Evaluation of Rate-based Protocols for Lambda-Grids

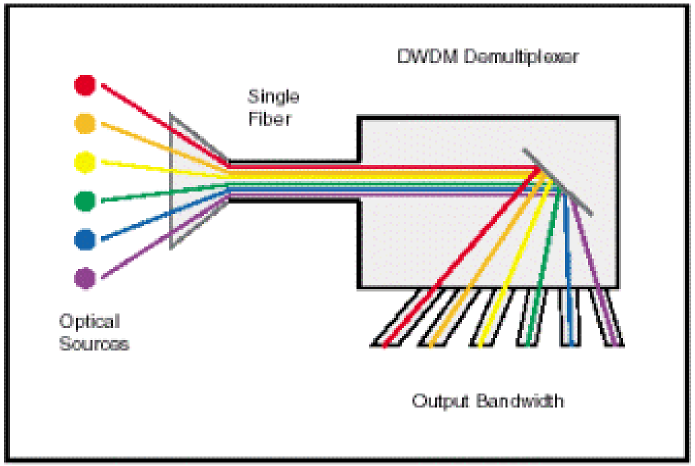
Ryan X. Wu and Andrew A. Chien
Computer Science and Engineering
University of California, San Diego

PFLDnet, Chicago, Illinois
Feb 17, 2004

Outline

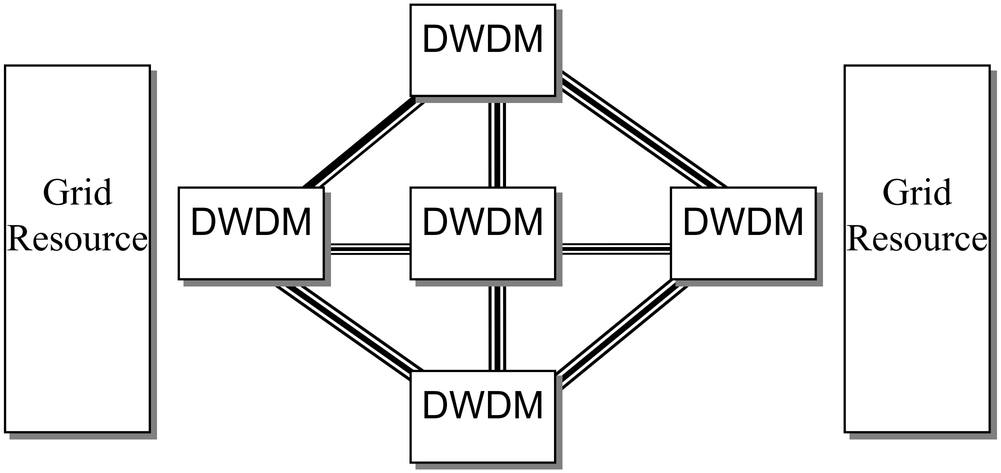
- Communication Challenges in Lambda-Grids
- Rate-based Protocols
- Evaluation
- Related Work
- Conclusion

Lambda-based Communication



The DWDM demultiplexer merges optical sources onto one common fiber,which allows high flexibility in expanding bandwidth.

DWDM(Lambda)



Lambda-Grids

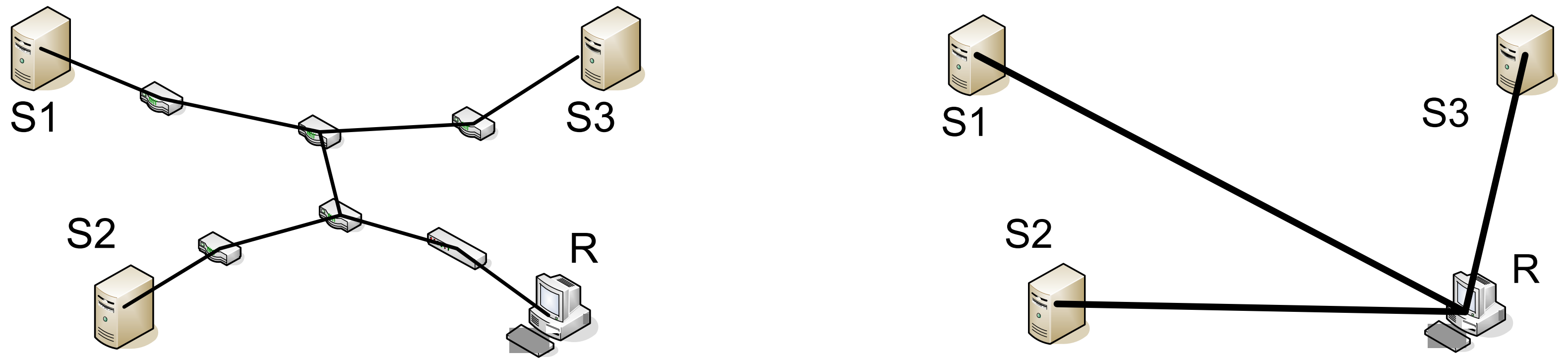
Lambda (wavelength) = end-to-end dedicated optical circuit

DWDM enables a single fiber to have 100’s of lambdas (10Gig) =>Terabits per fiber

Lambda-Grid: shared resource pool connected by on-demand “lambda’s”

Lambda-Grids Differ from Traditional IP Networks

- High speed dedicated connections (optical packet or circuit switching)
- Small number of endpoints (e.g. 10^3 not 10^8)
- Plentiful Network bandwidth: Network >> Computing & I/O speed
- => Congestion moves to the endpoints

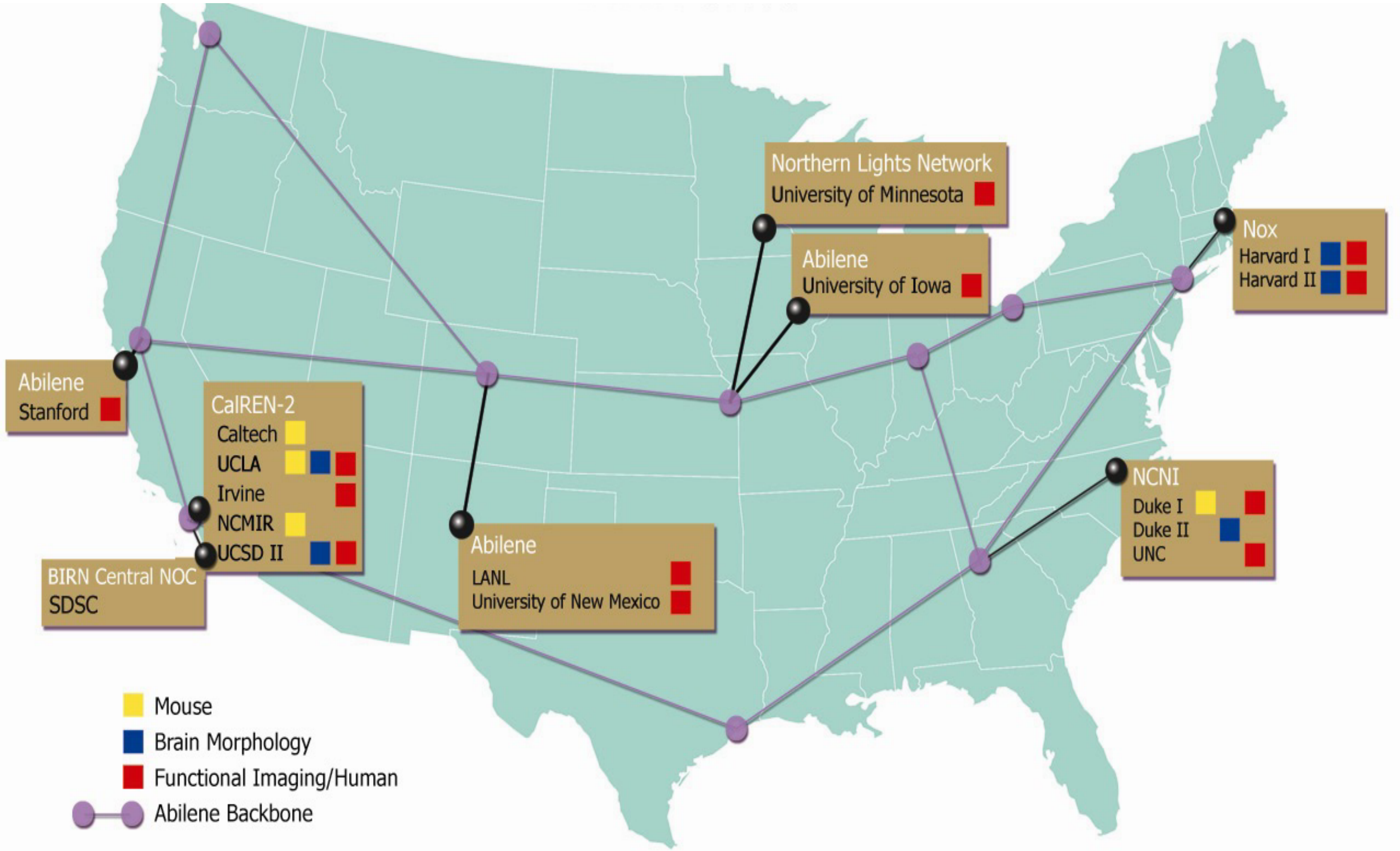


(a) Shared IP Network

(b) Dedicated lambda connections

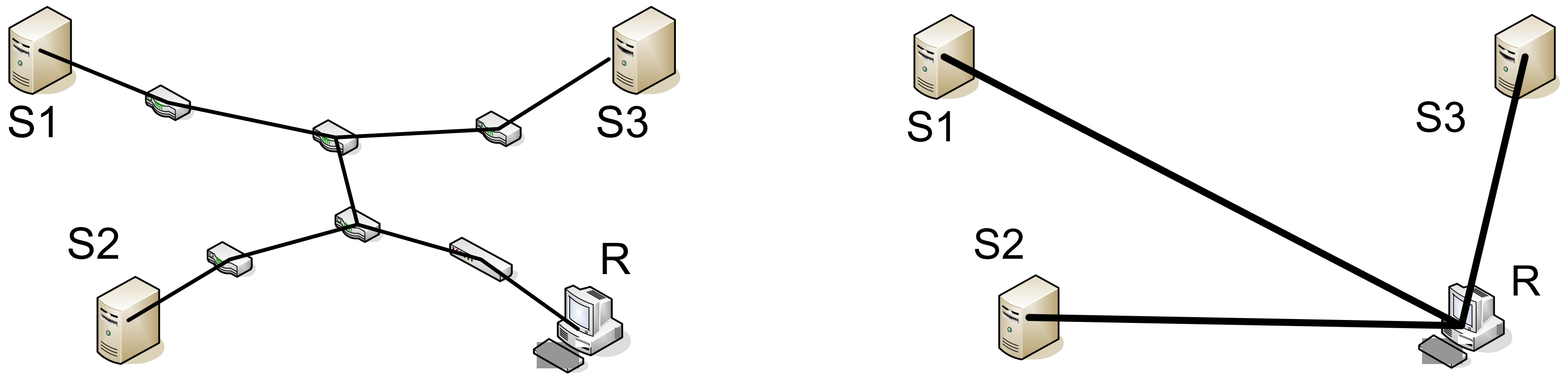
New Communication Patterns

- New applications are multipoint-to-point
  - Example: fetching data from multiple remote storage sites to feed real-time, local data computation needs
- Example: BIRN



Communication Challenges

- **Efficient Point-to-Point**
- **Efficient Multipoint-to-Point**
- **Intra- and Inter- Protocol Fairness**
- **Quick Response to Flow Dynamics**



(a) Shared IP network

(b) Dedicated lambda connections

Rate-based Protocols

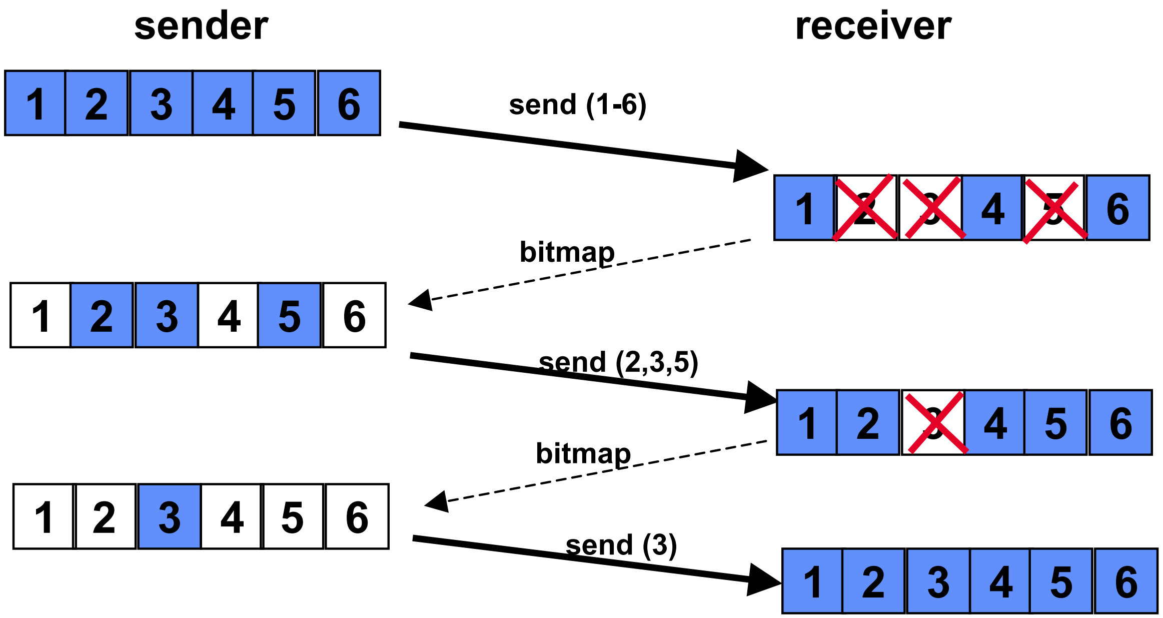
- TCP and its variants for shared, packet switched networks.
  - Internal network congestion; Router assistance.

- Rate-based Protocols to fill high bandwidth-delay product networks
  - Explicitly specified or negotiated transmission rates
  - UDP for data channel (user level implementation)
  - Differ with intended environment of use and performance characteristics

- Three Protocols
  - Reliable Blast UDP (RBUDP) [Leigh, et. al. 2002]
  - Simple Available Bandwidth Utilization Library (SABUL/UDT) [Grossman, et. al. 2003]
  - Group Transport Protocol (GTP) [Wu & Chien 2004]

Reliable Blast UDP (RBUDP)

- Designed for dedicated or QoS enabled links
- Sends data on UDP at fixed rate (user specified)
- Reliability for Payload achieved by Bitmap Tally
  - Send data in series of rounds
  - Received data blocks vector transmitted at the end of each round
- TCP connection used to reliably transmit receive vector information
- No rate adaptation

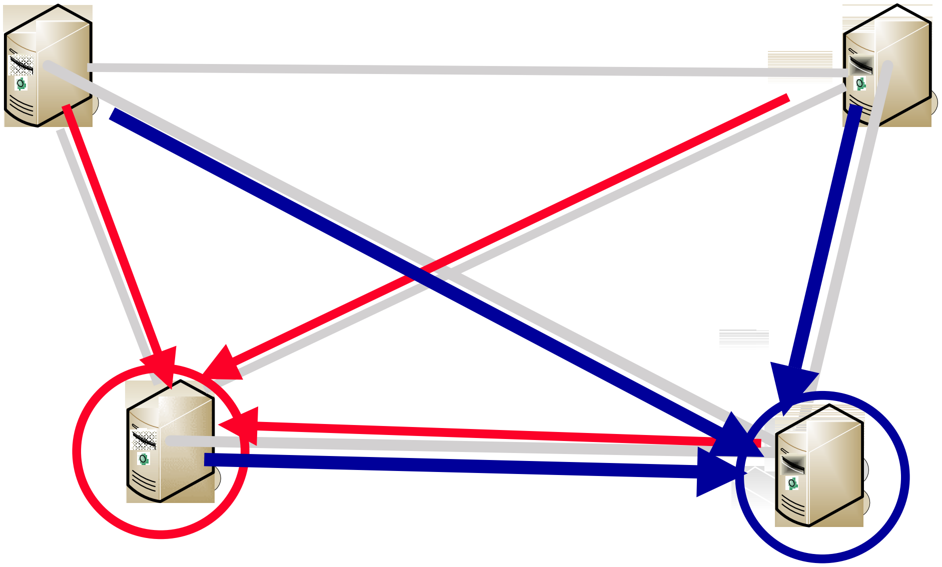


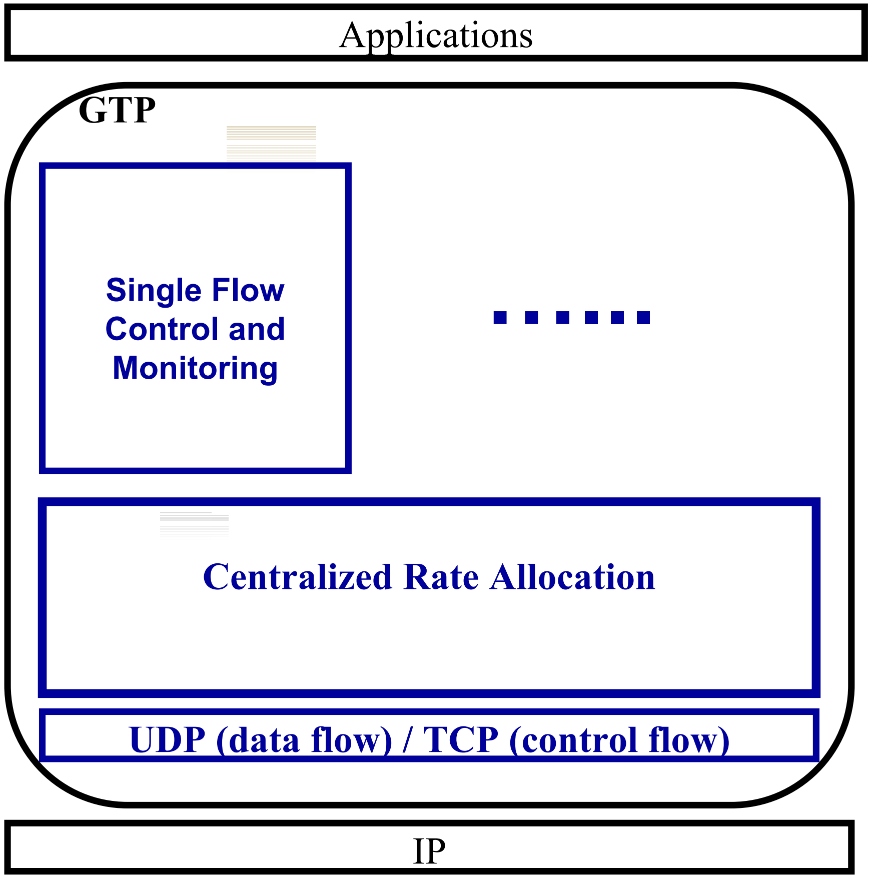
SABUL/UDT

- Designed for shared network
- Sends data on UDP with rate adaptation
- Combination of Rate Control, Window Control, and Delay-based control.
  - Rate control: Slow start, AIMD
  - Window control: Limit number of outstanding packets
  - Delay-based control: Fast response to packet delay
- TCP friendly

Group Transport Protocol: Why Groups?

- Point-to-point protocols do not manage endpoint contention well
- Groups enable cross-flow management
  - Manage concurrent data fetching from multiple senders
  - Clean transitions for rapid change (handoff)
  - Manage fairness across RTTs





Applications

GTP

Single Flow Control and Monitoring

......

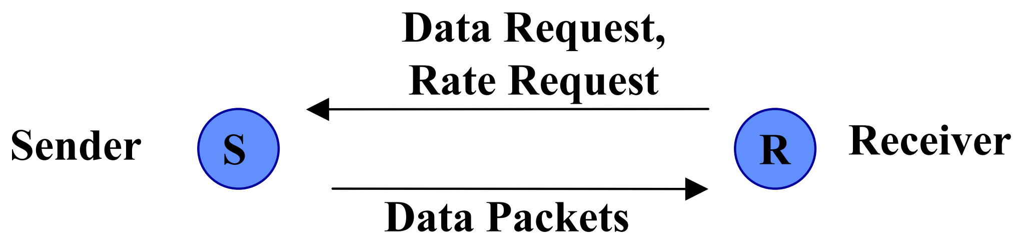
Centralized Rate Allocation

UDP (data flow) / TCP (control flow)

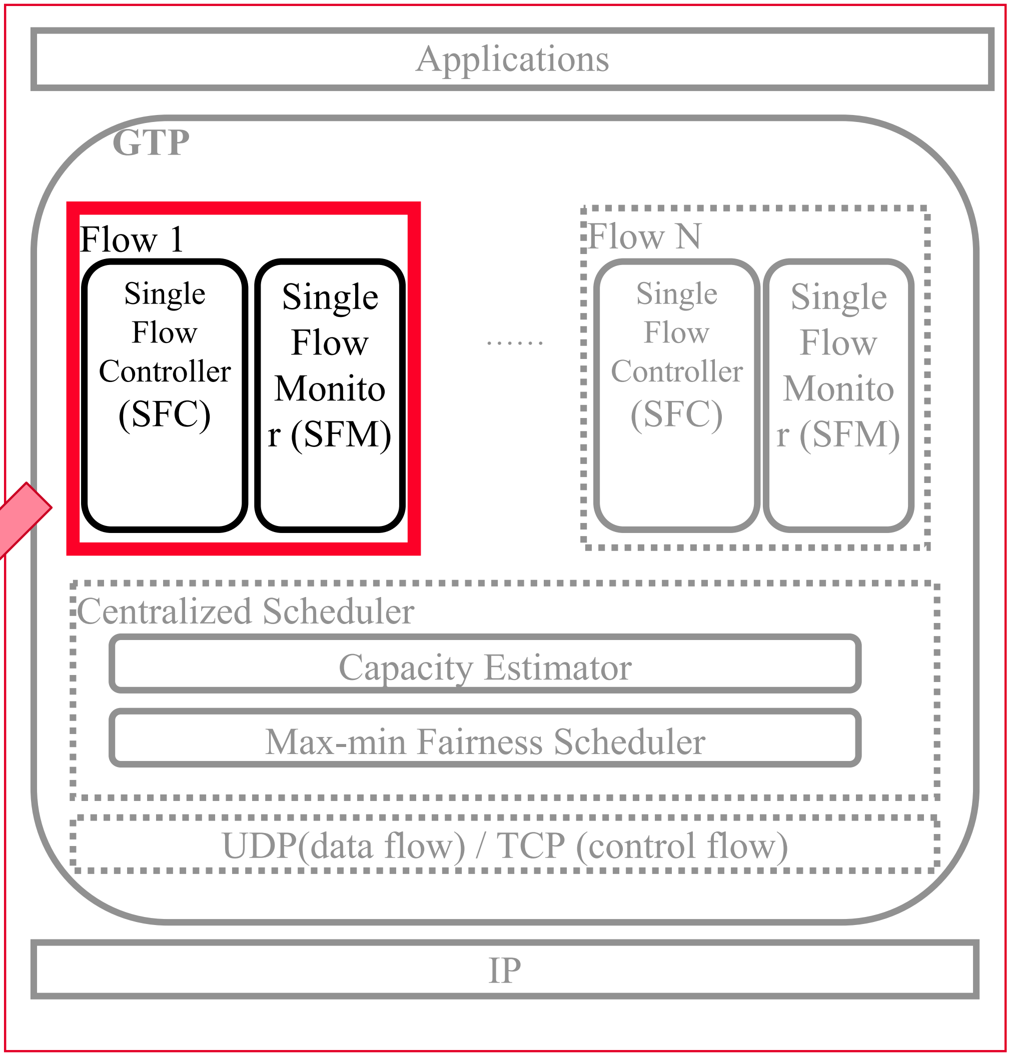
IP

How GTP Works: at Flow Level

- Data and control flows

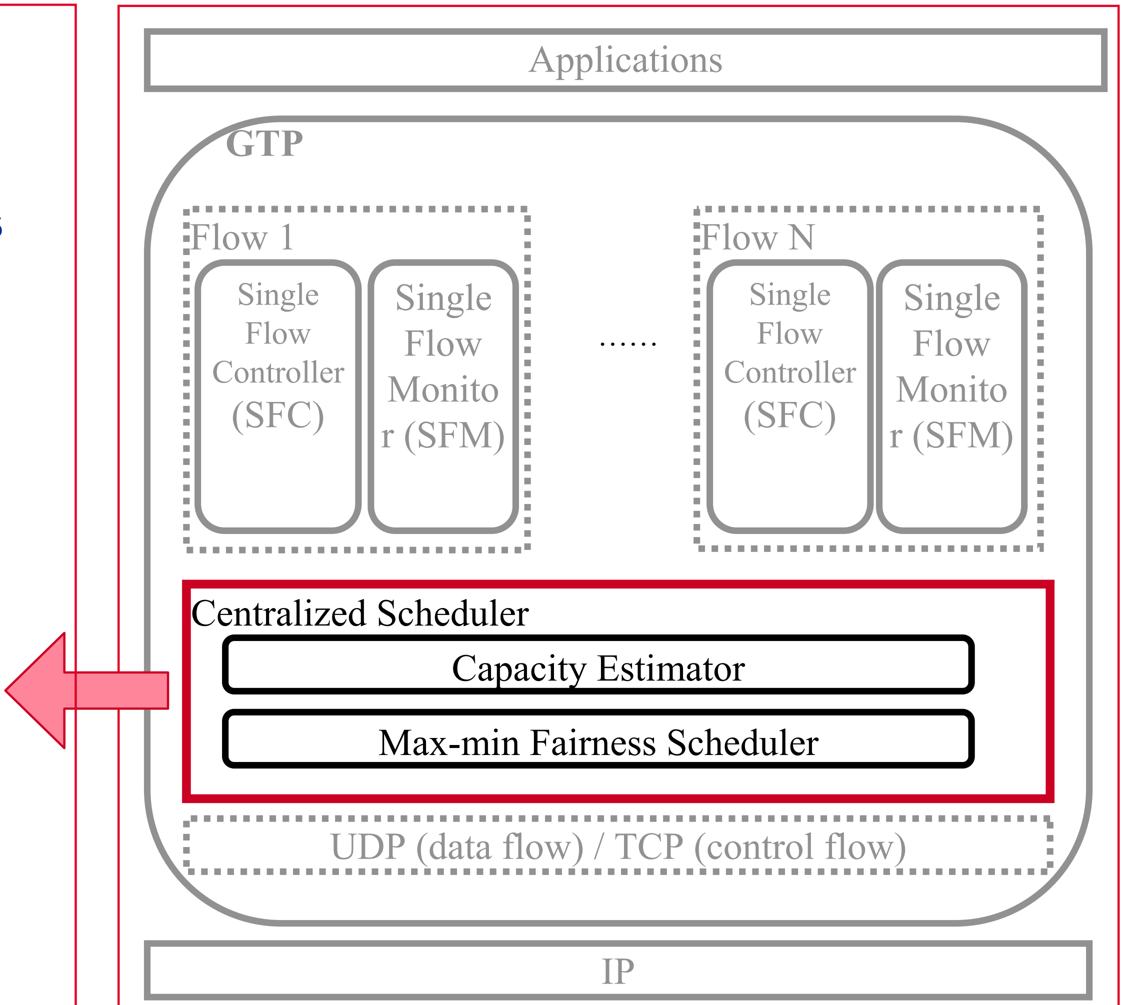


- Sender:
  - Send requested data at receiver-specified rate
- Receiver:
  - Resend data request for loss retransmission
  - Single flow control at RTT level
    - Update flow rate and send rate request to sender
  - Single Flow Monitoring


Applications
GTP
Flow 1
Single Flow Controller (SFC)
Single Flow Monitor (SFM)
……
Flow N
Single Flow Controller (SFC)
Single Flow Monitor (SFM)
Centralized Scheduler
Capacity Estimator
Max-min Fairness Scheduler
UDP(data flow) / TCP (control flow)
IP

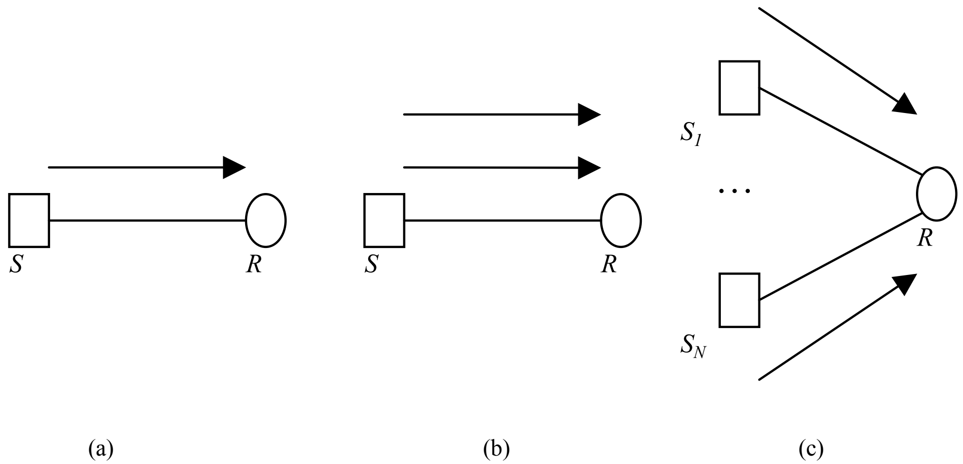
How GTP Works: Central Scheduler

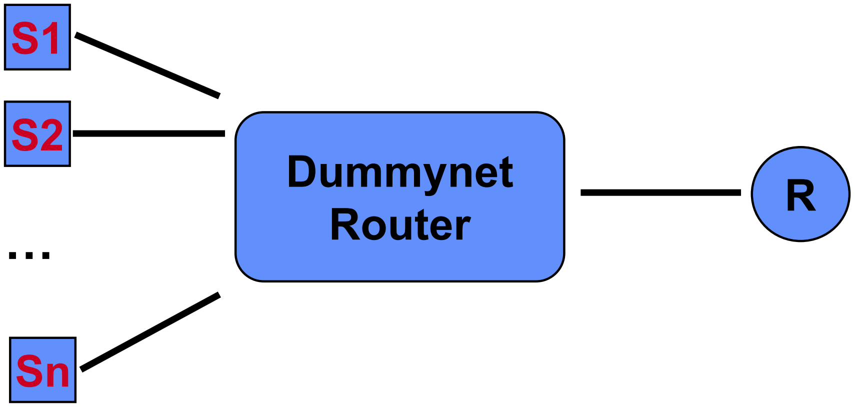
- **Capacity Estimator: for each flow**
  - Calculate the Increment: Exponential increasing and loss proportional decreasing;
  - Update estimated rate
- **Max-min Fair rate allocation**
  - Allocate receiver bandwidth across flows in a fair manner
  - Estimated rates as constrains


Applications
GTP
Flow 1
Single Flow Controller (SFC)
Single Flow Monitor (SFM)
……
Flow N
Single Flow Controller (SFC)
Single Flow Monitor (SFM)
Centralized Scheduler
Capacity Estimator
Max-min Fairness Scheduler
UDP (data flow) / TCP (control flow)
IP

Experiments

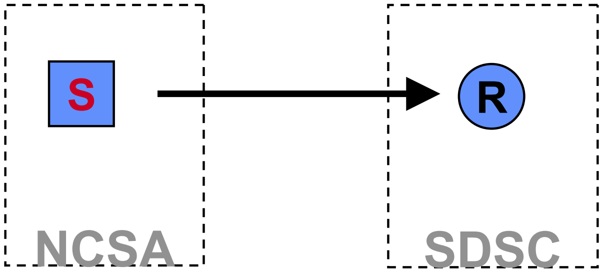
- Dummynet emulation and real measurement on TeraGrid
- Three communication patterns:
  - Single flow; Parallel flows; Converging flows
- Performance metrics
  - Sustained throughput and loss ratio
  - Intra-protocol fairness
  - Inter-protocol fairness
  - Interaction with TCP

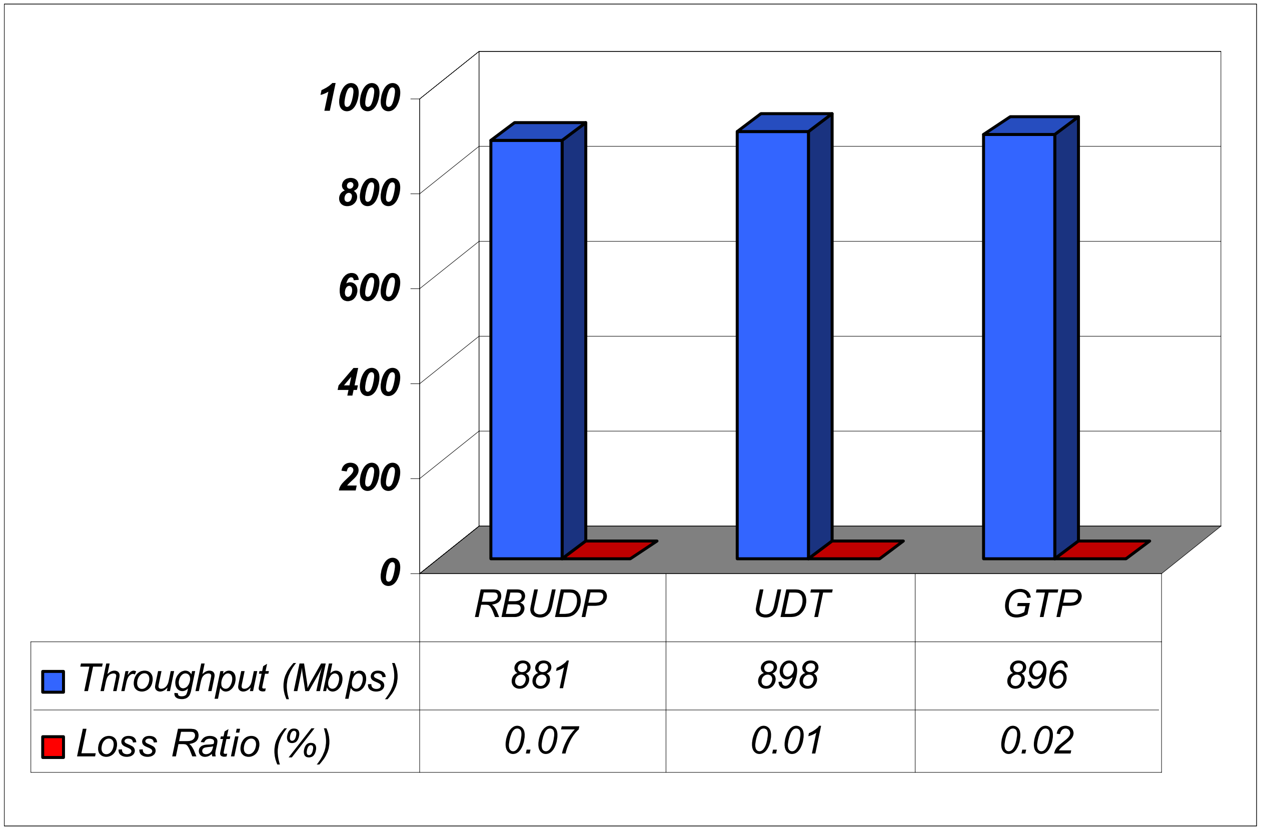




Single Flow Performance

- SDSC -- NCSA, 10GB transfer (1Gbps link capacity), 58ms RTT

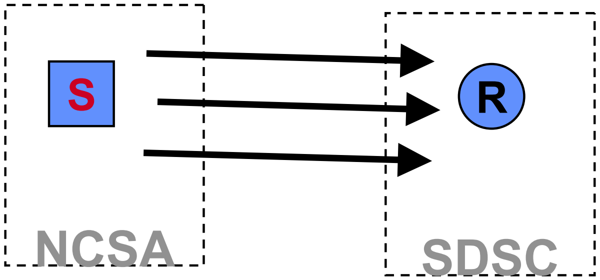


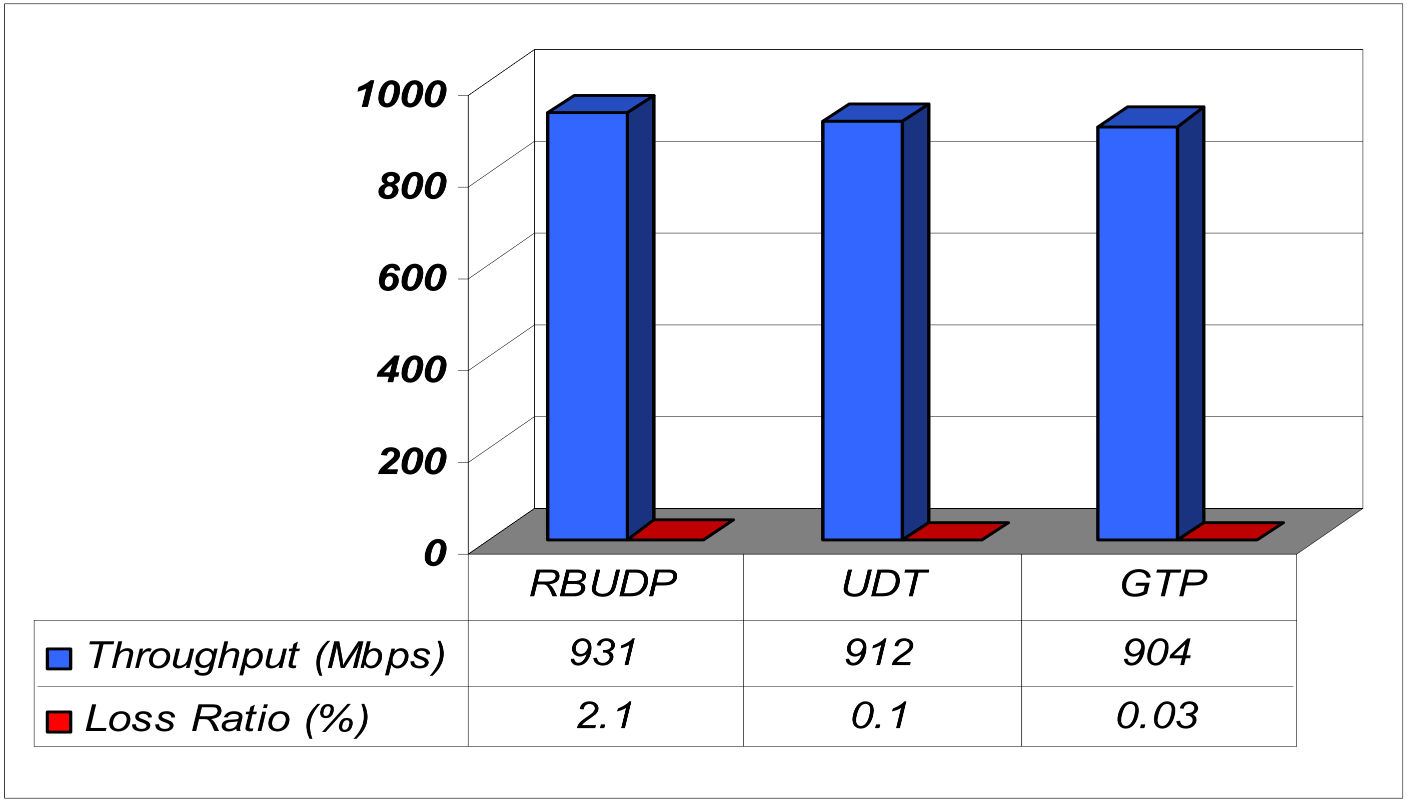


	RBUDP	UDT	GTP
🟦 *Throughput (Mbps)*	*881*	*898*	*896*
🟥 *Loss Ratio (%)*	*0.07*	*0.01*	*0.02*

Parallel Flow Performance

- SDSC -- NCSA, 10GB transfer (1Gbps link capacity), 58ms RTT
- Three parallel flows between sender/receiver

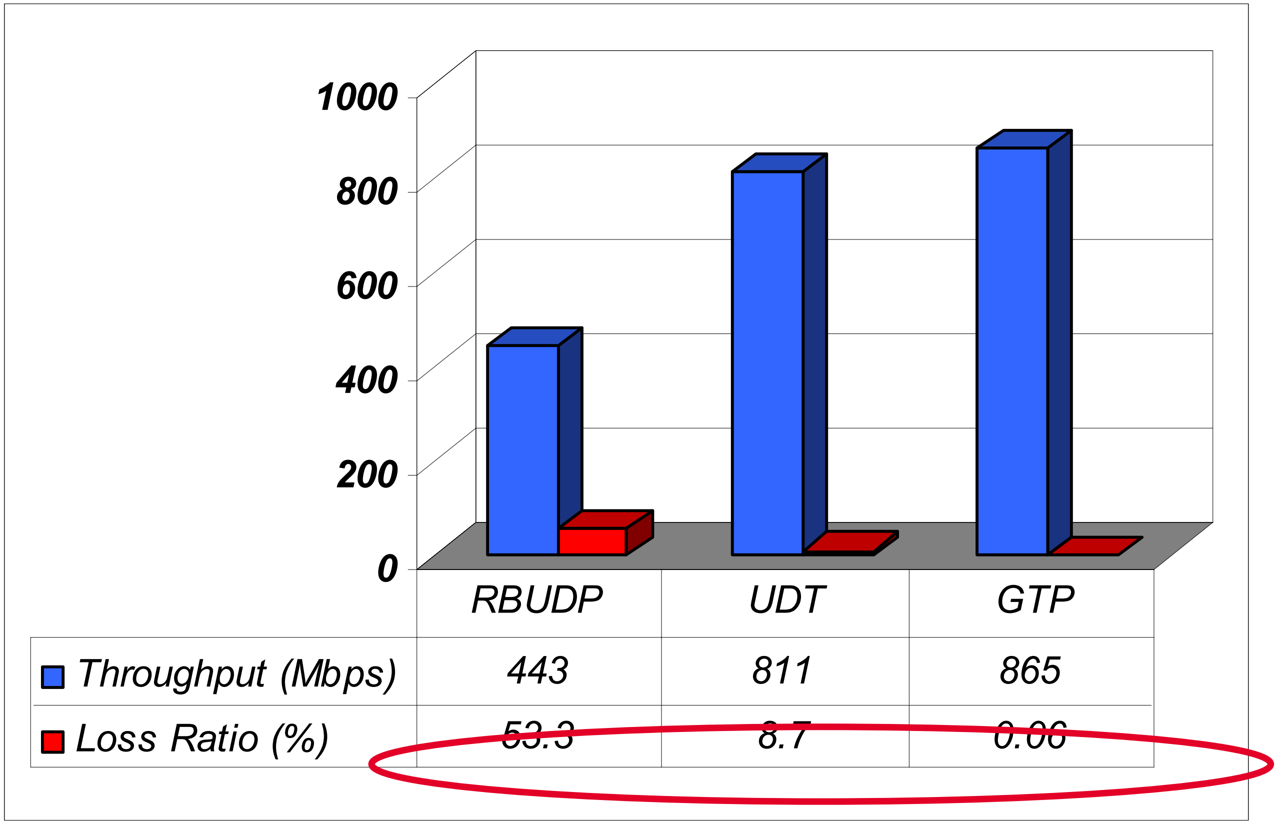




	RBUDP	UDT	GTP
🟦 *Throughput (Mbps)*	*931*	*912*	*904*
🟥 *Loss Ratio (%)*	*2.1*	*0.1*	*0.03*

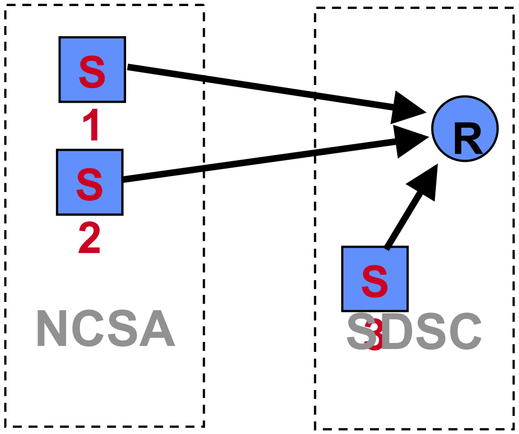
Converging Flow Performance

- **SDSC -- NCSA, 10GB transfer (1Gbps link capacity), 58ms RTT**



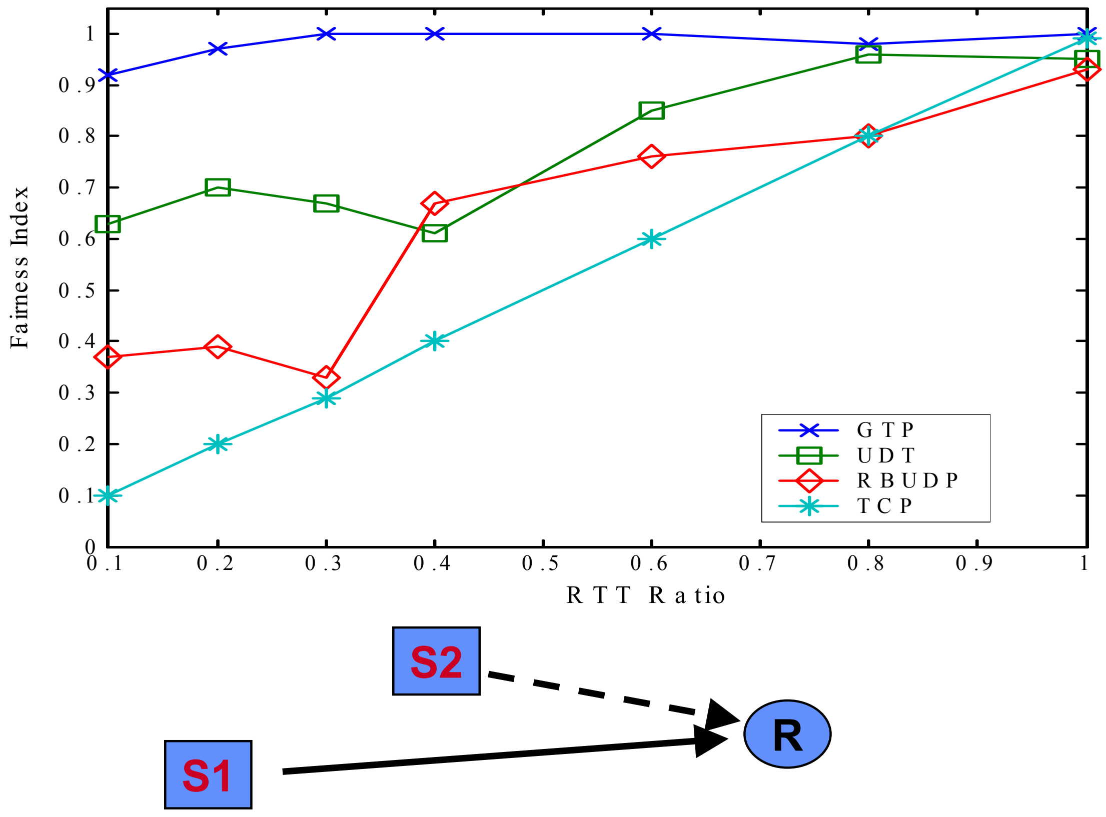
	RBUDP	UDT	GTP
🟦 *Throughput (Mbps)*	*443*	*811*	*865*
🟥 *Loss Ratio (%)*	*53.3*	*8.7*	*0.06*

Converging flows:



Intra-Protocol fairness

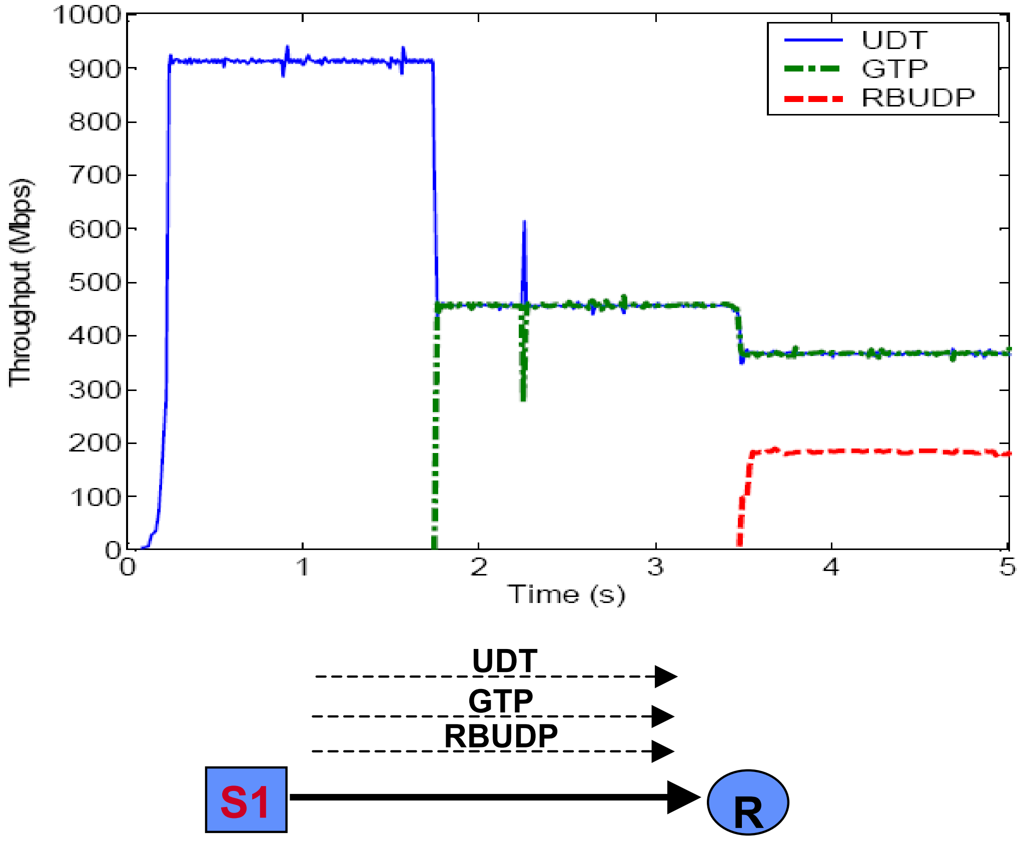
- *Fairness Index = Minimum rate / Maximum rate*
- Fair for converging flows?
- => Others (incl. TCP) don't achieve fairness with variable RTT, GTP does



Two converging flows with diff. RTT

Inter-Protocol Fairness: Parallel Flows

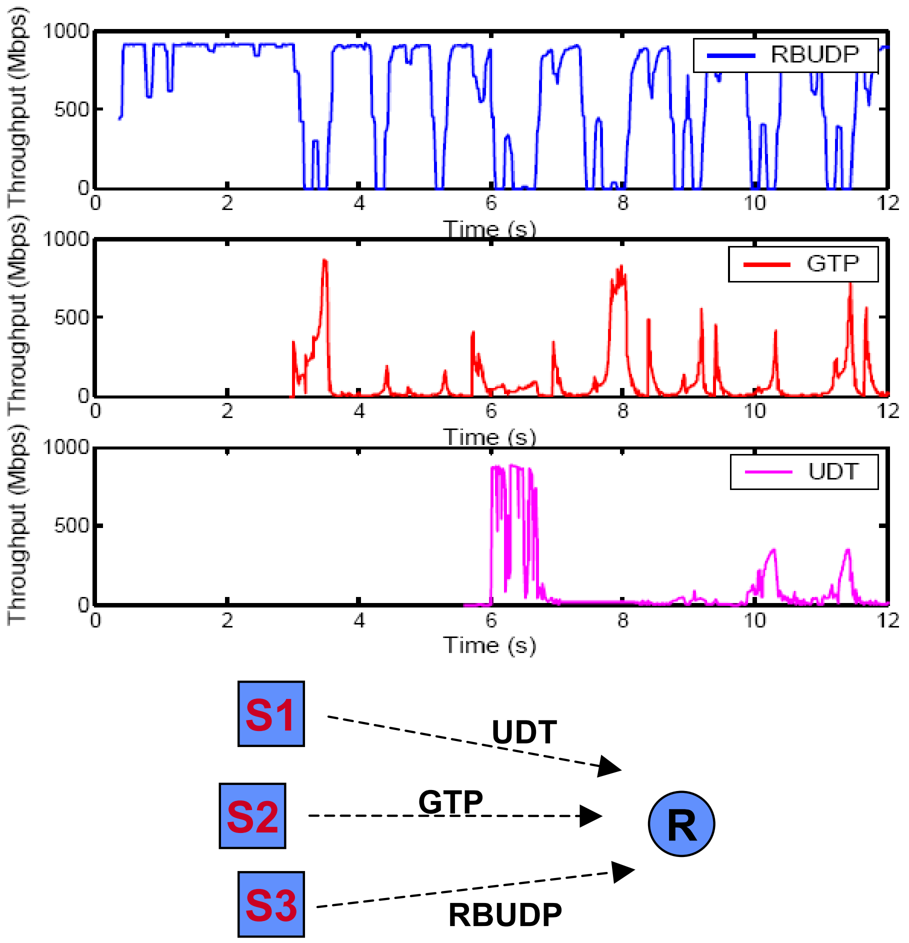
- **Interaction among rate-based protocols: parallel flow case**
- **Conclusion: parallel different aggressiveness**



Single link, parallel flows

Inter-Protocol Fairness: Converging Flows

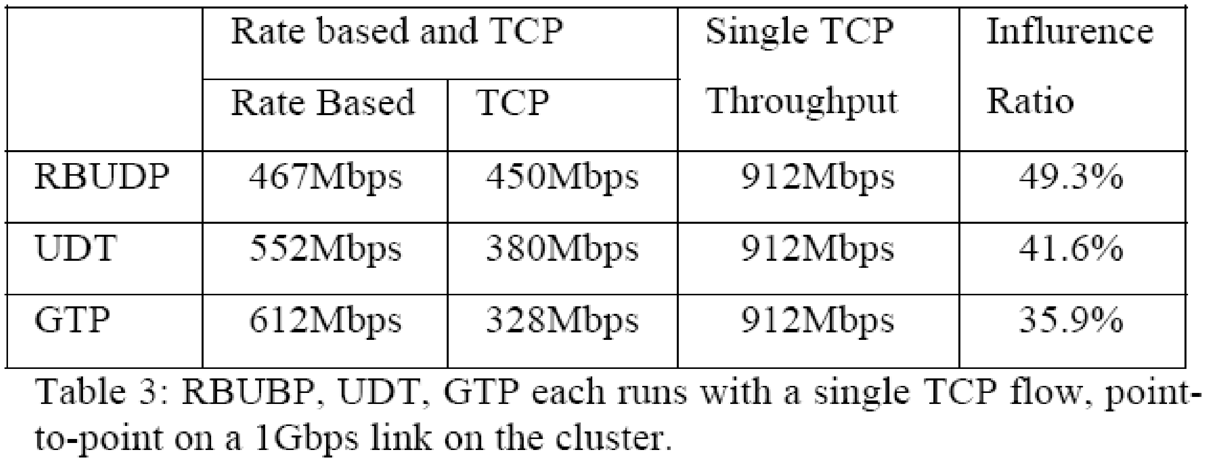
- **Interaction among rate-based protocols: Converging flows**
- **Convergent: don’t coexist nicely – this is a problem**



Converging flows

Inter-Protocol Fairness: Interaction with TCP

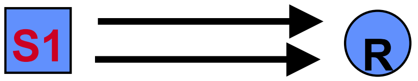
$$\text{Influence ratio} = \frac{\text{TCP throughput in presence of rate-based flow}}{\text{TCP throughput without rate-based flow}}$$

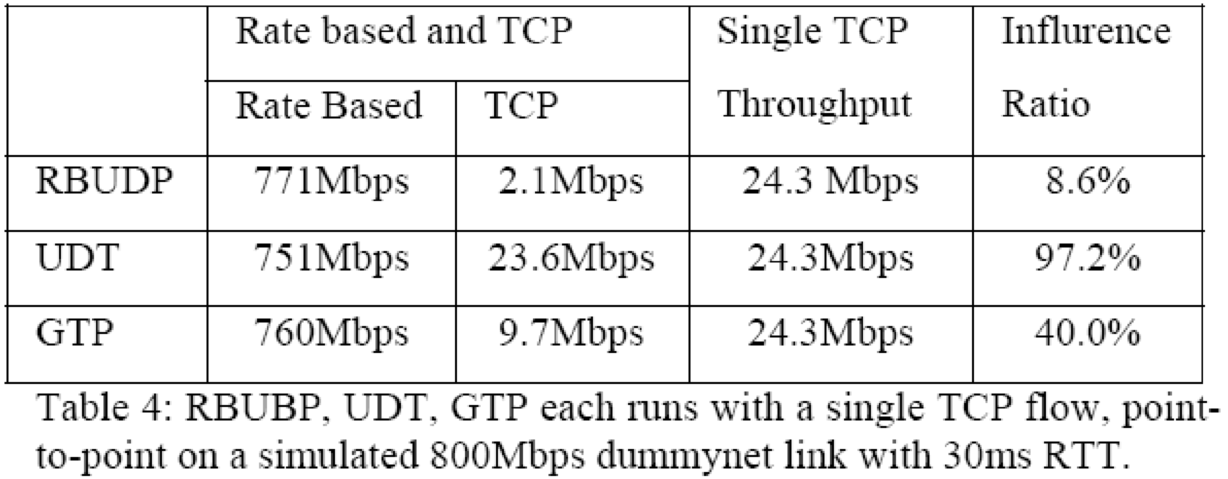


	Rate based and TCP		Single TCP	Influence
	Rate Based	TCP	Throughput	Ratio
RBUDP	467Mbps	450Mbps	912Mbps	49.3%
UDT	552Mbps	380Mbps	912Mbps	41.6%
GTP	612Mbps	328Mbps	912Mbps	35.9%

Table 3: RBUBP, UDT, GTP each runs with a single TCP flow, point-to-point on a 1Gbps link on the cluster.

Parallel flows
0.3ms RTT

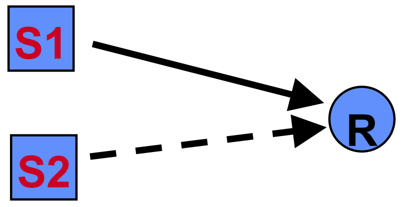




	Rate based and TCP		Single TCP	Influence
	Rate Based	TCP	Throughput	Ratio
RBUDP	771Mbps	2.1Mbps	24.3 Mbps	8.6%
UDT	751Mbps	23.6Mbps	24.3Mbps	97.2%
GTP	760Mbps	9.7Mbps	24.3Mbps	40.0%

Table 4: RBUBP, UDT, GTP each runs with a single TCP flow, point-to-point on a simulated 800Mbps dummynet link with 30ms RTT.

Converging flows
30ms RTT



Related Work

- Other rate based protocols
  - NETBLT, satellite channels [Clark87]
  - RBUDP on Amsterdam—Chicago OC-12 link [Leigh2002]
  - SABUL/UDT [Grossman2003]
  - Tsunami
- Other high speed protocol work
  - HSTCP [Floyd2002]
  - XCP [Katabi2002] and Implementations [USC ISI]
  - FAST TCP[Jin2004]
  - drsTCP[Feng2002]

Summary

- Communications in Lambda-Grids
  - Networks have plentiful bandwidth but limited end-system capacity
  - Endpoint congestion
- Evaluation of Rate-based protocols
  - High performance for point-to-point single or parallel flows
  - Challenging for the case of converging flows
  - GTP outperforms RBUDP and UDT due to its receiver-based schemes
- Remaining challenges
  - End system contention management
  - Interaction with TCP
  - Analytical modeling rate-based control schemes