Adaptive Transfer Adjustment in Efficient Bulk Data Transfer Management for Climate Datasets

Alex Sim, Mehmet Balman, Dean Williams,
Arie Shoshani, Vijaya Natarajan

Lawrence Berkeley National Laboratory
Lawrence Livermore National Laboratory

The 22nd IASTED International Conference on Parallel and Distributed Computing and Systems
PDCS2010 - Nov 9th, 2010

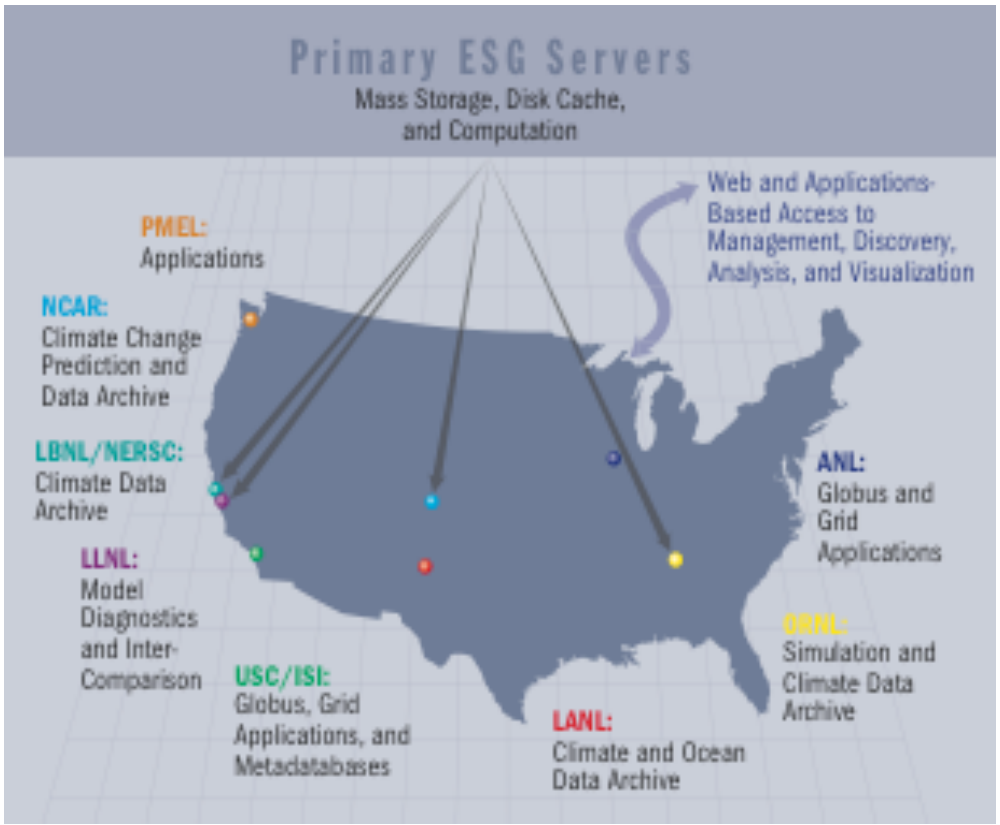
ESG (Earth System Grid)

Supports the infrastructure for climate research

Provide technology to access, distributed, transport, catalog climate simulation data

Production since 2004
- ESG-I (1999-2001)
- ESG-II (2002-2006)
- ESG-CET(2006-present)

ANL, LANL, LBNL, LLNL, NCAR, ORNL, NERSC, …



ESG (Earth System Grid) / Climate Simulation Data

NCAR CCSM ESG portal

237 TB of data at four locations: (NCAR, LBNL, ORNL, LANL)

965,551 files

Includes the past 7 years of joint DOE/NSF climate modeling experiments

LLNL CMIP-3 (IPCC AR4) ESG portal

35 TB of data at one location

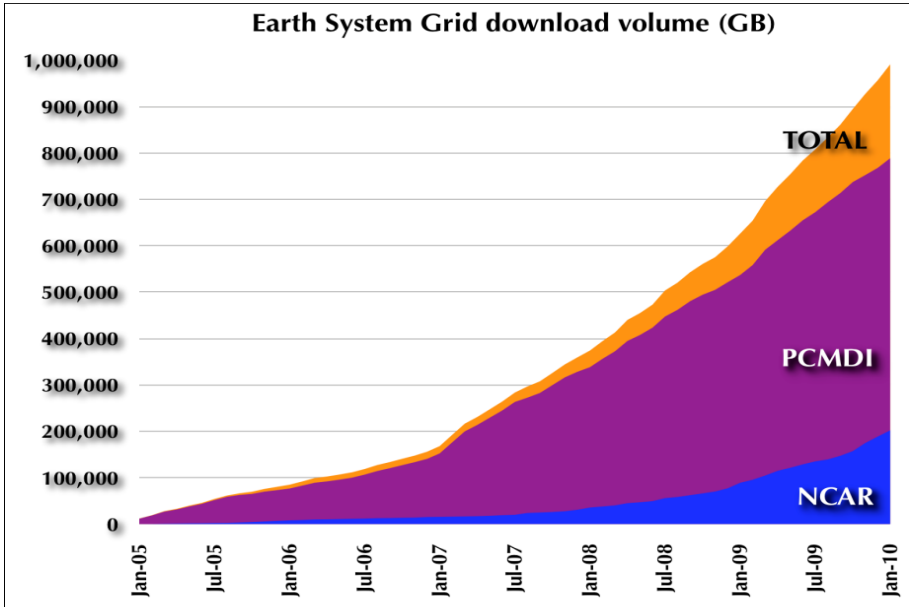
83,337 files

model data from 13 countries

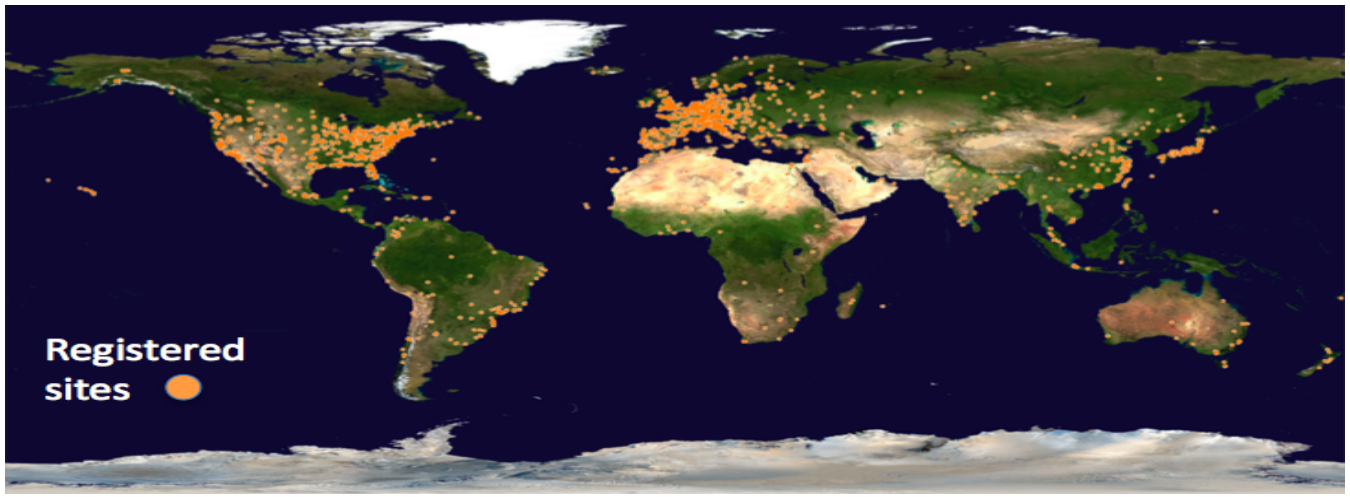
Generated by a modeling campaign coordinated by the Intergovernmental Panel on Climate Change (IPCC)

Over 565 scientific peer-review publications

ESG (Earth System Grid) / Climate Simulation Data



ESG web portals distributed worldwide
- Over 2,700 sites
- 120 countries
- 25,000 users
- Over 1 PB downloaded



Climate Data : ever-increasing sizes

Early **1990**'s (e.g., AMIP1, PMIP, CMIP1)
modest collection of monthly mean 2D files: ~1 GB
Late **1990**'s (e.g., AMIP2)
large collection of monthly mean and 6-hourly 2D and 3D fields: ~500 GB
2000's (e.g., IPCC/CMIP3)
fairly comprehensive output from both ocean and atmospheric components; monthly, daily, and 3 hourly: ~35 TB
2011:
The IPCC 5th Assessment Report (AR5) in 2011: expected 5 to 15 PB

The Climate Science Computational End Station (CCES) project at ORNL: expected around 3 PB

The North American Regional Climate Change Assessment Program (NARCCAP): expected around 1 PB

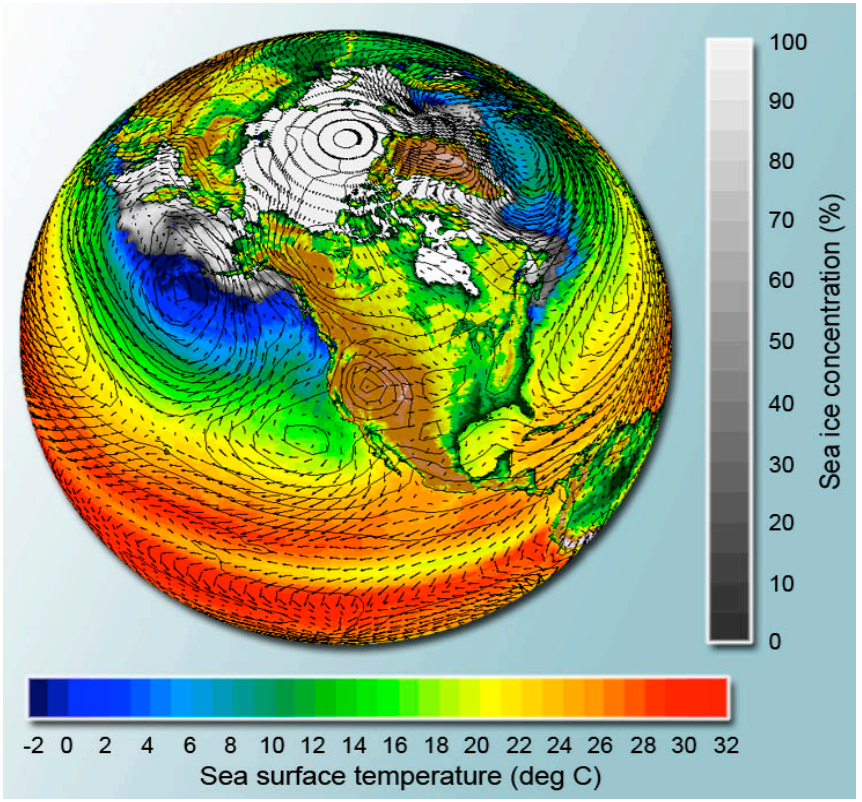
The Cloud Feedback Model Intercomparison Project (CFMIP) archives: expected to be .3 PB

ESG (Earth System Grid) / Climate Simulation Data

Massive data collections:

- **shared by thousands of researchers**
- **distributed among many data nodes around the world**

Replication of published core collection (for scalability and availability)

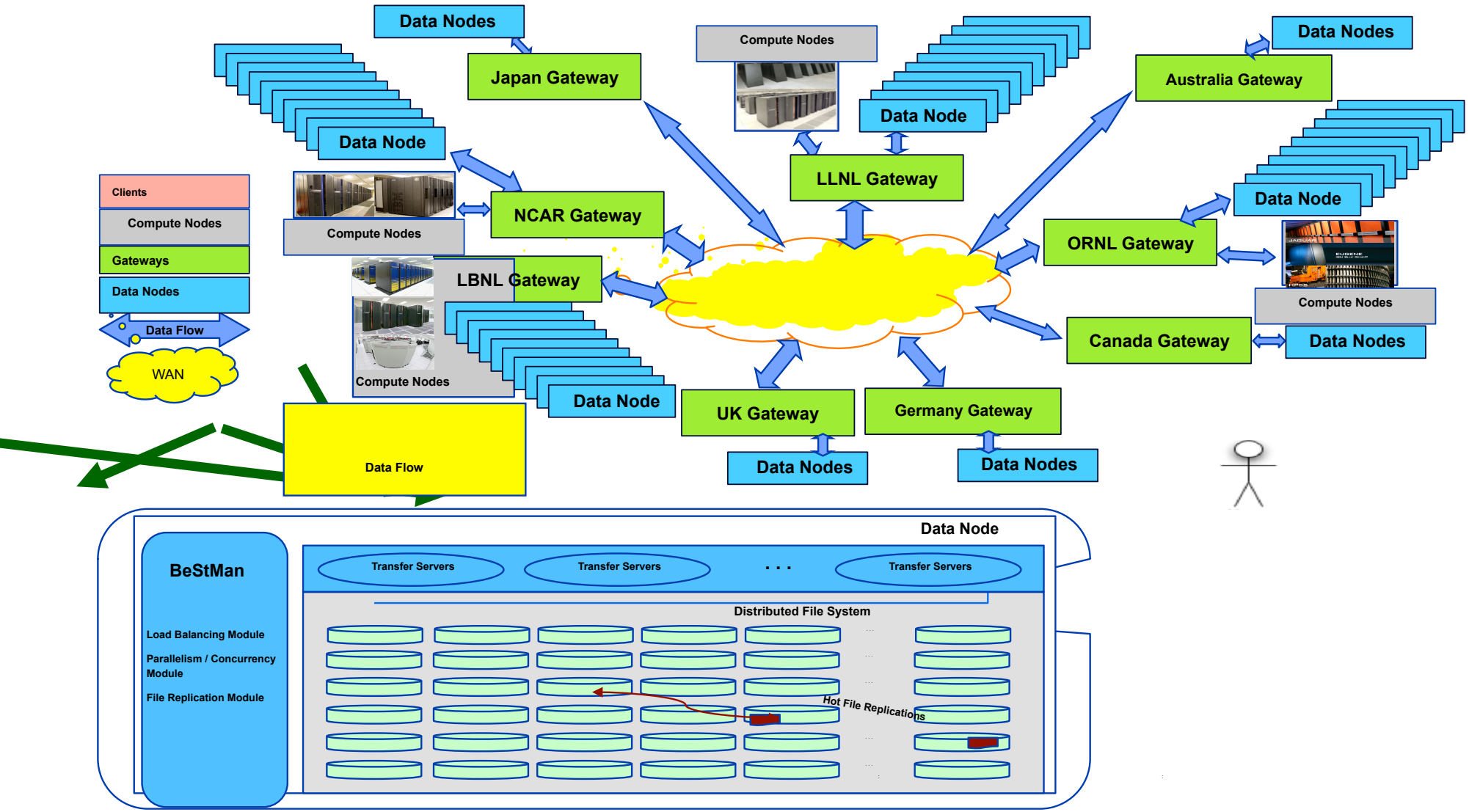


Results from the Parallel Climate Model (PCM) depicting wind vectors, surface pressure, sea surface temperature, and sea ice concentration. Prepared from data published in the ESG using the FERRET analysis tool by Gary Strand, NCAR.

Bulk Data Movement in ESG

- Move terabytes to petabytes (many thousands of files)
  - Extreme variance in file sizes
- Reliability and Robustness
- Asynchronous long-lasting operation
  - Recovery from transient failures and automatic restart
  - Support for checksum verification
- On-demand transfer request status
- Estimation of request completion time

Replication Use-case



Faster Data Transfers

End-to-end bulk data transfer (latency wall)

- TCP based solutions
  - Fast TCP, Scalable TCP etc
- UDP based solutions
  - RBUDP, UDT etc
- Most of these solutions require kernel level changes
- Not preferred by most domain scientists

Application Level Tuning

- Take an application-level transfer protocol (i.e. GridFTP) and tune-up for better performance:
  - Using Multiple (Parallel) streams
  - Tuning Buffer size

  (efficient utilization of available network capacity)

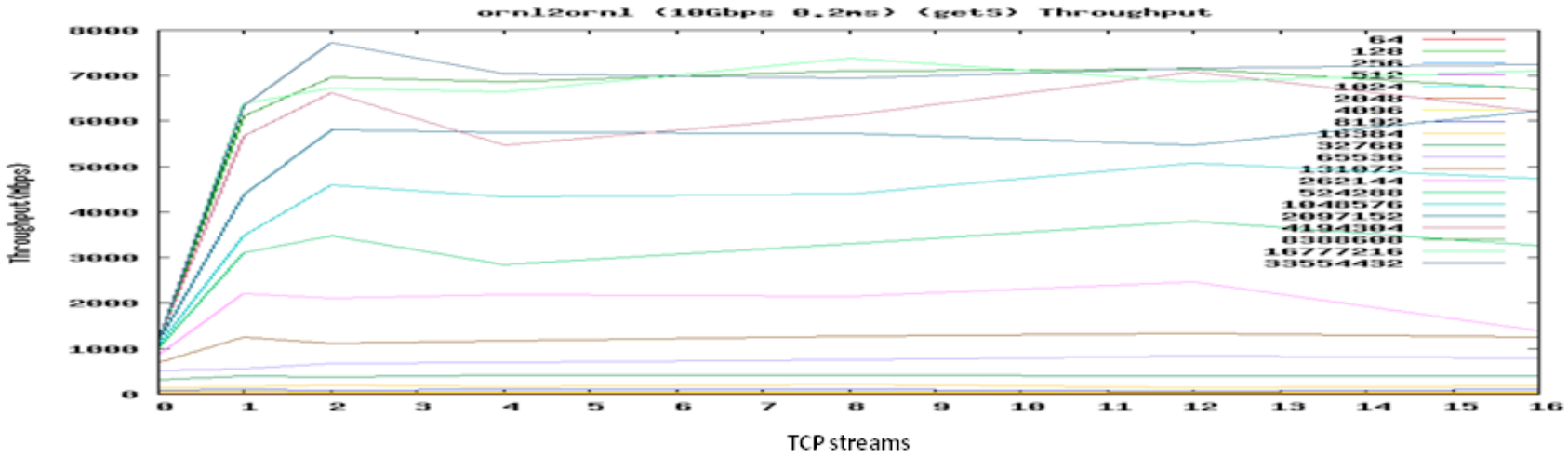
Level of Parallelism in End-to-end Data Transfer

- number of parallel data streams connected to a data transfer service for increasing the utilization of network bandwidth
- number of concurrent data transfer operations that are initiated at the same time for better utilization of system resources.

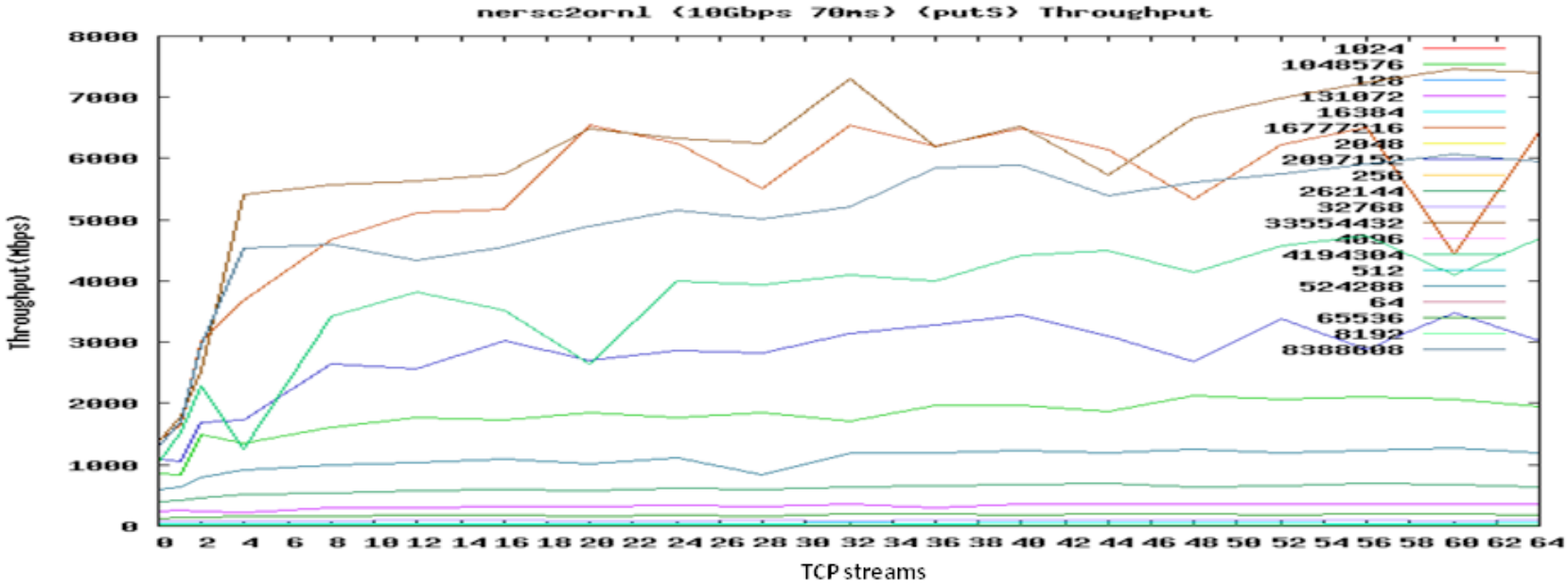
Parallel TCP Streams

- Instead of a single connection at a time, multiple TCP streams are opened to a single data transfer service in the destination host.

- We gain larger bandwidth in TCP especially in a network with less packet loss rate; parallel connections better utilize the TCP buffer available to the data transfer, such that N connections might be N times faster than a single connection
- Multiple TCP streams puts extra system overhead



(a)



Parallel TCP Streams

(b)

Bulk Data Mover (BDM)

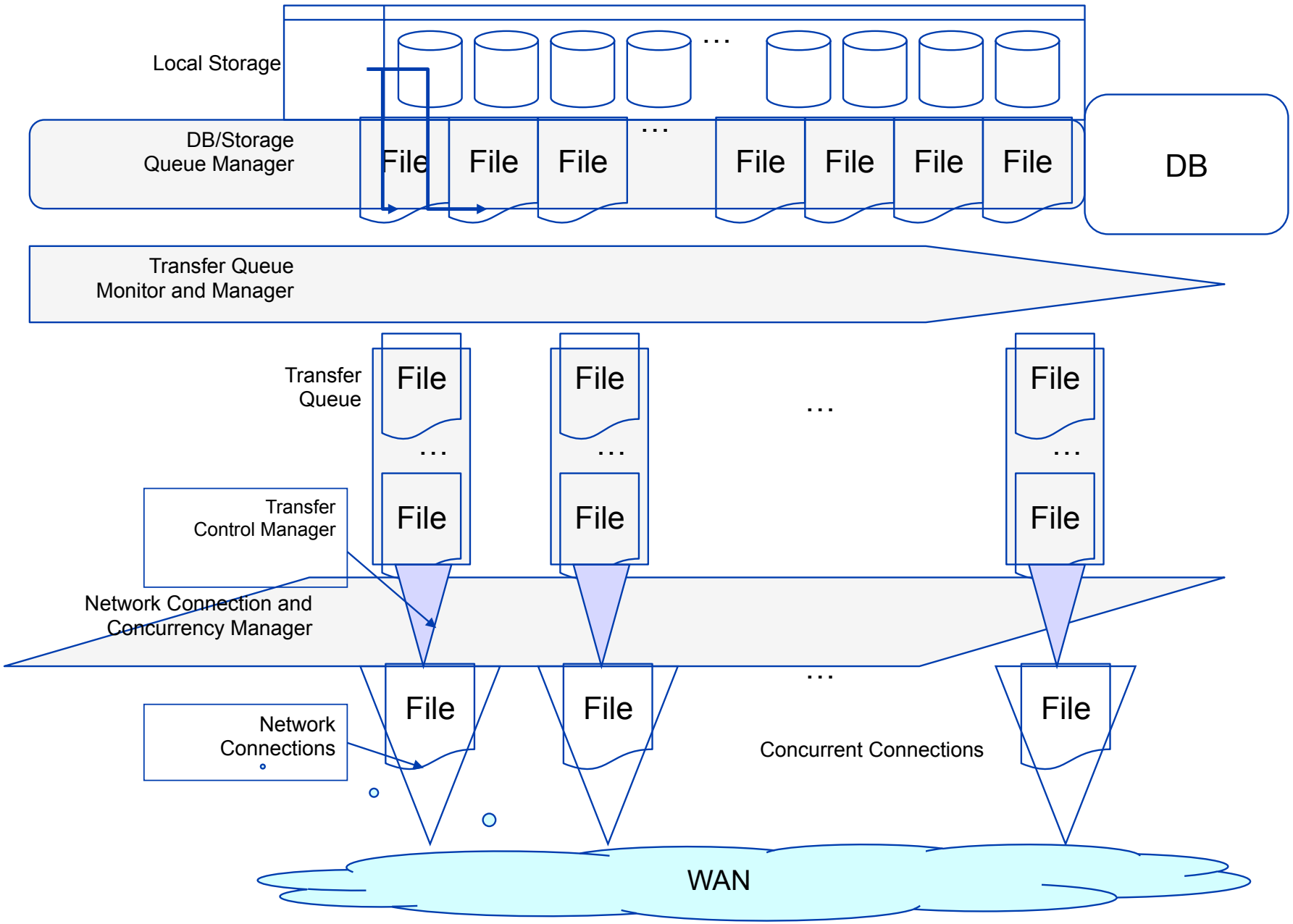
- Monitoring and statistics collection
- Support for multiple protocols (GridFTP, HTTP)

- Load balancing between multiple servers
- Data channel connection caching, pipelining

- Parallel TCP stream, concurrent data transfer operations

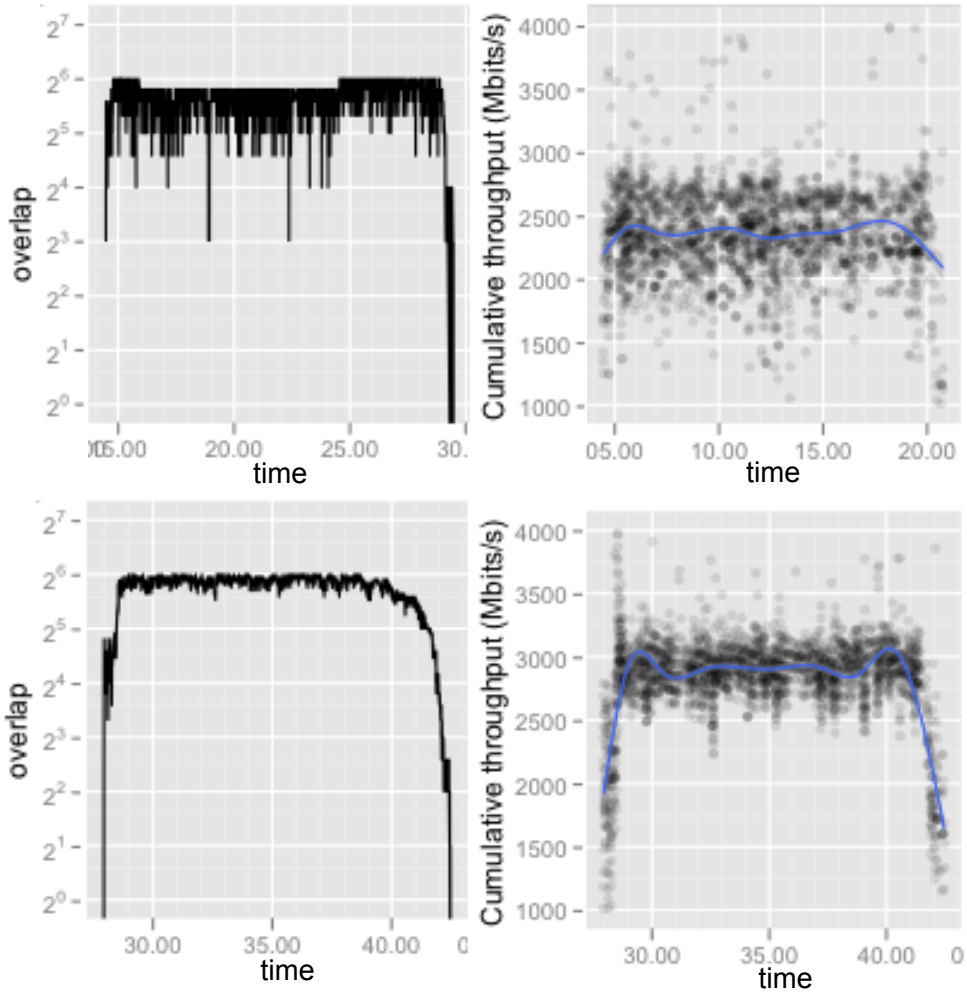
- Multi-threaded transfer queue management

- Adaptability to the available bandwidth



Bulk Data Mover (BDM)

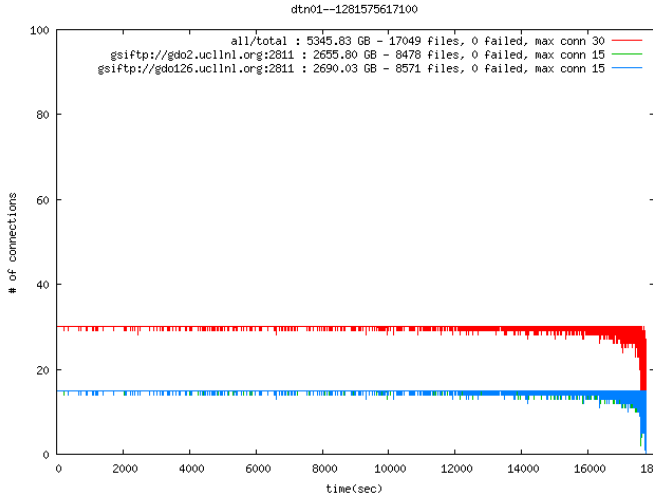
Transfer Queue Management

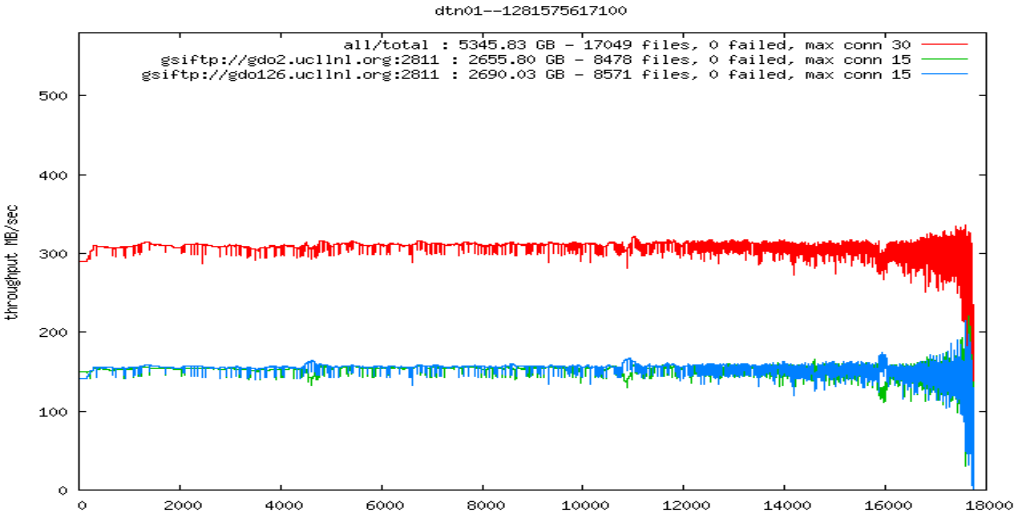


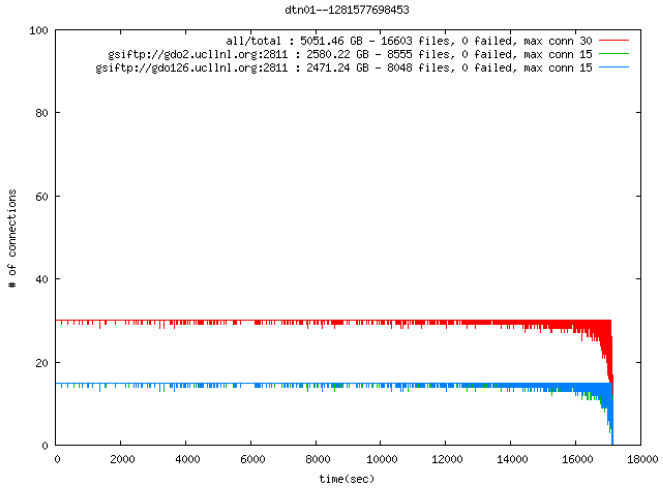
The number of concurrent transfers on the left column shows consistent over time in well-managed transfers shown at the bottom row, compared to the ill or non-managed data connections shown at the top row. It leads to the higher overall throughput performance on the lower-right column.

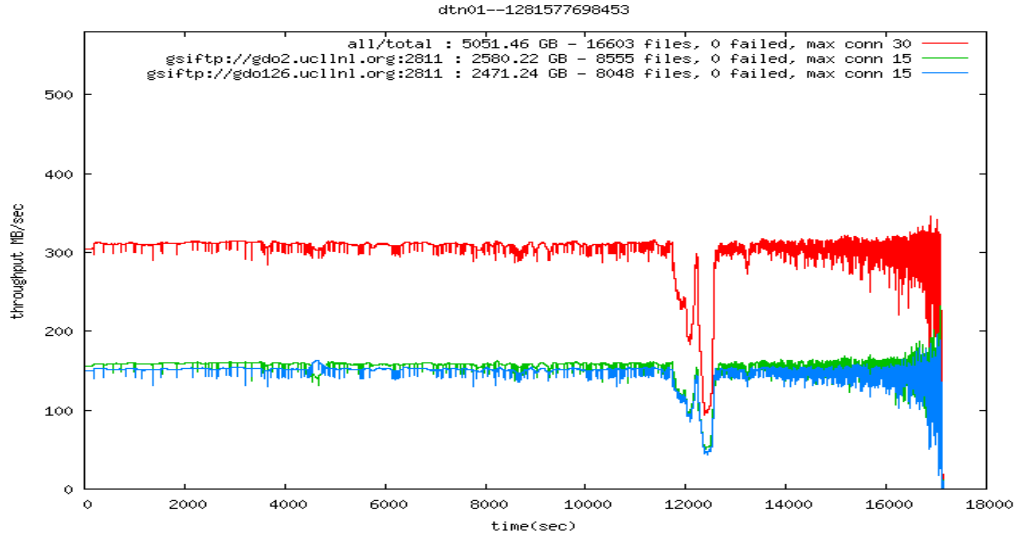
* Plots generated from NetLogger

Adaptive Transfer Management







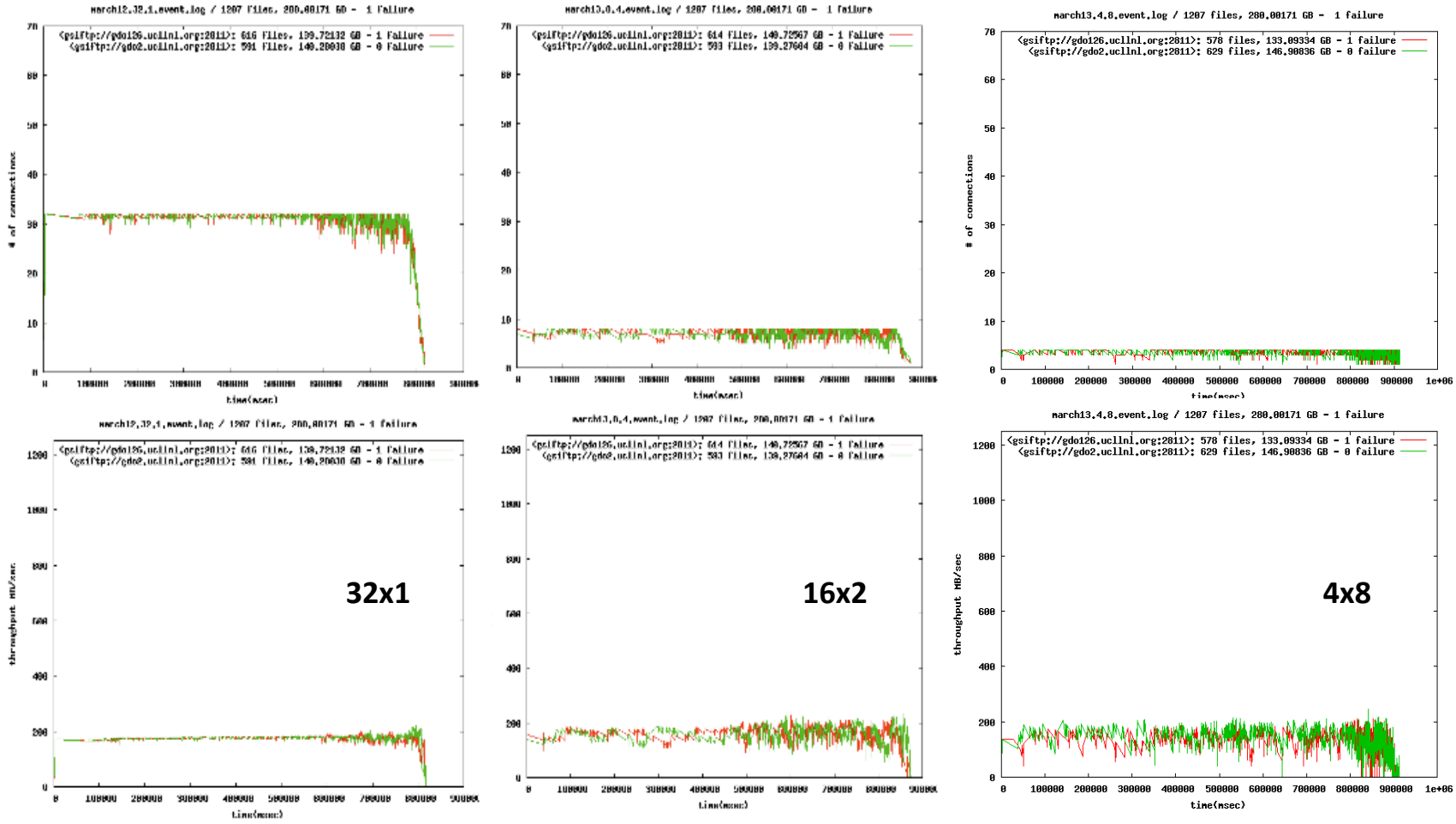


Bulk Data Mover (BDM)

- number parallel TCP streams?
- number of concurrent data transfer operations?

- Adaptability to the available bandwidth

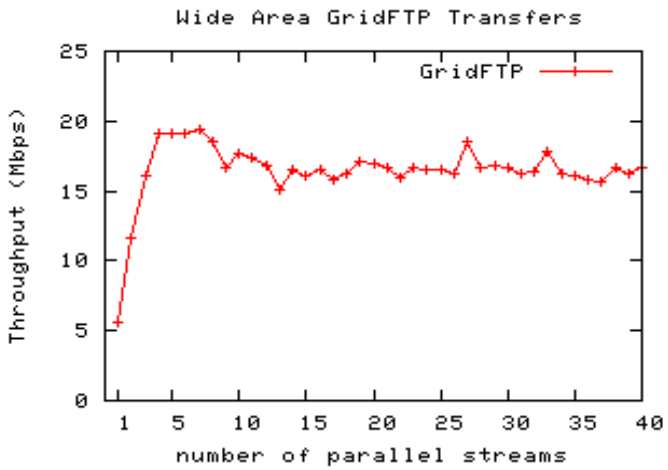
Parallel Stream vs Concurrent Transfer



- Same number of total streams, but different number of concurrent connections

Parameter Estimation

- Can we predict this behavior?



- Yes, we can come up with a good estimation for the parallelism level
  - Network statistics
  - Extra measurement
  - Historical data

Parallel Stream Optimization

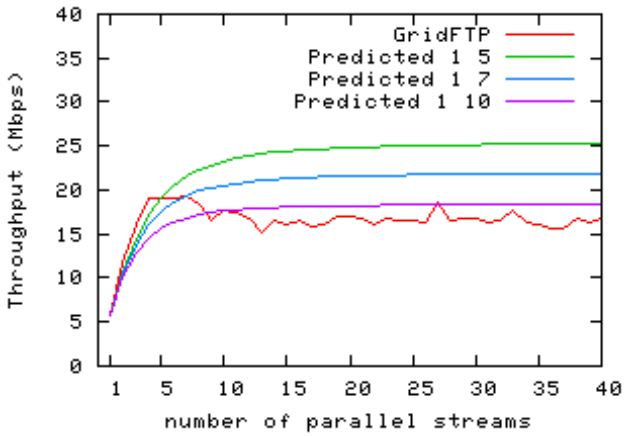
single stream, theoretical calculation of throughput base on MSS, RTT and packet loss rate:

$$Th <= \frac{MSS}{RTT}\frac{c}{\sqrt{p}}$$

n streams gains as much as total throughput of n single stream: (not correct)

$$Th_n <= \frac{MSS \times c}{RTT}\left(\frac{n}{\sqrt{p}}\right)$$

A better model: a relation is established between *RTT* and the number of streams *n:*



$$p'_n = p_n \frac{RTT_n^2}{c^2 MSS^2} = a'n^2 + b'$$

$$Th_n = \frac{n}{\sqrt{p'_n}} = \frac{n}{\sqrt{a'n^2 + b'}}$$

Parameter Estimation

- Might not reflect the best possible current settings (Dynamic Environment)
  - What if network condition changes?
- Requires three sample transfers (curve fitting)
  - need to probe the system and make measurements with external profilers
- Does require a complex model for parameter optimization

Adaptive Tuning

- Instead of predictive sampling, use data from actual transfer
- transfer data by chunks (partial transfers) and also set control parameters on the fly.
- measure throughput for every transferred data chunk
- gradually increase the number of parallel streams till it comes to an equilibrium point

Adaptive Tuning

- No need to probe the system and make measurements with external profilers
- Does not require any complex model for parameter optimization
- Adapts to changing environment

But, overhead in changing parallelism level

Fast start (exponentially increase the number of parallel streams)

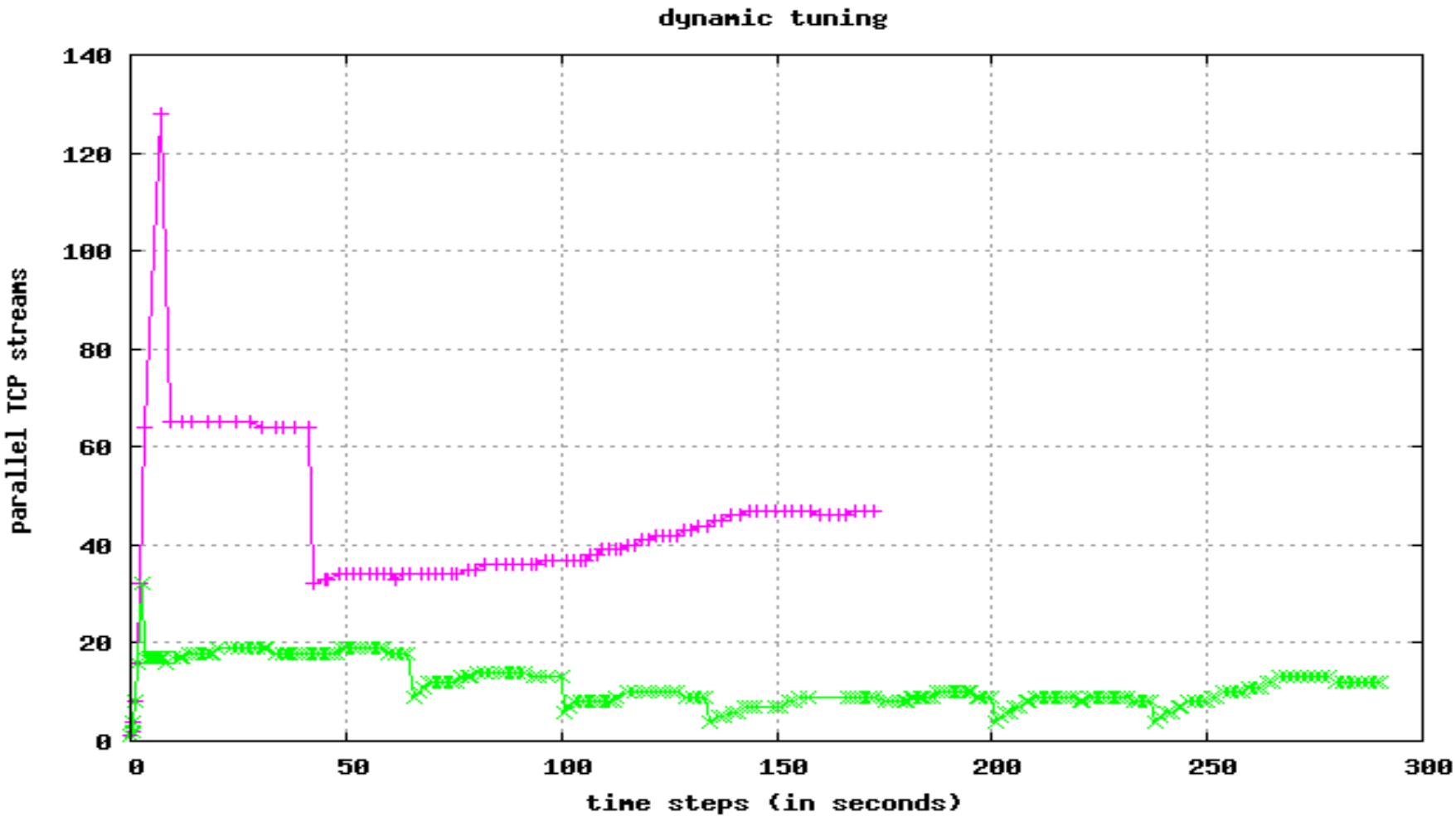
Adaptive Tuning

- Start with single stream (n=1)
- Measure instant throughput for every data chunk transferred

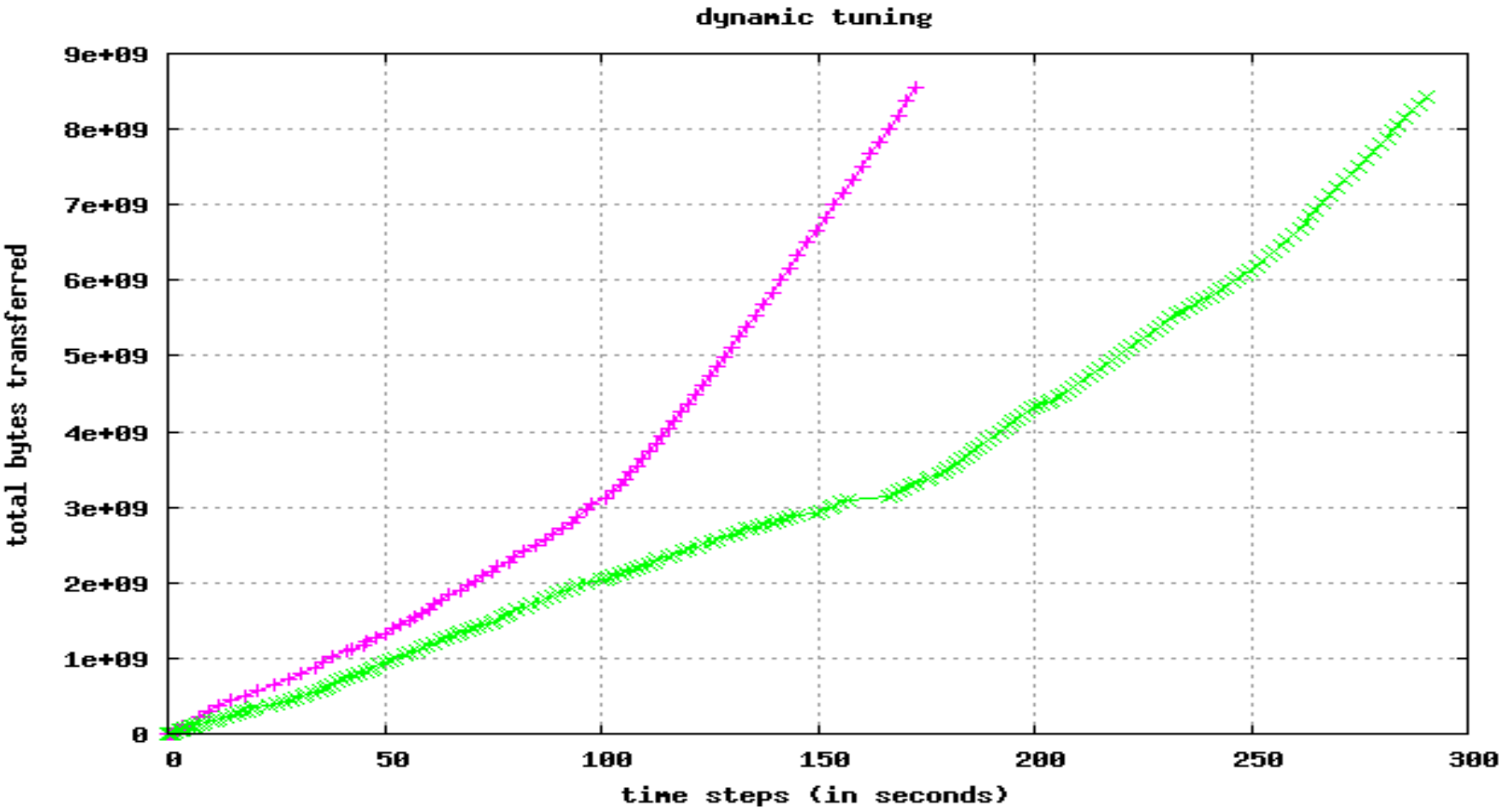
  (fast start)
  - Increase the number of parallel streams (n=n*2),
  - transfer the data chunk
  - measure instant throughput
- If current throughput value is better than previous one, continue
- Otherwise, set n to the old value and gradually increase parallelism level (n=n+1)

- If no throughput gain by increasing number of streams (found the equilibrium point)
  - Increase chunk size (delay measurement period)

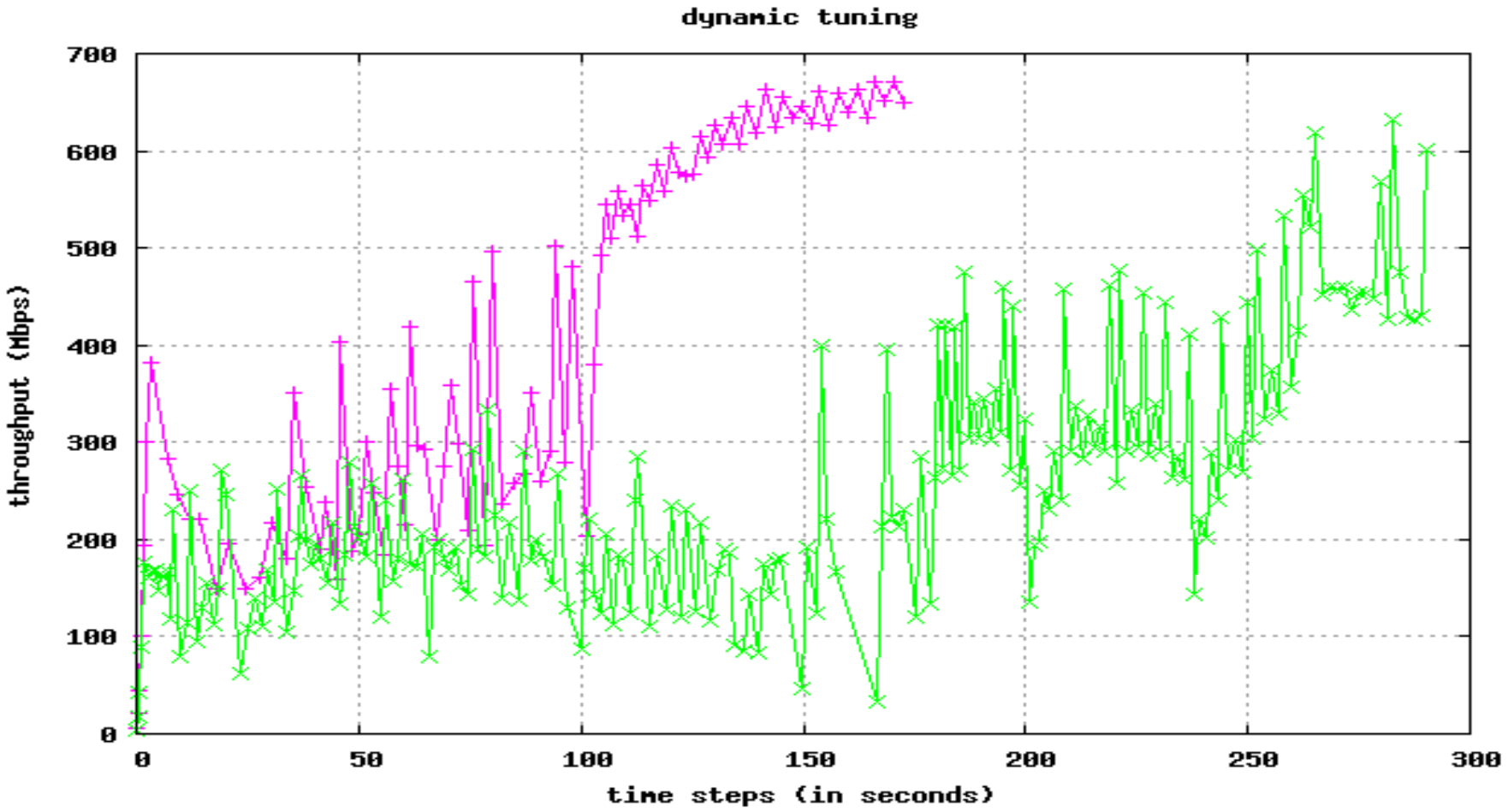
Dynamic Tuning Algorithm



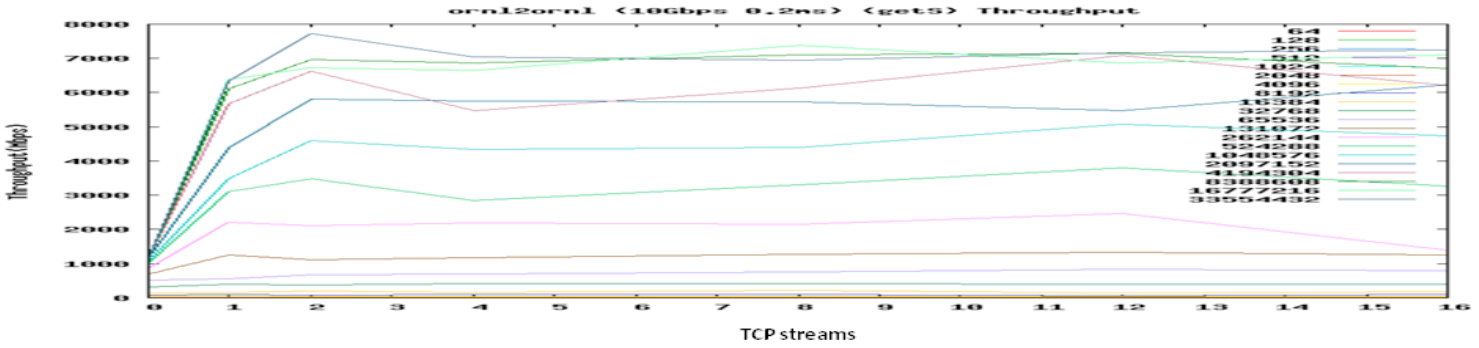
Dynamic Tuning Algorithm



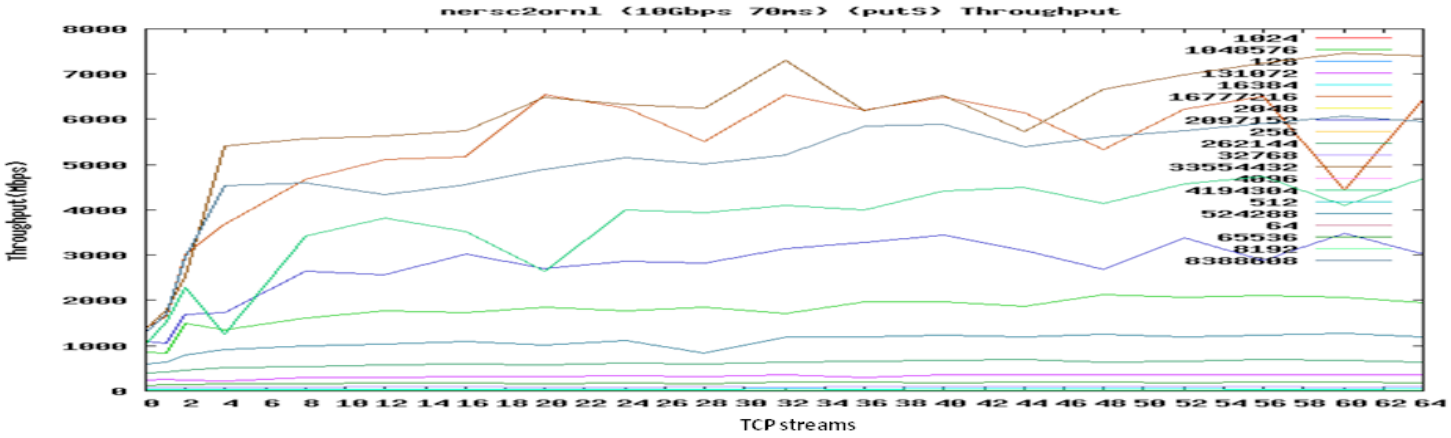
Dynamic Tuning Algorithm



Parallel Streams (estimate starting point)



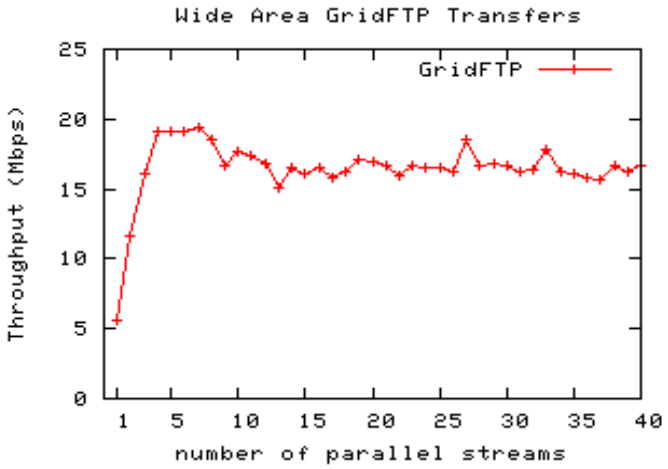
(a)

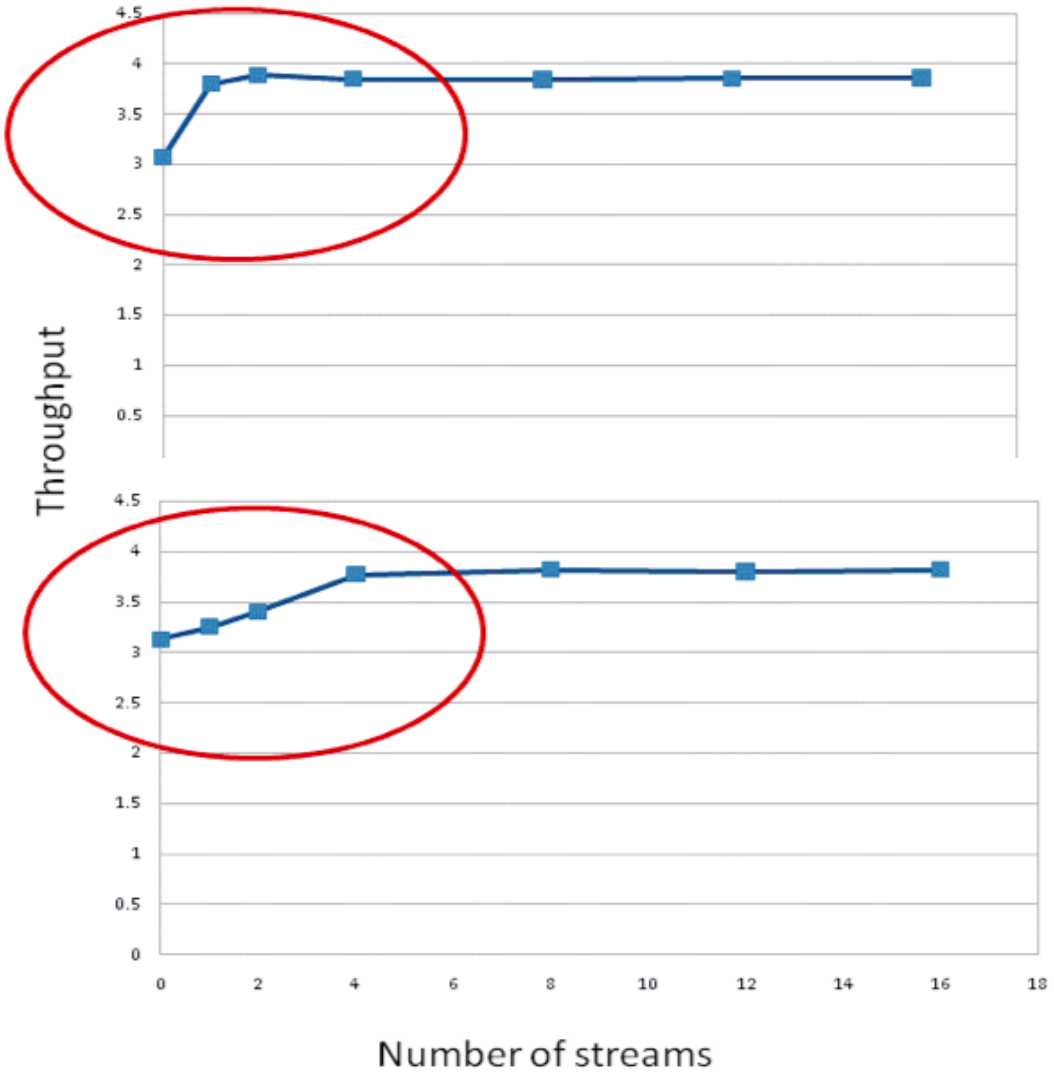


(b)

Log-log scale

Can we predict this behavior?





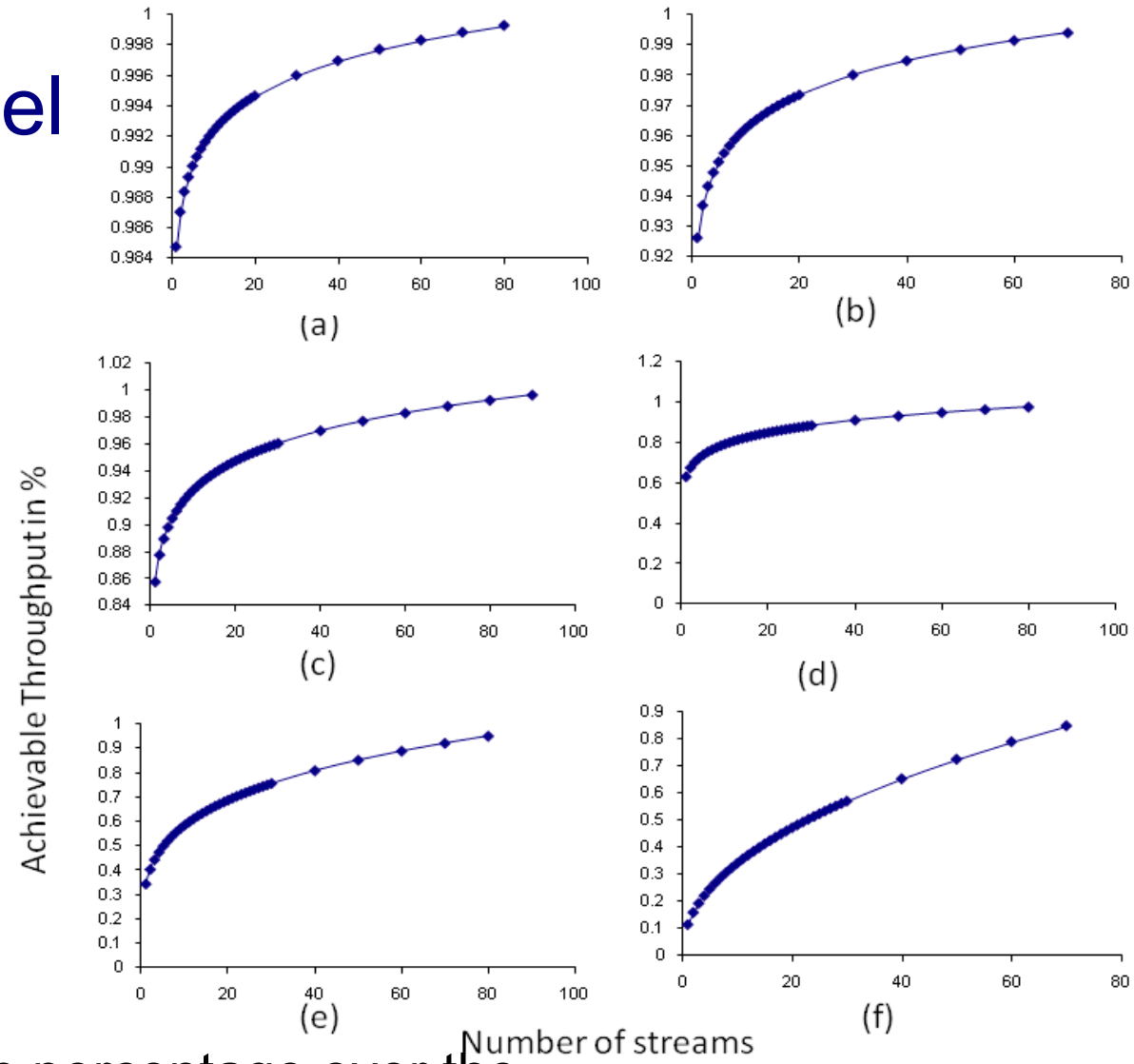
Power-law model

$$T = (n / c)^{(RTT / k)}$$

80-20 rule-Pareto dist.

$$0.8 = (n / c)^{(RTT / k)}$$

$$n = (e^{(k * \ln 0.8 / RTT)}) \cdot c$$



Achievable throughput in percentage over the number of streams with low/medium/high RTT;
(a) RTT=1ms, (b) RTT=5ms, (c) RTT=10ms, (d) RTT=30ms, (e) RTT=70ms, (f) RTT=140ms (c=100 (n/c)<1 k =300 max RRT)

Future Work

Extend power-law model:

Unlike other models in the literature (trying to find an approximation model for the multiple streams and throughput relationship), this model only focuses on the initial behavior of the transfer performance. When RTT is low, the achievable throughput starts high with the low number of streams and quickly approaches to the optimal throughput. When RTT is high, more number of streams is needed for higher achievable throughput.

Get prepared to next-generation networks:

- -100Gbps
- -RDMA

Bulk Data Mover

http://sdm.lbl.gov/bdm/

Earth System Grid

http://www.earthsystemgrid.org

http://esg-pcmdi.llnl.gov/

Support emails

esg-support@earthsystemgrid.org