNL-E, 366 ms, 9.6 Gbps

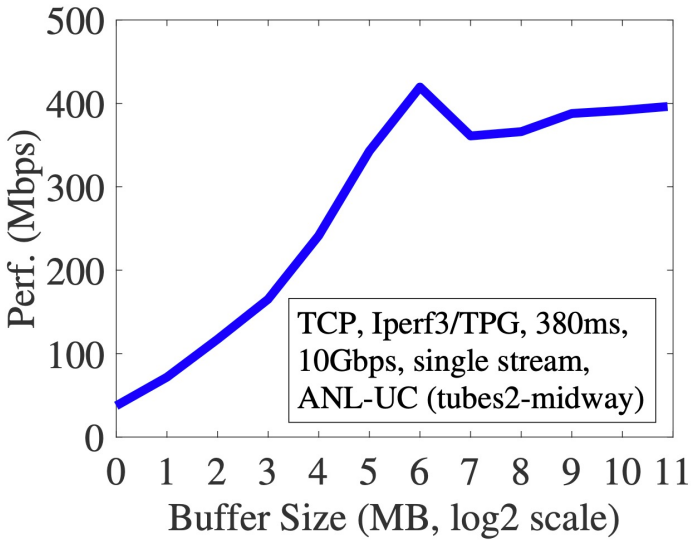


(b) ANL-UC, 2 ms, 10 Gbps

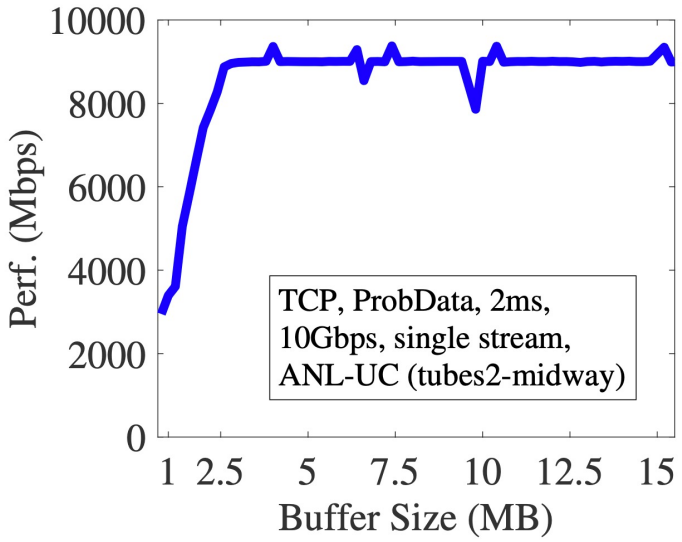
Observations:

- In the region where buffer size is relatively small (e.g., less than BDP), the performance almost linearly increases as buffer size increases.
- As buffer size increases up to around the BDP, other factors start to impose further limitation on the performance.

Effects of Buffer Size on TCP performance



(a) ANL-UC, 380 ms, 10 Gbps

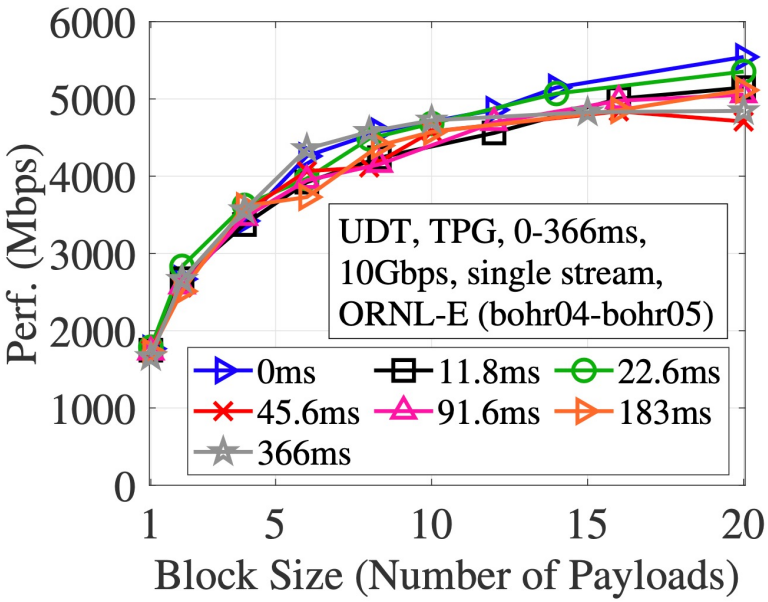


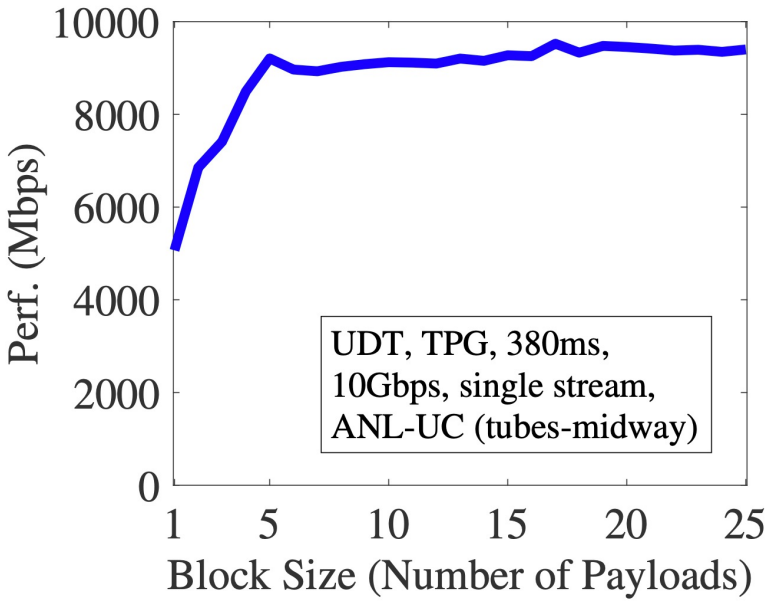
(b) ANL-UC, 2 ms, 10 Gbps

Observations:

- TCP and UDT behave similarly, and the performance almost linearly increases as buffer size increases.
- The slope of the linear increase depends on end host configurations and network properties.

Effects of Block Size on UDT Performance



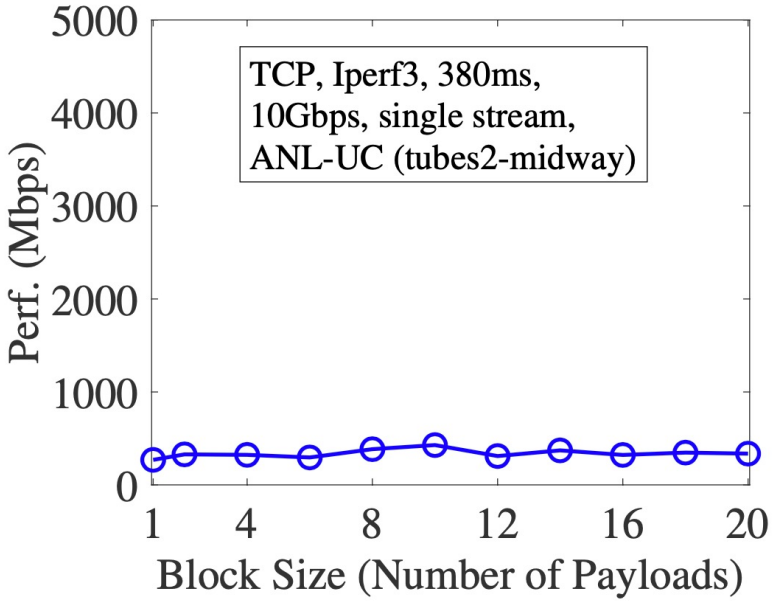


Observations:

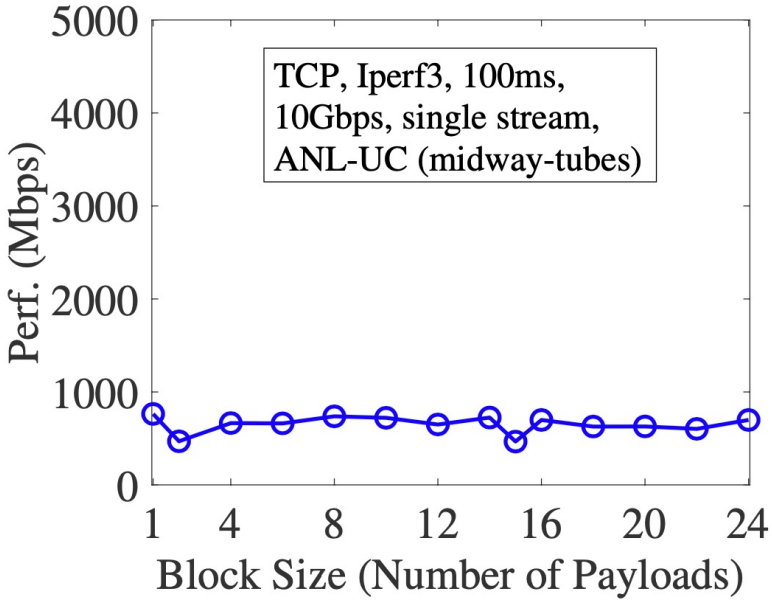
- UDT performance increases with block size given a sufficiently large buffer.

- After the block size reaches a certain point, the improvement brought by enlarging data block becomes marginal, and then stabilizes at a peak.

Effects of Block Size on TCP performance



(a) ANL-UC, 380 ms, 10 Gbps

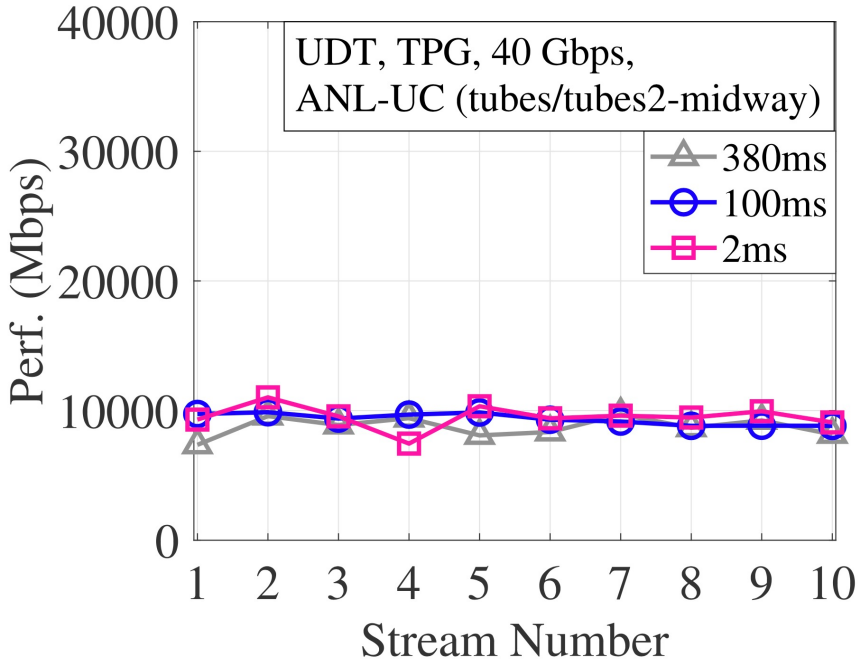


(b) ANL-UC, 100 ms, 10 Gbps

Observations:

- TCP performance is not significantly affected by block size, and the stabilized performance is mainly determined by other factors such as buffer size

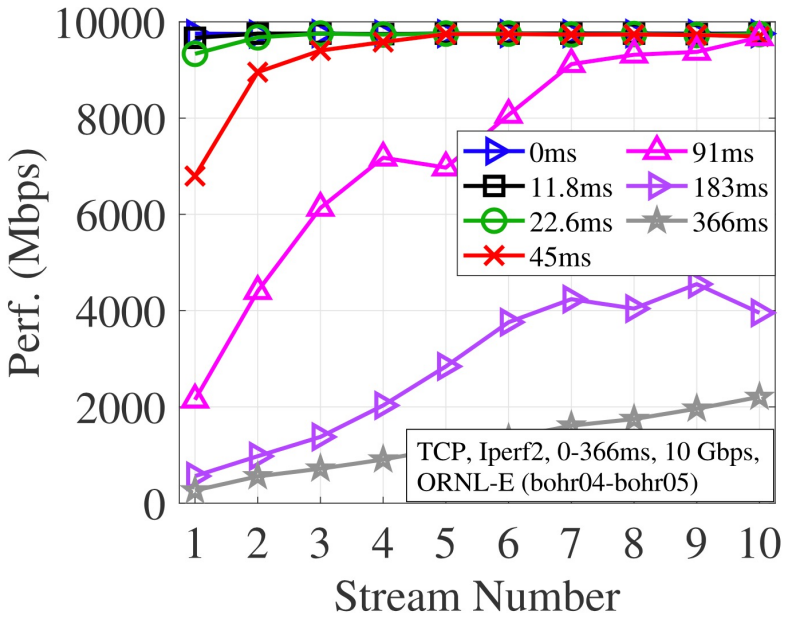
Effects of Stream Count on UDT performance



Observations:

- The UDT performance is expected to be insensitive to the number of streams since it is not designed for environments with high concurrency.

Effects of Stream Count on TCP Performance



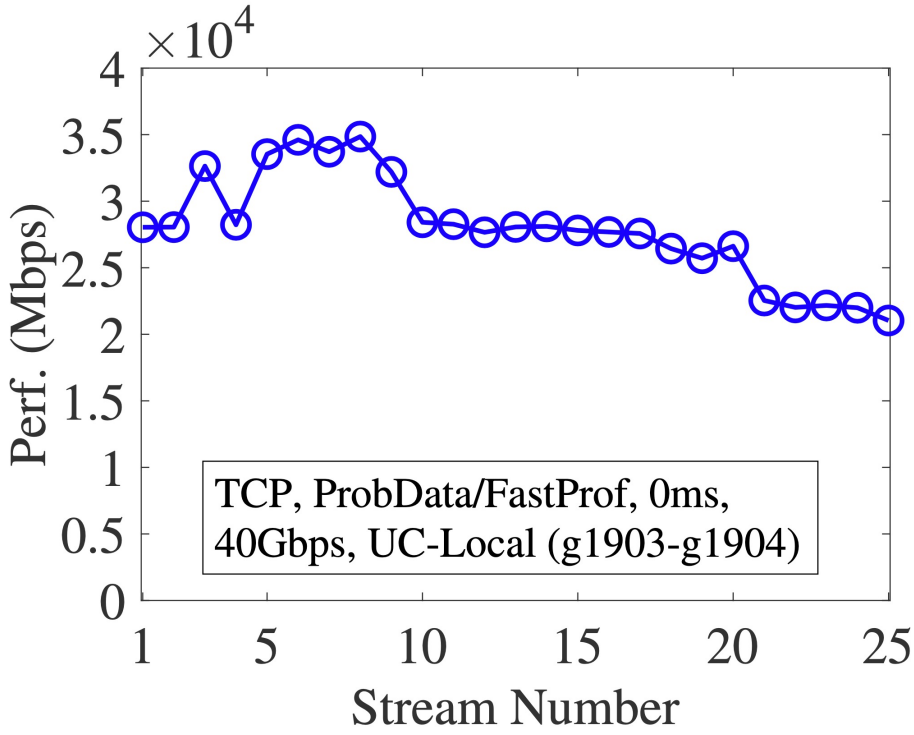
(b) ORNL-E, 0–366 ms, 10 Gbps, TCP

Observations:

- Single-stream TCP achieves near-capacity throughput over connections of short RTTs.

- The throughput suffers over long-haul connections, where using multiple streams helps achieve higher performance.

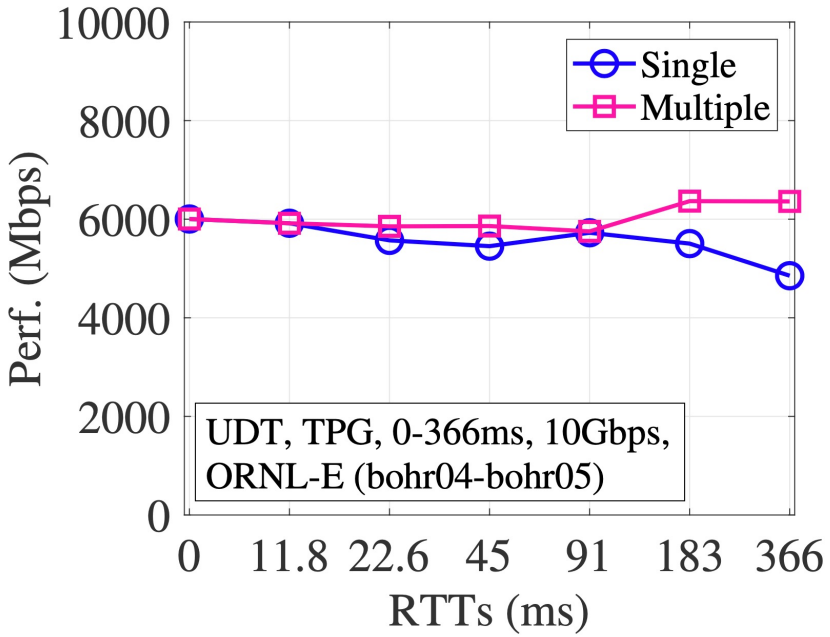
Effects of Stream Count on TCP Performance



Observations:

- Increasing the number of streams may decrease performance (due to extra overhead incurred by multiple streams) after achieving the peak performance.

Effects of RTT on UDT performance

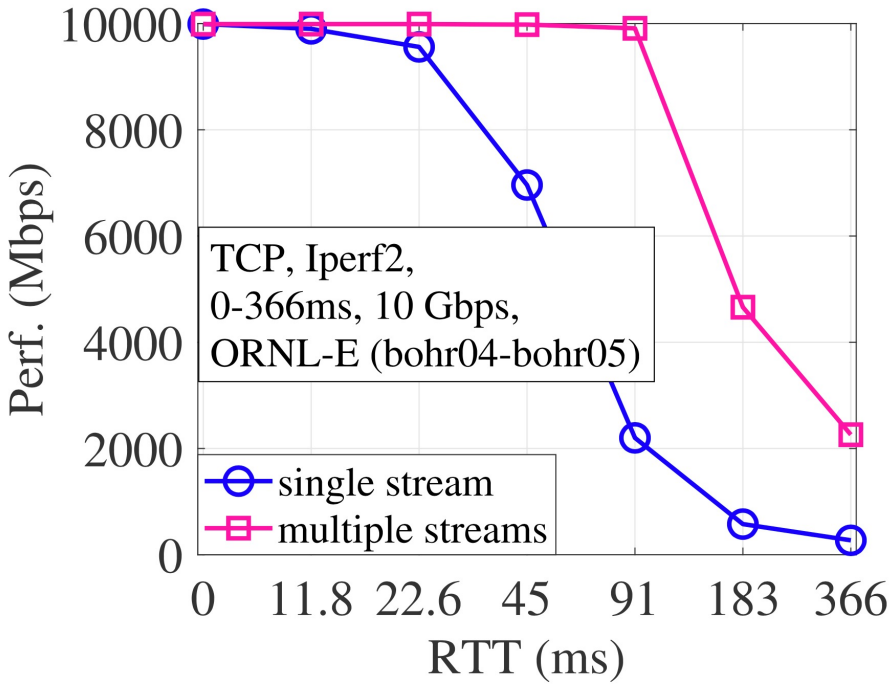


Observations:

- UDT-based data transfer requires certain tuning efforts to achieve good performance since the default settings typically do not achieve satisfactory performance especially over connections with long RTTs (e.g., > 90 ms).

- UDT is not as sensitive to RTTs as TCP due to its Decreasing Additive Increase and Multiplicative Decrease (DAIMD) rate control algorithm.

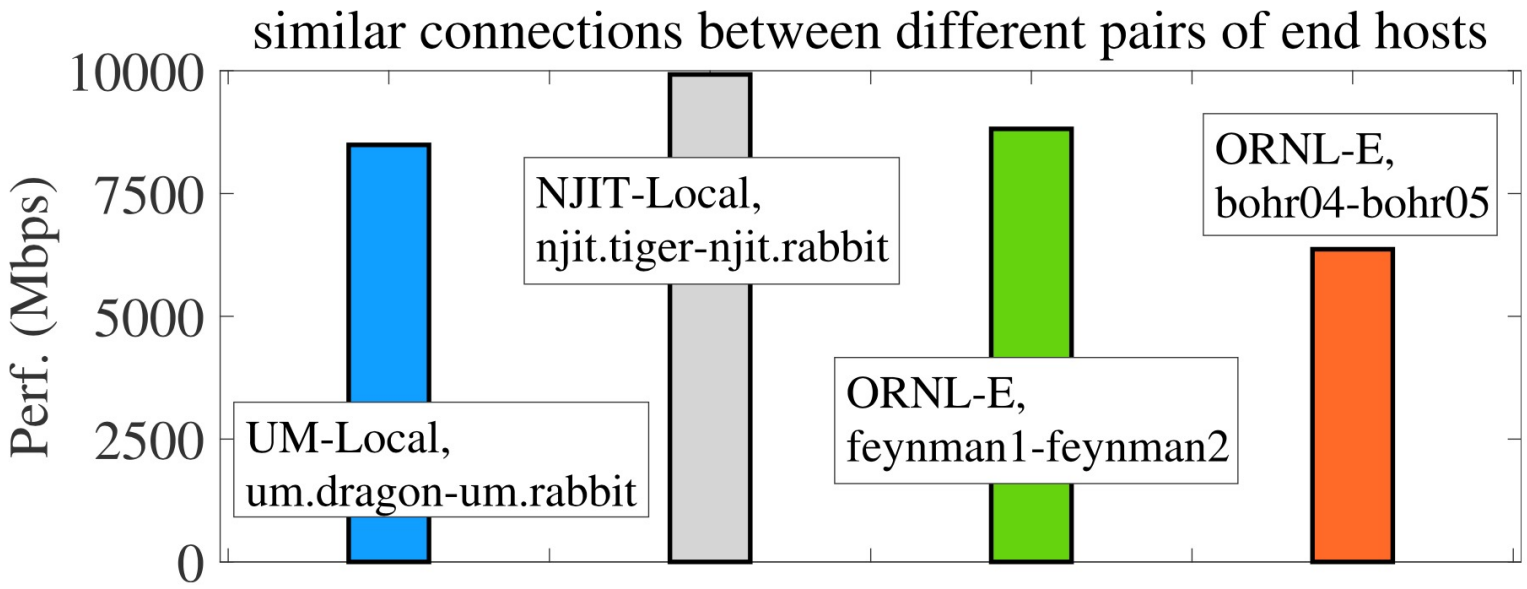
Effects of RTT on TCP performance



Observations:

- The throughput generally decreases (especially with a single stream) as RTT increases.
- UDT is more stable than TCP across different RTTs.
- TCP outperforms UDT for short RTTs but fails to keep up with UDT for mid-range and long RTTs

Effects of End-host on Throughput Performance



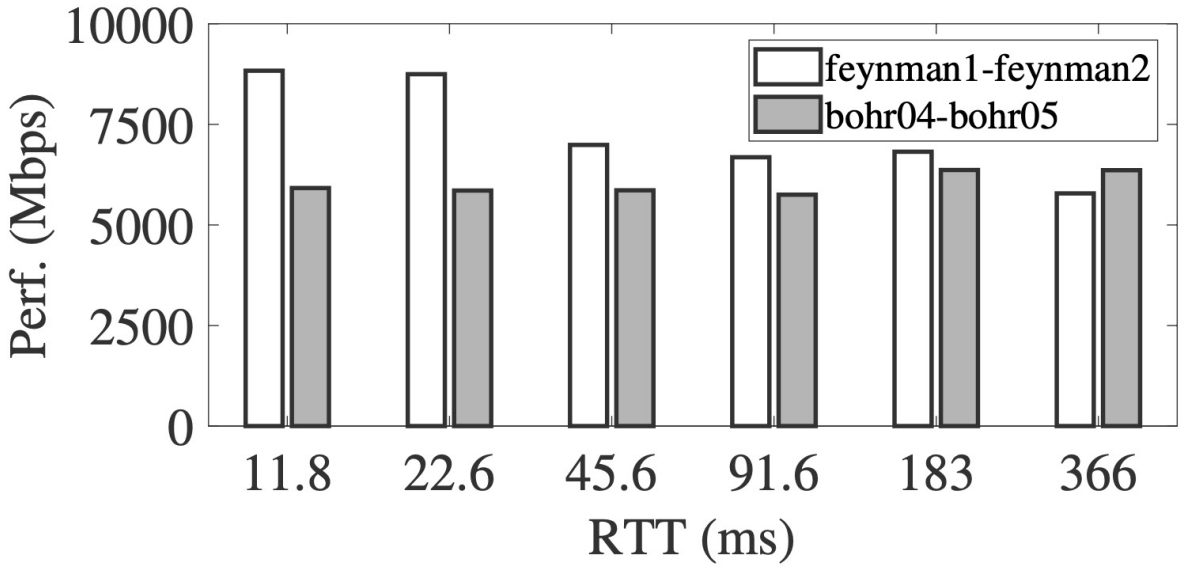
Observations:

- For high-speed data transfer in HPNs, it is important to ensure that both ends can keep up with the speed of incoming/outgoing traffics.

- Together with network properties, it impose an upper bound on the achievable throughput using different transport methods.

Effects of End-host on Throughput Performance

Maximal achievable performance of **UDT** over emulated connections



Observations:

- Similar or identical connections between different end hosts may result in very different maximal performance achievable by “near-exhaustive” performance tuning.

Extensive Transport Profiling

- Extensive data transfer tests
  - Various protocols and toolkits
    - Iperf2/3, FastProf, etc.
    - TCP and its variants, UDP, UDT[2], etc.
  - Different network environments
    - Back-to-back connecting two workstations
    - Physical network between 2 institutions with a total of 5 VMs
    - Local connection between two VMs in the same institution
    - Emulated network in an institution with a total of 4 VMs
    - Long-haul (380ms) WAN from ANL – UChicago

[2] Y. Gu and R. L. Grossman. 2007. UDT: UDP-based data transfer for high-speed wide area networks. Computer Networks.

Dataset Description

1. Each data transfer test typically takes time on the order of minutes to complete.

2. The entire dataset consists of total 109,683 tabular data records, 30,433 of which are performance measurements of TCP tests, and the rest (79,250 records) are performance measurements of UDT tests.

List of attributes in the data transfer performance dataset.

Categories	Attributes	Remarks
Identifier	Record ID	Integer, unique
Testbed	Testbed	String, nominal
End host	CPU frequency	Double, hertz
	# of processors	Integer
	# of cores per processor	Integer
	Memory size	Integer, MB
	Kernel buffer size	Integer, byte
Connection	Bandwidth	Double, Gbps
	RTT	Double, millisecond
	Loss rate	Double, emulated
Toolkits and protocols	Data transfer protocol	String, nominal
	Data transfer toolkit	String, nominal
Control parameters	Frame size	Integer, byte
	Packet size	Integer, byte
	Payload size	Integer, byte
	Block size	Integer, byte
	TCP send buffer size	Double, MB
	TCP receive buffer size	Double, MB
	UDP send buffer size	Double, MB
	UDP receive buffer size	Double, MB
	UDT send buffer size	Double, MB
	UDT receive buffer size	Double, MB
	Number of streams	Integer
	Data size	Double, MB
	Time duration	Integer, second
Performance	Throughput	Double, Mbps

Motivations for Using Machine Learning

- It is extremely hard derive an analytical form to describe the relationship between throughput and control parameters
  - Dynamic host/network environments
  - Numerous hyperparameters to consider
  - Complex behaviors of transfer protocols and methods
- We use machine learning to understand the behaviors of big data transfer and predict maximal achievable performance
  - Critical for the reservation of resources (bandwidths) that are actually needed (to avoid under or overprovisioning) in HPNs.

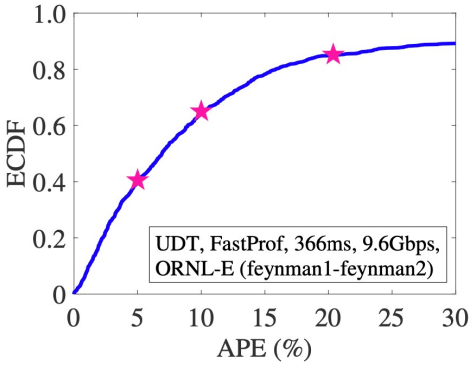
Throughput Performance Prediction

A predictor using a regression model

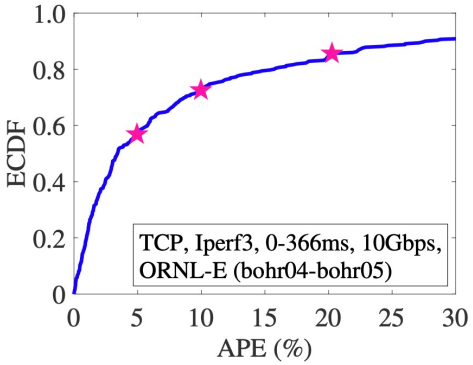
- Support Vector Regression (SVR)
- Fine tuned using the k-fold cross validation approach

The predictor achieves 10% APE roughly among 70% to 80% of all test cases for both TCP and UDT on ORNL-E and NJIT-Local testbed.

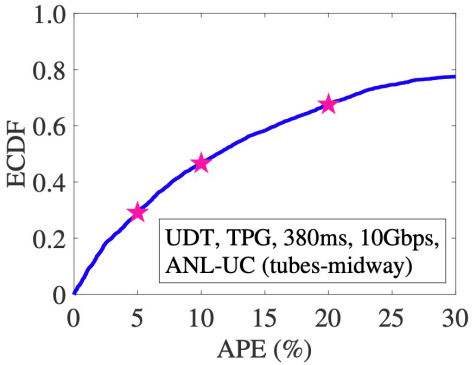
About 10% APE is achieved around 50% of the time for ANL-UC.



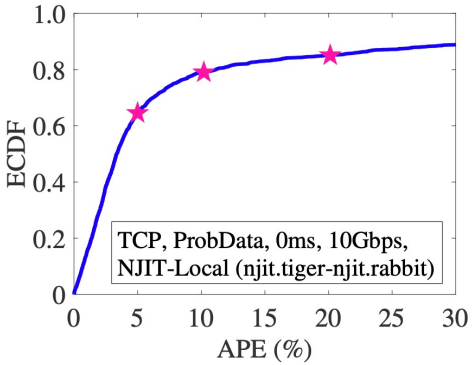
(a) ORNL-E, UDT



(b) ORNL-E, TCP



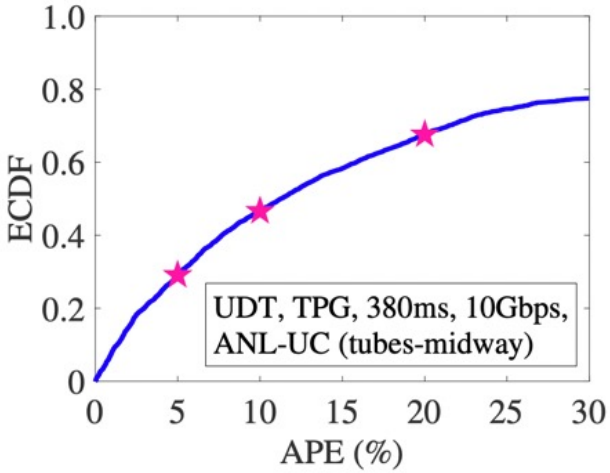
(c) ANL-UC, UDT



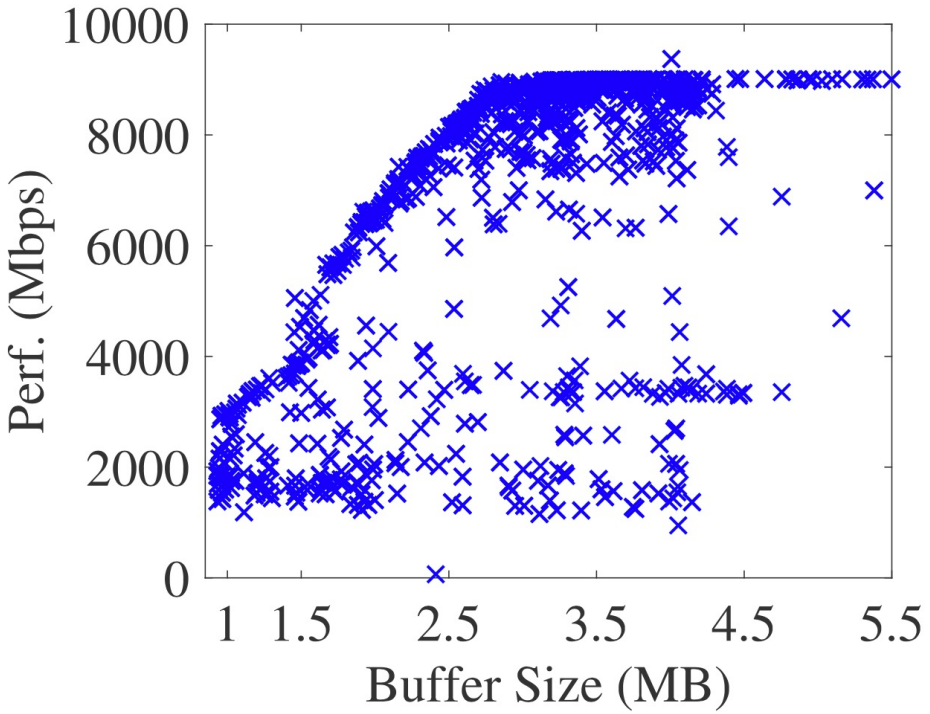
(d) NJIT-Local, TCP

Note: *x*-axis is the absolute percentage error (APE), *y*-axis is the Empirical Cumulative Distribution Function (ECDF).

Latent Effects on Throughput Performance



(c) ANL-UC, UDT

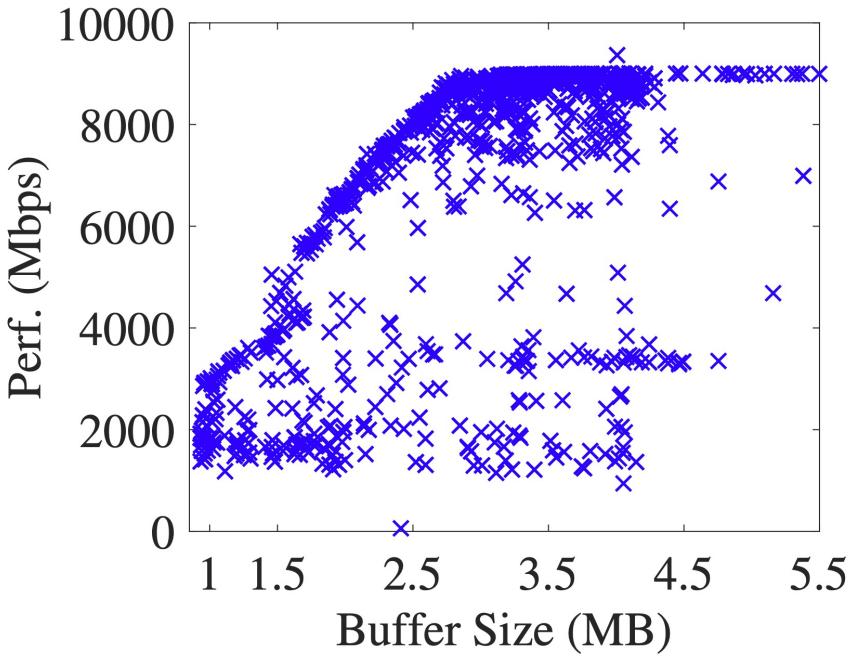


(a) ANL-UC

Latent effects are significant in the TCP measurements of the ***same*** set of data transfer tests conducted on a production HPN (ANL-UC), where the hosts are simultaneously shared by many users.

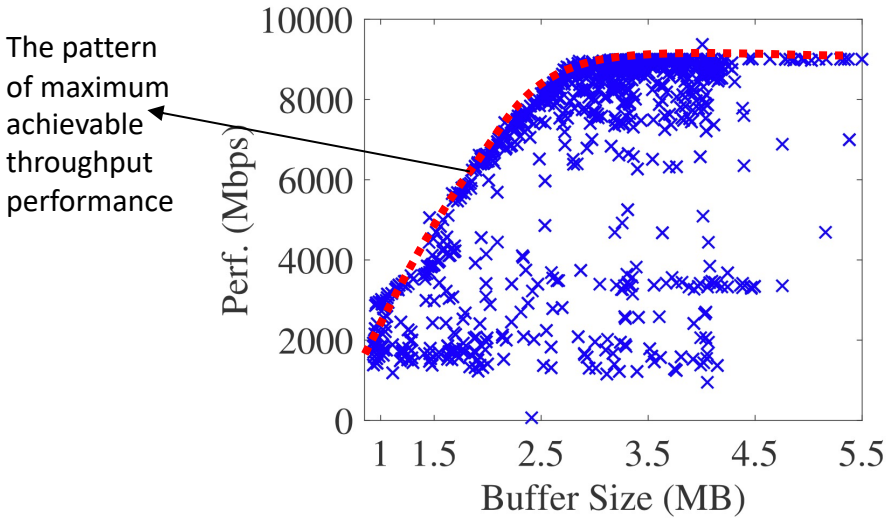
Unexpected Observations

- Latent factors
  - System dynamics due to multi-user resource sharing, unknown competing loads, etc.
- The large number of abnormal data points complicates the underlying pattern, which is hard to learn by machine learning models.
- We conduct an additional "preprocessing" step before learning.

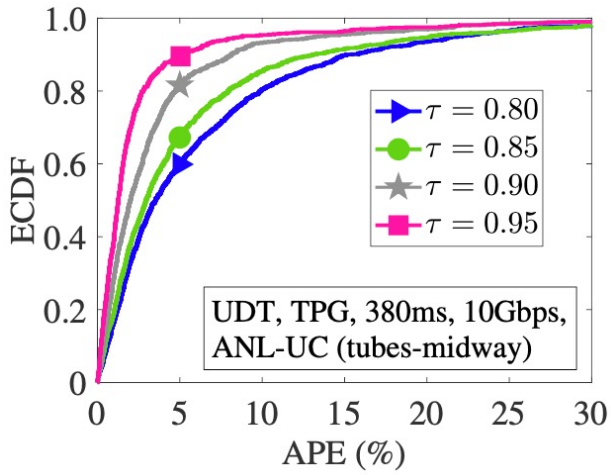


Eliminate Latent Effects

- Eliminate latent affects using a simple threshold-based method
  - If there are multiple measurements for a given vector of control parameters, those with a performance y_i below $\tau \cdot \max\{y_i\}$ are discarded.
- This method may exclude an excessively large number of data points.

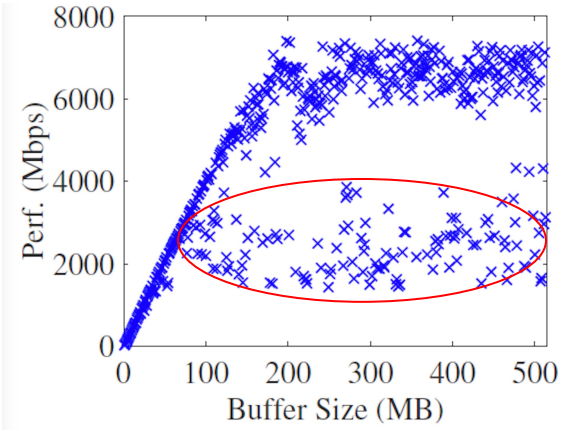


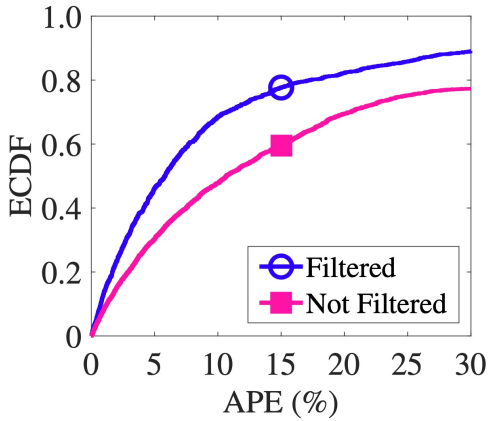
(a) ANL-UC

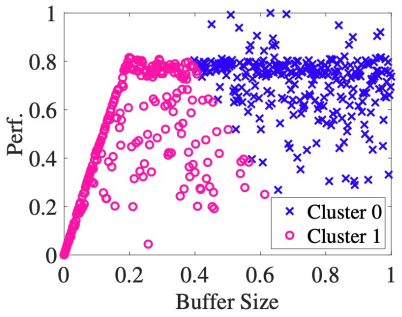


Different Clustering Methods for Abnormalcy Removal

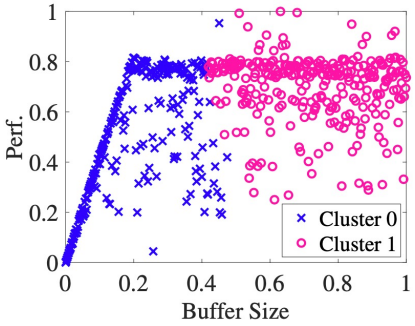
"Abnormalcy" removal:



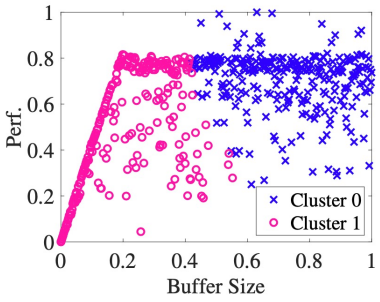




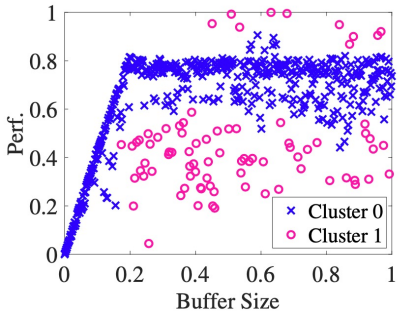
K-means



GMM



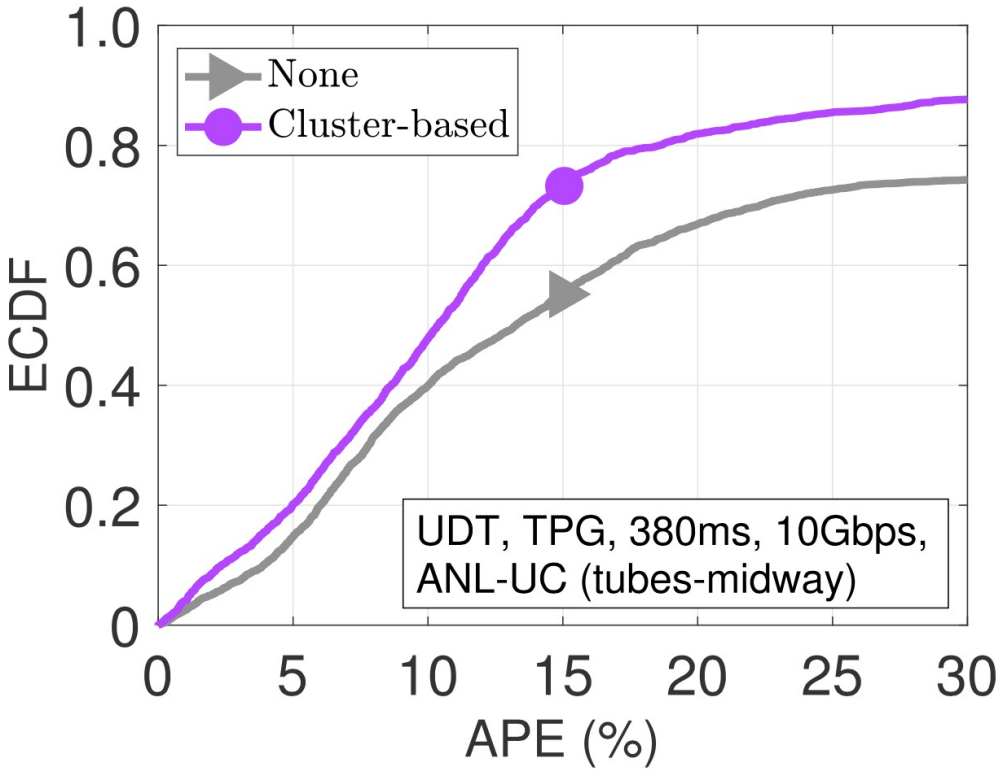
Spectral clustering



DBSCAN

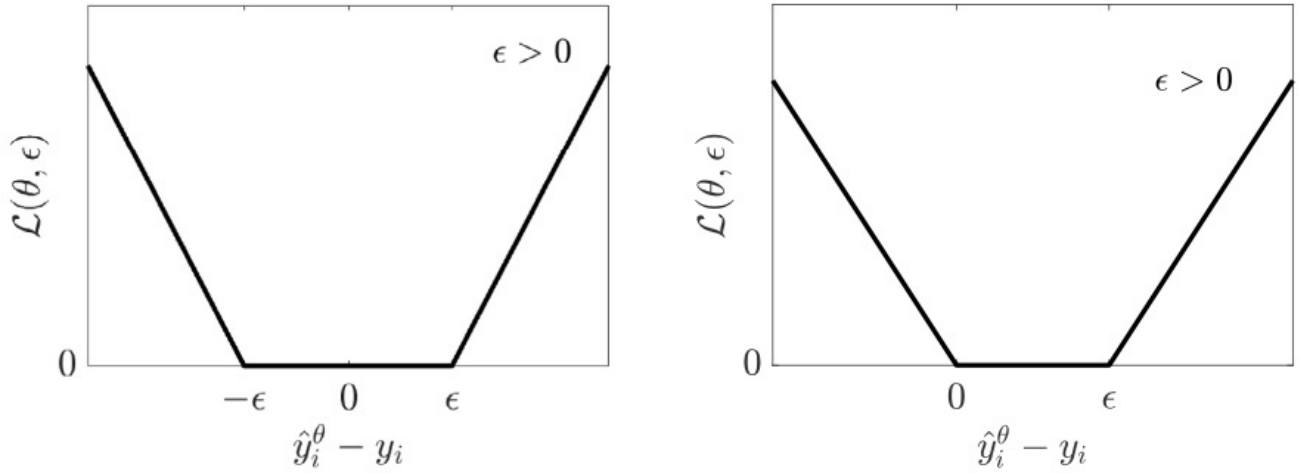
Density based spatial clustering of applications with noises (DBSCAN)

Performance Improvement by Eliminating Latent Effects



Prediction accuracy is significantly improved, and the 15% percentile of APE is increased to 77% (a 20%+ improvement).

SVR-based Performance Predictor With a Customized Loss Function



(a) ϵ-insensitive loss (b) one-side ϵ-insensitive loss

A customized loss function motivated by the domain knowledge of HPN management:

- the reserved bandwidth over a dedicated connection should meet the bandwidth requirement of a data transfer request with minimal waste.

$$\mathcal{L}(\theta, \epsilon) = \begin{cases} -(\hat{y}_i^\theta - y_i), & \text{if } \hat{y}_i^\theta - y_i < 0 \\ 0, & \text{if } 0 \le \hat{y}_i^\theta - y_i \le \epsilon \\ \hat{y}_i^\theta - y_i, & \text{if } \hat{y}_i^\theta - y_i > \epsilon \end{cases}$$

More Performance Metrics for Evaluation

Evaluation Metrics:

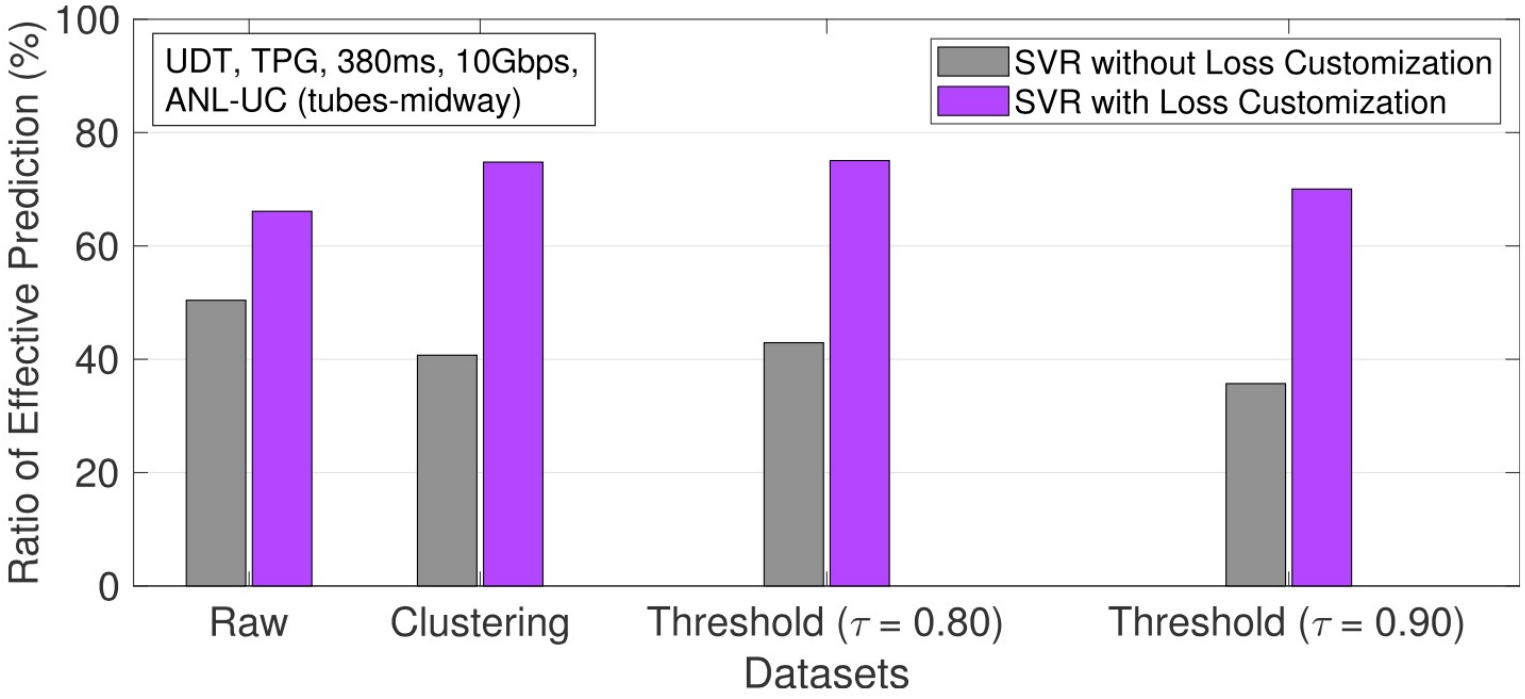
- Root Mean Square Error (RMSE)
- Mean Absolute Error (MAE)
- Custom Mean Absolute Percentage Error (CMAPE)

$$\frac{1}{n}\sum_{i=1}^{n}\{\max\left(y_i - \hat{f}_\theta(\boldsymbol{x}_i), 0\right) + \max(\hat{f}_\theta(\boldsymbol{x}_i) - \varepsilon \cdot y_i, 0)\}$$

- Effective Prediction Percentage (EPP):

$$\beta = \frac{1}{n}\sum_{i=1}^{n}\mathcal{I}\{y_i \leq \hat{f}_\theta(\mathbf{x}_i) \leq \epsilon \cdot y_i\}$$

Experimental Results



Comparison of the ratio of effective prediction using SVR with and without loss function customization: customized loss always performs better.

Experimental Results

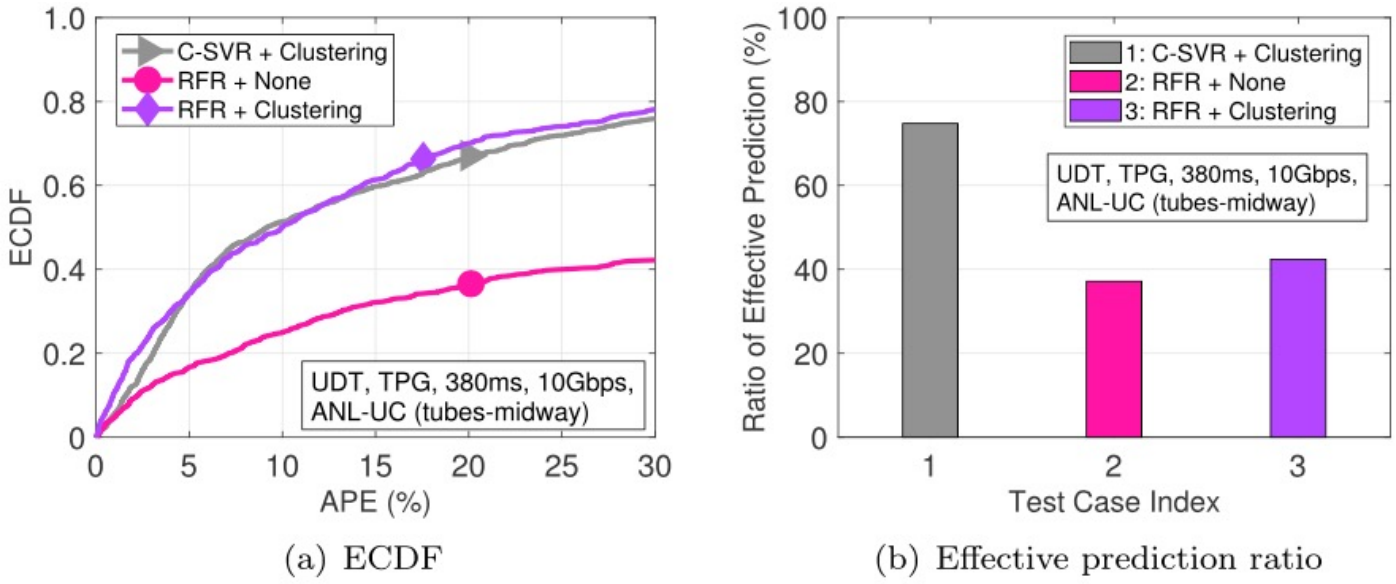


Fig. 19. Comparison of the prediction results using SVR and RFR.

SVR and RFR perform roughly equally well: 10% of APE is achieved for 70% of the cases among all tests considered.

Theoretical Analysis

- Throughput:
  - The expected throughput performance y_i of a big data transfer during time interval [0, ΔT] is given by

$$\overline{y}_i = \frac{\int_0^{\Delta T} y_i(\mathbf{x}_i, \mathbf{u}_i, t)\, dt}{\Delta T},$$

where $y_i(\boldsymbol{x}_i, \boldsymbol{u}_i, t)$ is the throughput at time point t in response to a specific feature vector $\boldsymbol{x}_i$ and latent variables $\boldsymbol{u}_i$.

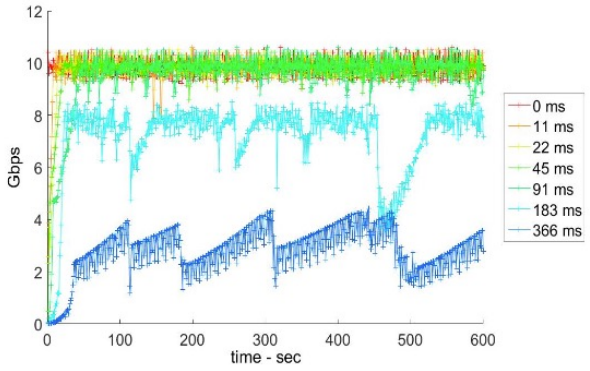
Throughput Estimates

The throughput $y(\boldsymbol{x})$ is a random quantity with a complex distribution $P_{y(\boldsymbol{x})}$ as it depends on many factors including:

i) End host system configurations and dynamics.

ii) Network connection properties and randomness.

iii) Data transfer applications and their underlying protocols (control parameter values, congestion control mechanisms, etc.).



Throughput profile and time traces of Scalable-TCP [1]

1. N. Rao, Q. Liu. S. Sen, D. Towlsey, G. Vardoyan, R. Kettimuthu and I. Foster. (2017). TCP Throughput Profiles Using Measurements Over Dedicated Connections. The International ACM Symposium on High-Performance Parallel and Distributed Computing.

Confidence Estimates

The statistical significance of estimated throughput:

Expected throughput:

$$\overline{y}(\mathbf{x}) = E\left[y(\mathbf{x})\right] = \int y(\mathbf{x})\, d\,\mathbf{P}_{y(\mathbf{x})},$$

where $\boldsymbol{P}_{y(\boldsymbol{x})}$ is a complex distribution depending on many factors.

Estimated from profile:

$$\hat{y}(\mathbf{x}_k) = \frac{1}{n_k} \sum_{i=1}^{n_k} y(\mathbf{x}_k, t_i^k),$$

We show that $\hat{y}(\mathbf{x}_k)$ is indeed a good estimate of $\overline{y}(\mathbf{x}_k)$.

Confidence Estimates

Consider an estimate g(·) of $\bar{y}(\boldsymbol{x})$ based on performance measurements from a class $\mathcal{F}$ of unimodal functions bounded in [0,B], i.e., 0 ≤ g ≤ B, g ∈ $\mathcal{F}$.

The expected quadratic loss: $$I(g) = \int \left[g(\mathbf{x}) - y(\mathbf{x}, t)\right]^2 d\mathbf{P}_{y(\mathbf{x},t)}$$

The best estimator: $$I(g^*) = \min_{g \in \mathcal{F}} I(g)$$

Confidence:

$$P\left\{I(\hat{y}) - I(g^*) > \lambda\right\} \leq 36 \left(\frac{16K^2L^2n}{\lambda^2}\right)^{\left(1+\frac{8BKL}{\lambda}\right)\log_2\left(\frac{en}{B}\right)} \cdot n \cdot \exp\left(-\frac{n\lambda^2}{16K^2}\right)$$

where L is a positive Lipschitz constant and *n* is the number of records.

Decay faster

Conclusion

- Many applications require the transfer of big data through bandwidth reservation in high-performance networks.
- Bandwidth reservation requires performance modeling and prediction.
- The performance of big data transfer is dependent on many factors at the host, network, and application levels.
- Machine learning seems to be an effective approach to model and predict the performance of big data transfer.

Thanks!

Questions?