Predicting Macroscopic Dynamics in Large Distributed Systems

For More: http://www.nist.gov/itl/antd/emergent_behavior.cfm

K. Mills & J. Filliben (NIST), D-Y. Cho (NIH), & E. Schwartz (CMU)

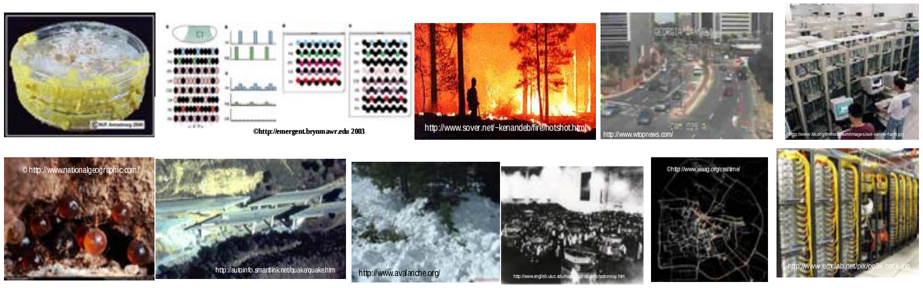
LSN Mini Seminar June 30, 2011

image generated with http://www.wordle.net/ applied to contents of a paper entitled “Predicting Macroscopic Dynamics in Large Distributed Systems” written by Mills, Filliben, Cho and Schwartz

What are complex systems?

Large collections of interconnected components whose interactions lead to macroscopic behaviors

- – Biological systems (e.g., slime molds, ant colonies, embryos)
- – Physical systems (e.g., earthquakes, avalanches, forest fires)
- – Social systems (e.g., transportation networks, cities, economies)
- – Information systems (e.g., Internet, Web services, compute grids)

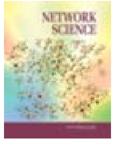

©http://emergent.brynmawr.edu 2003
http://www.sover.net/~kenandeb/fire/hotshot.html
http://www.wtopnews.com/
http://www.kk.org/thetechnium/images/aol-server-farm.jpg
© http://www.nationalgeographic.com/
http://autoinfo.smartlink.net/quake/quake.htm
http://www.avalanche.org/
http://www.english.uiuc.edu/maps/depression/photoessay.htm
©http://www.waag.org/realtime/
http://www.emulab.net/pix/pc3k-back.jpg

What is the problem?

No one understands how to measure, predict or control macroscopic behavior in complex information systems

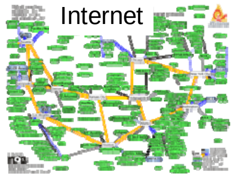
"[Despite] society's profound dependence on networks, fundamental knowledge about them is primitive. [G]lobal communication … networks have quite advanced technological implementations but their behavior under stress still cannot be predicted reliably.… There is no science today that offers the fundamental knowledge necessary to design large complex networks [so] that their behaviors can be predicted prior to building them."

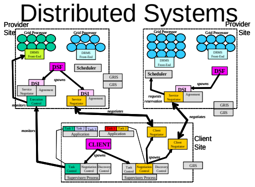
— Network Science 2006, recently released NRC report



What is the new idea?

Leverage models and mathematics from the physical sciences to define a systematic method to measure, understand and control macroscopic behavior in the Internet and distributed software systems built on the Internet


Internet


Distributed Systems

Technical Approach

- Evaluate models and analysis methods
  - Computationally tractable?
  - Reveal macroscopic behavior?
  - Establish causality?
- Evaluate distributed control techniques
  - Can economic mechanisms elicit desired behaviors?
  - Can biological mechanisms organize elements?

Hard Issues & Approaches Investigated

Hard Issues	Solutions Investigated and Evaluated
1. Model Scale	• Model restriction and parameter clustering (leading to MesoNet and Koala) • 2-level experiment designs • Orthogonal fractional factorial (OFF) experiment designs • Markov chains
2. Model Validation	• Sensitivity analysis • Key comparisons with empirical results in small topologies • Generating Markov chain models from discrete-event simulations
3. Tractable Analysis	• Correlation analysis with clustering • Principal components analysis • 10-step graphical analysis • Cluster analysis • Custom multidimensional visualizations • Exploratory interactive multidimensional visualization • Eigenanalysis of matrices
4. Causal Analysis	• Principal components analysis • Detailed measurements of model behavior • Time series analysis • Hypothesis testing • Exploratory analyses • Cut set analysis of graphs and perturbation of Markov chain models
5. Controlling Behavior	• Economic algorithms for resource allocation in computational grids • Proposed Internet congestion control algorithms • Heuristics for resource allocation in infrastructure clouds

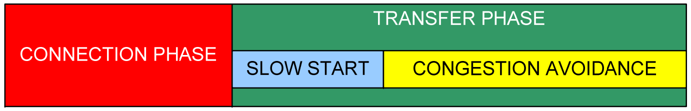
Outline

- Introduction to the Internet Congestion-Control Problem
- Methods to Reduce Stimulus and Response State Spaces
- Selected Methods to Analyze Multidimensional Datasets
- Application to Predict Macroscopic Dynamics of Proposed TCP Replacements

Introduction to the Problem

Internet Congestion Control & TCP

Phases in a TCP Flow



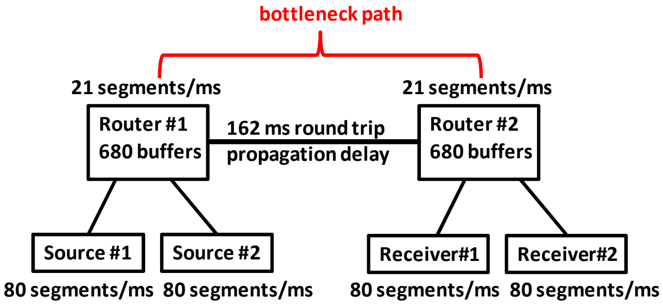
CONNECTION PHASE

TRANSFER PHASE

SLOW START

CONGESTION AVOIDANCE

252 Mbps Bottleneck Path



bottleneck path

21 segments/ms

21 segments/ms

Router #1 680 buffers

162 ms round trip propagation delay

Router #2 680 buffers

Source #1

Source #2

Receiver#1

Receiver#2

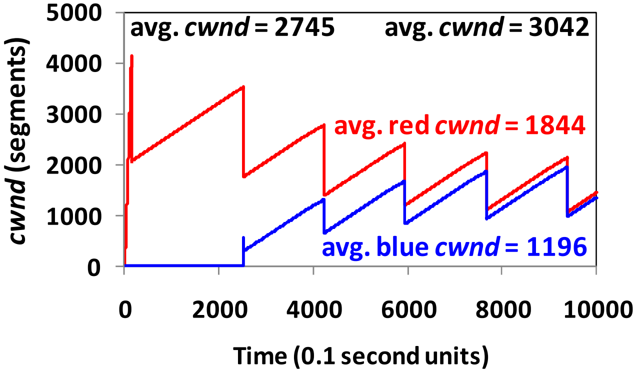
80 segments/ms

80 segments/ms

80 segments/ms

80 segments/ms

TCP Flows Achieve 89 % Utilization



Utilizations Much Lower on High-Speed, Long-Delay Paths

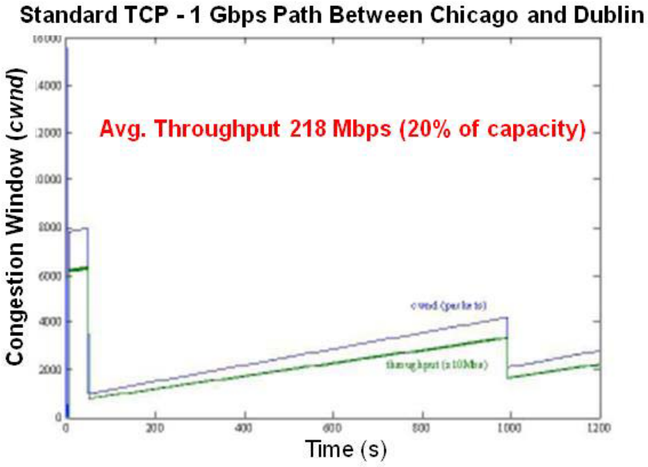
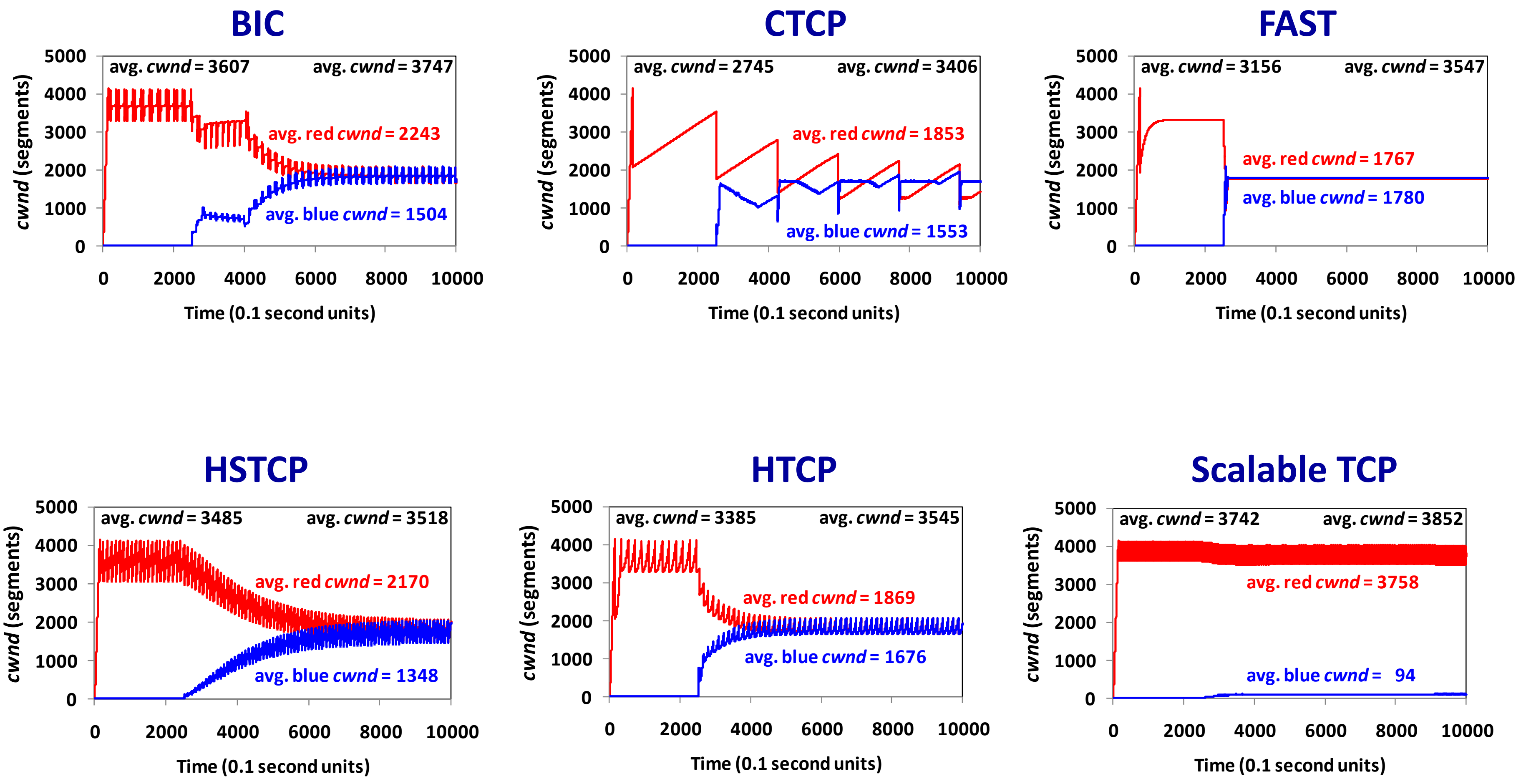


Figure 1 from Li et al. 2007. Experimental Evaluation of TCP Protocols for High-Speed Networks. *Transactions on Networking*. 15:5, 1109-1122.

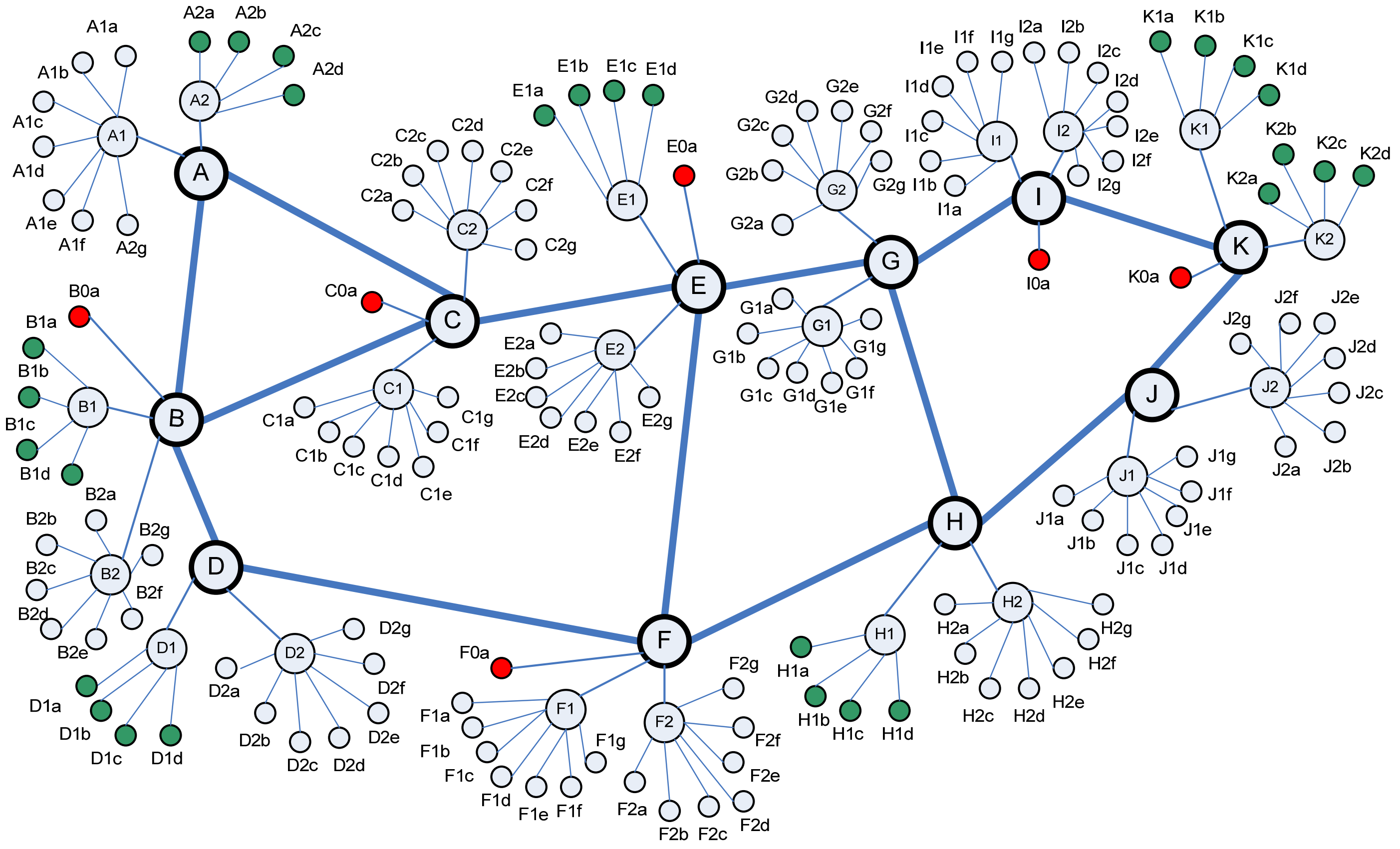
So Researchers Proposed TCP Replacements

Identifier	Acronym	Name of Algorithm
1	BIC	Binary Increase Congestion Control
2	CTCP	Compound TCP
3	FAST	Fast Active Queue Management Scalable TCP
4	HSTCP	High Speed TCP
5	HTCP	Hamilton TCP
6	STCP	Scalable TCP
7	TCP	Transmission Control Protocol

Proposed TCP Replacements Well-Studied on Bottleneck Path



What would be the macroscopic dynamics if these proposed TCP replacements were deployed in the Internet?



Four-Tier Topology (based on the Abilene Network) with 11 Backbone Routers (A-K), 22 Point of Presence Routers (A1-K2) and 139 Access Routers (A1a-K2d) – 6 red and 28 green Access Routers may operate at different speeds from the 105 others – not shown is a fourth tier of hundreds of thousands of attached computers

Stimulus and Response State Spaces Are Large for Network Simulation Models

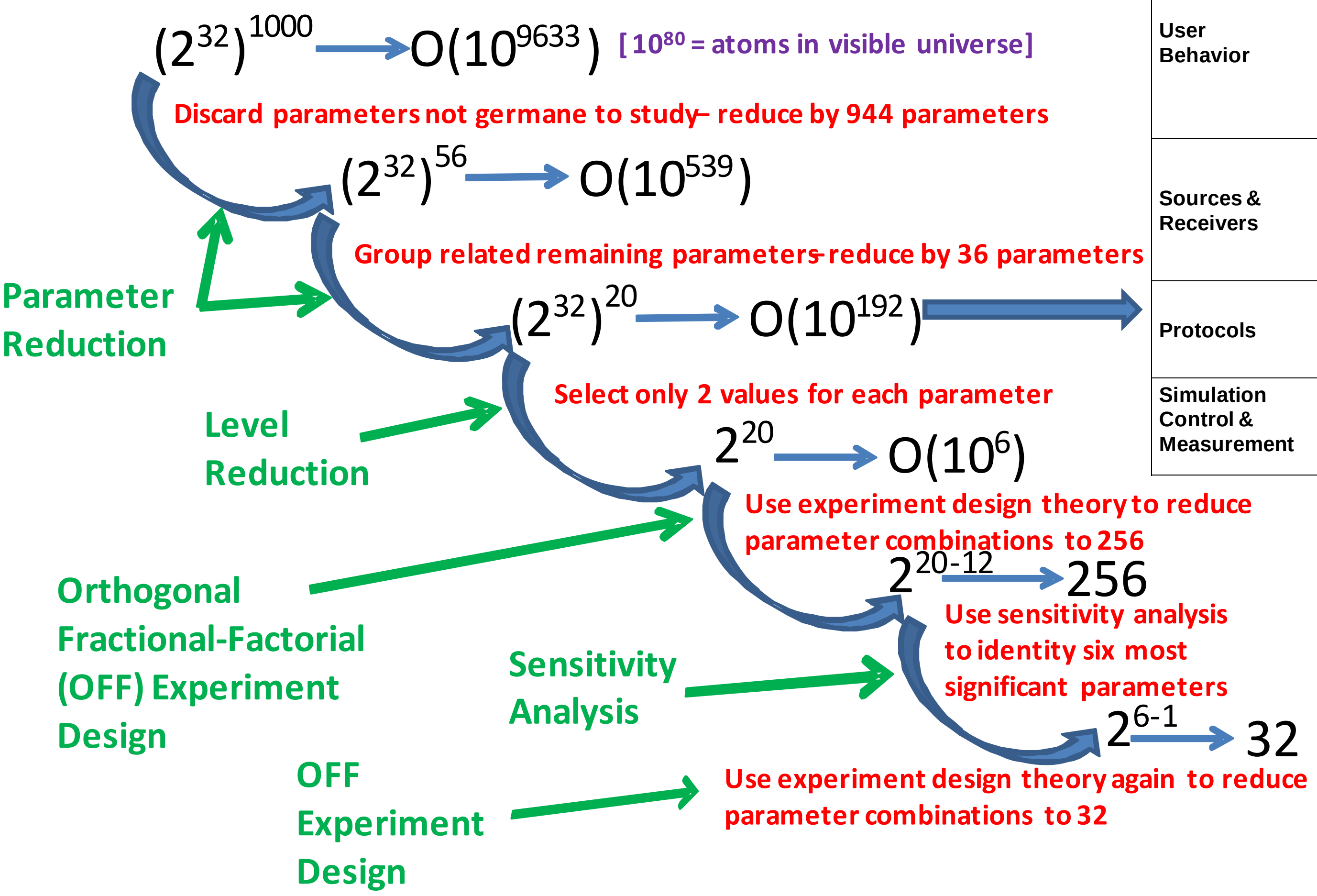
$$\underbrace{y_1, \ldots, y_z}_{\textbf{Response State-Space}} = \mathrm{f}(\underbrace{x_{1|[1,\ldots,\ell]} \ldots, x_{p|[1,\ldots,\ell]})}_{\textbf{Stimulus State-Space}}$$

e.g., $z = 500$

e.g., $p = 1000$ and $\ell = 2^{32}$

Methods to Reduce the Scale of the Stimulus and Response Spaces

We Employ Stimulus State-Space Reduction Techniques



$(2^{32})^{1000} \rightarrow O(10^{9633})$ [10^{80} = atoms in visible universe]

Discard parameters not germane to study – reduce by 944 parameters

$(2^{32})^{56} \rightarrow O(10^{539})$

Group related remaining parameters-reduce by 36 parameters

$(2^{32})^{20} \rightarrow O(10^{192})$

Parameter Reduction

Select only 2 values for each parameter

$2^{20} \rightarrow O(10^{6})$

Level Reduction

Use experiment design theory to reduce parameter combinations to 256

$2^{20-12} \rightarrow 256$

Orthogonal Fractional-Factorial (OFF) Experiment Design

Use sensitivity analysis to identity six most significant parameters

$2^{6-1} \rightarrow 32$

Sensitivity Analysis

Use experiment design theory again to reduce parameter combinations to 32

OFF Experiment Design

NIST *MesoNet* Simulator

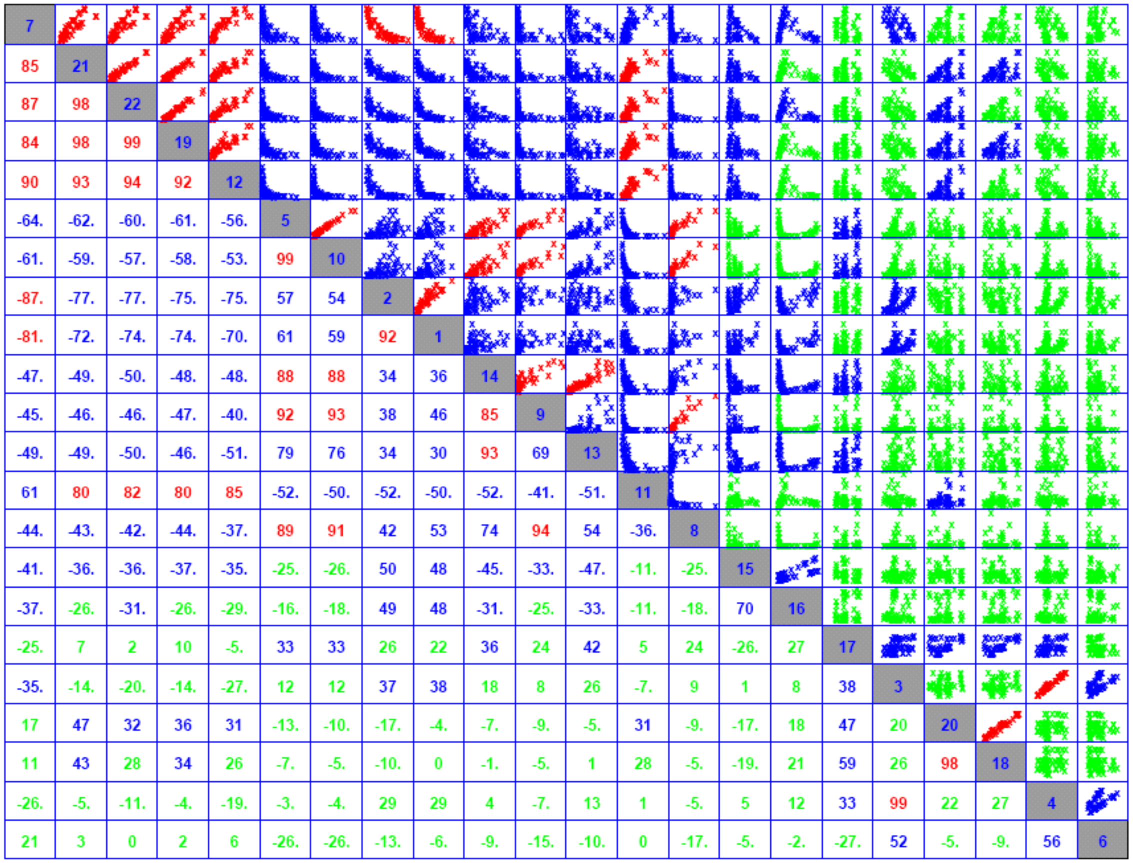
Category	Identifier	Name
Network Configuration	X1	Network Speed
	X2	Propagation Delay
	X3	Buffer Provisioning
	X4	Topology
User Behavior	X5	Web Object Size for Browsing
	X6	Proportion & Size of Larger Files
	X7	Think Time
	X8	Patience
	X9	Selected Spatiotemporal Congestion
	X10	Long-lived Flows
Sources & Receivers	X11	Source & Receiver Interface Speeds
	X12	Number of Sources & Receivers
	X13	Distribution of Sources
	X14	Distribution of Receivers
Protocols	X15	Congestion Control Procedures
	X16	Initial Congestion Window
	X17	Initial Slow Start Threshold
Simulation Control & Measurement	X18	Measurement Interval Size
	X19	Simulation Duration
	X20	Startup Pattern

We Employ Response State-Space Reduction Techniques
(here we demonstrate correlation analysis & clustering)

Response	Definition
y1	Number of sources transmitting
y2	Proportion transmitting [y1 / all sources]
y3	Number segments entering network
y4	Number of segments leaving network
y5	Loss Rate [y4/ (y3 + y4)]
y6	Flows completing per interval
y7	Flow-completion rate [y6/(y6+y1)]
y8	Connection failures per interval
y9	Connection-failure rate [y8/(y8+y1)]
y10	Retransmission rate
y11	Congestion window (*cwnd*)
y12	cwnd increases per interval
y13	Duplicate ACKs per interval
y14	Timeouts per interval
y15	Round-trip time (*rtt*)
y16	Queuing delay

Response	Definition
y17	Average Segments/Second on DD flows
y18	Average Segments/Second on DF flows
y19	Average Segments/Second on DN flows
y20	Average Segments/Second on FF flows
y21	Average Segments/Second on FN flows
y22	Average Segments/Second on NN flows

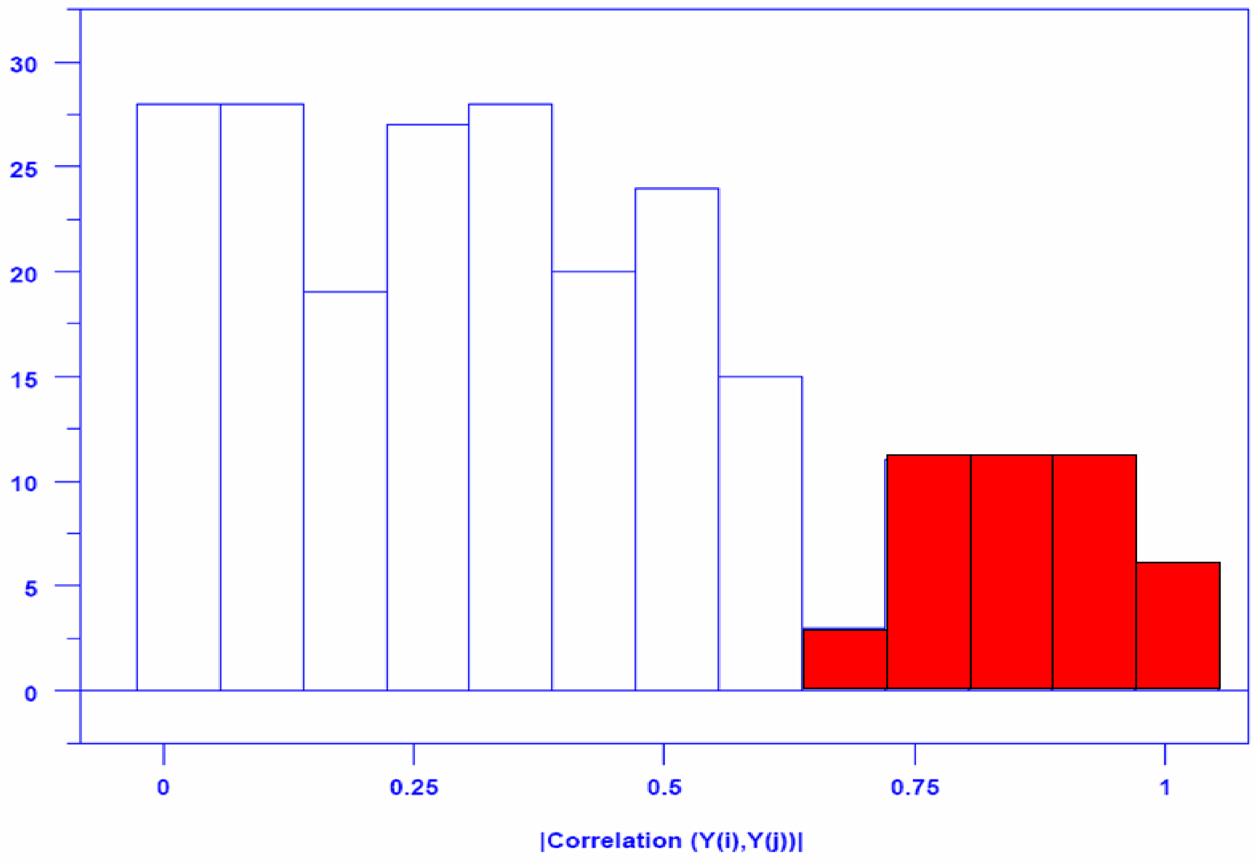
Response Correlation Matrix



7																					
85	21																				
87	98	22																			
84	98	99	19																		
90	93	94	92	12																	
-64.	-62.	-60.	-61.	-56.	5																
-61.	-59.	-57.	-58.	-53.	99	10															
-87.	-77.	-77.	-75.	-75.	57	54	2														
-81.	-72.	-74.	-74.	-70.	61	59	92	1													
-47.	-49.	-50.	-48.	-48.	88	88	34	36	14												
-45.	-46.	-46.	-47.	-40.	92	93	38	46	85	9											
-49.	-49.	-50.	-46.	-51.	79	76	34	30	93	69	13										
61	80	82	80	85	-52.	-50.	-52.	-50.	-52.	-41.	-51.	11									
-44.	-43.	-42.	-44.	-37.	89	91	42	53	74	94	54	-36.	8								
-41.	-36.	-36.	-37.	-35.	-25.	-26.	50	48	-45.	-33.	-47.	-11.	-25.	15							
-37.	-26.	-31.	-26.	-29.	-16.	-18.	49	48	-31.	-25.	-33.	-11.	-18.	70	16						
-25.	7	2	10	-5.	33	33	26	22	36	24	42	5	24	-26.	27	17					
-35.	-14.	-20.	-14.	-27.	12	12	37	38	18	8	26	-7.	9	1	8	38	3				
17	47	32	36	31	-13.	-10.	-17.	-4.	-7.	-9.	-5.	31	-9.	-17.	18	47	20	20			
11	43	28	34	26	-7.	-5.	-10.	0	-1.	-5.	1	28	-5.	-19.	21	59	26	98	18		
-26.	-5.	-11.	-4.	-19.	-3.	-4.	29	29	4	-7.	13	1	-5.	5	12	33	99	22	27	4	
21	3	0	2	6	-26.	-26.	-13.	-6.	-9.	-15.	-10.	0	-17.	-5.	-2.	-27.	52	-5.	-9.	56	6

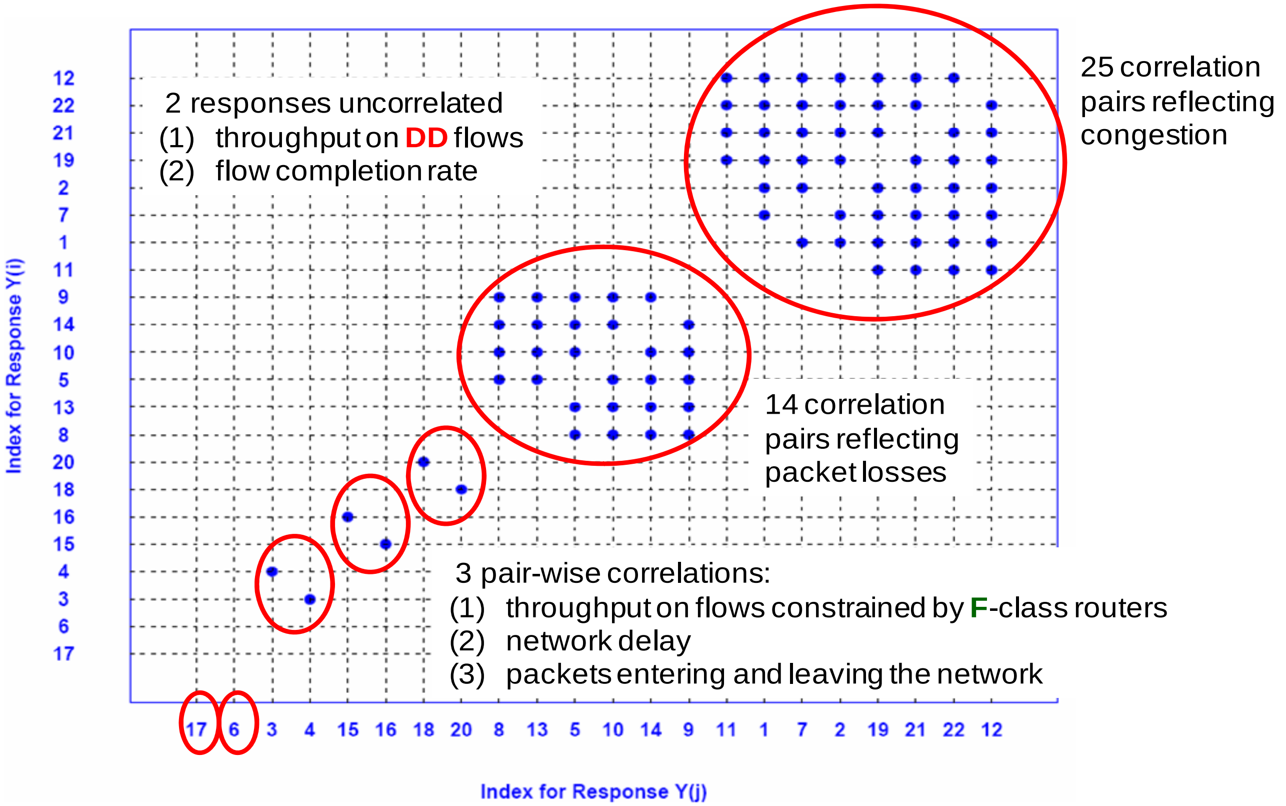
Red 80 ≥ |r|x100 ≤ 100 Blue 30 ≥ |r|x100 < 80 Green |r|x100 < 30

Decide Which Pairs of Correlated Responses to Analyze



Frequency distribution of |*r*| for pair-wise correlations – bins highlighted for |*r*| > 0.65

Cluster Retained Correlated Responses



2 responses uncorrelated
(1) throughput on **DD** flows
(2) flow completion rate

25 correlation pairs reflecting congestion

14 correlation pairs reflecting packet losses

3 pair-wise correlations:
(1) throughput on flows constrained by **F**-class routers
(2) network delay
(3) packets entering and leaving the network

Index-Index Plot Sorted by Increasing Count of Correlated Pairs to Indentify Clusters of Mutual Correlations that Represent Seven Behavioral Dimensions

Selected Methods to
Analyze Multidimensional Datasets

Even with Response State-Space Reduction Combinatorial Comparisons Can Yield Large, Multidimensional Datasets

For example, comparing 7 congestion control algorithms with respect to 48 responses under 32 conditions over three time periods with a mixture of competing TCP flows yields a multidimensional dataset with > 64,000 numbers

We used a range of multidimensional data analysis techniques, including:

1. Main-Effects Analysis [a, b, c] (skipping this)
2. Cluster Analysis [a, b]
3. Primary Principal Components Plots (skipping this)
4. Condition-Response Summaries [a, b, c]
5. Biplots (skipping this)
6. Colored Bar Graphs (skipping this)
7. Rank Analysis (skipping this)

2 (a) Cluster Analysis

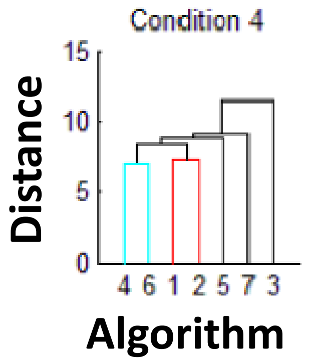
Compute normalized Euclidian distance between a pair of congestion-control algorithms in 45 dimensional space (45 responses used in this case)

$$Dist(Y_i, Y_j) \equiv \sqrt{\sum_{m=1}^{45} \frac{(Y_{im} - Y_{jm})^2}{(\sigma_m)^2}}$$

Compute linkage between clusters of congestion-control algorithms

$$D(r,s) \equiv \frac{1}{n_r n_s} \sum_{k=1}^{n_r} \sum_{l=1}^{n_s} Dist(Y_{k,r}, Y_{l,s})$$

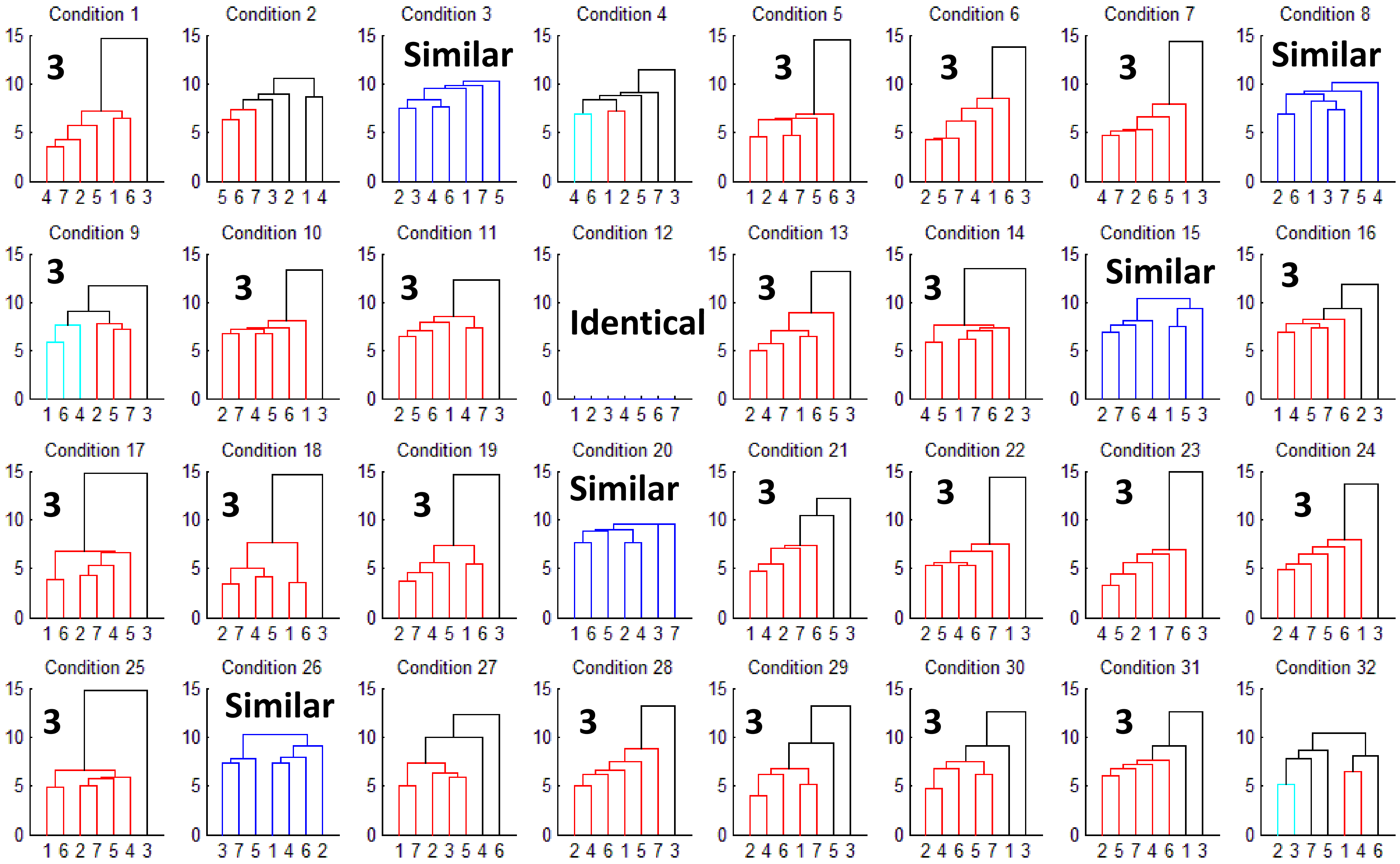
Plot as dendrogram and color when the linkage value for a pair of clusters falls below 70 % of the maximum linkage value



Note that congestion-control algorithms may only be compared under the same conditions (i.e., parameter combinations) because the conditions are very different from each other – this leads to the need to plot one dendrogram for each condition (32 here)

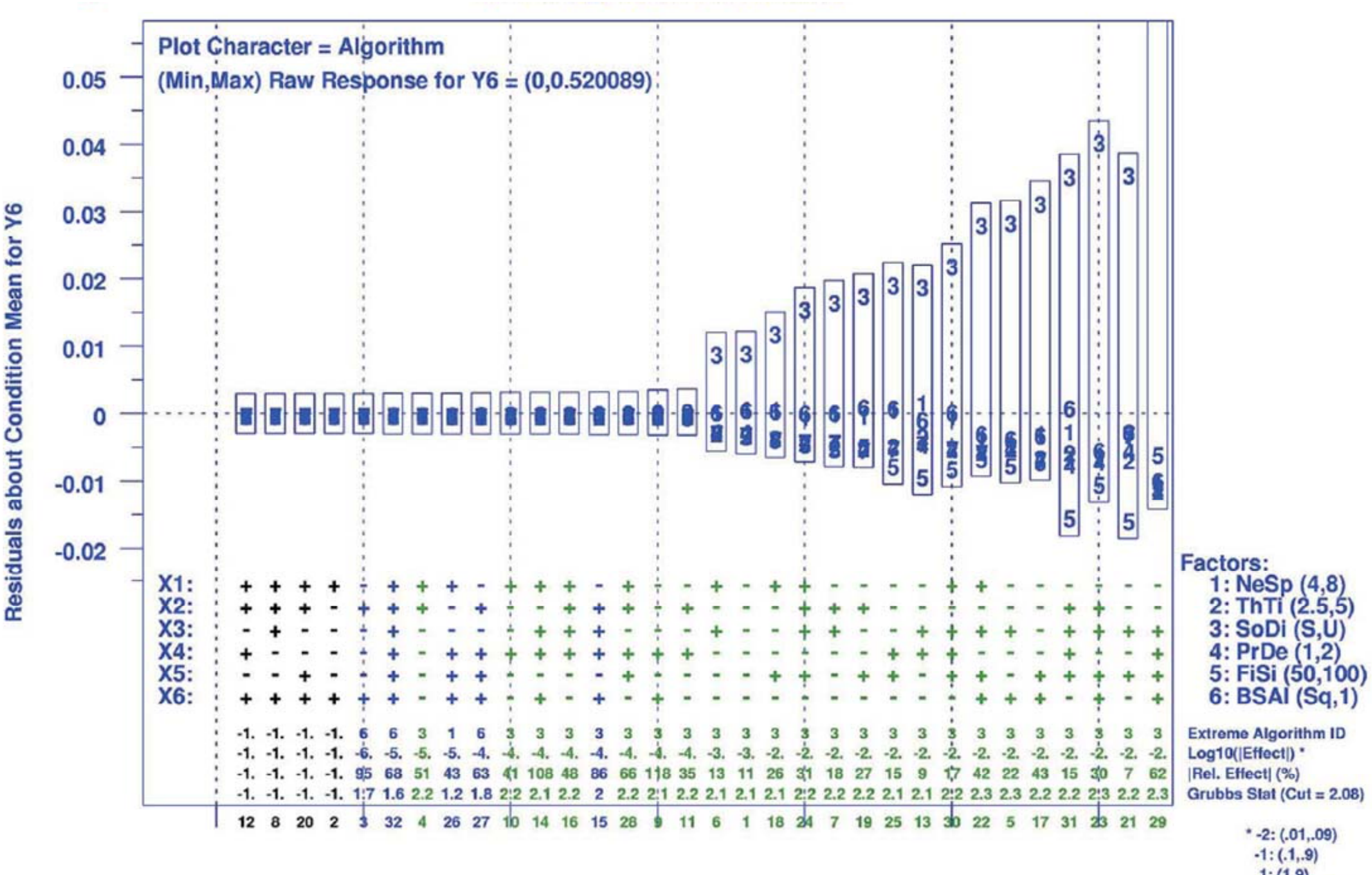
2 (b) Dendrograms for Each of 32 Conditions

Identical under no congestion; Similar under low congestion; algorithm 3 distinct under heavy congestion



4 (a) Condition-Response Summaries: Underlying Analyses

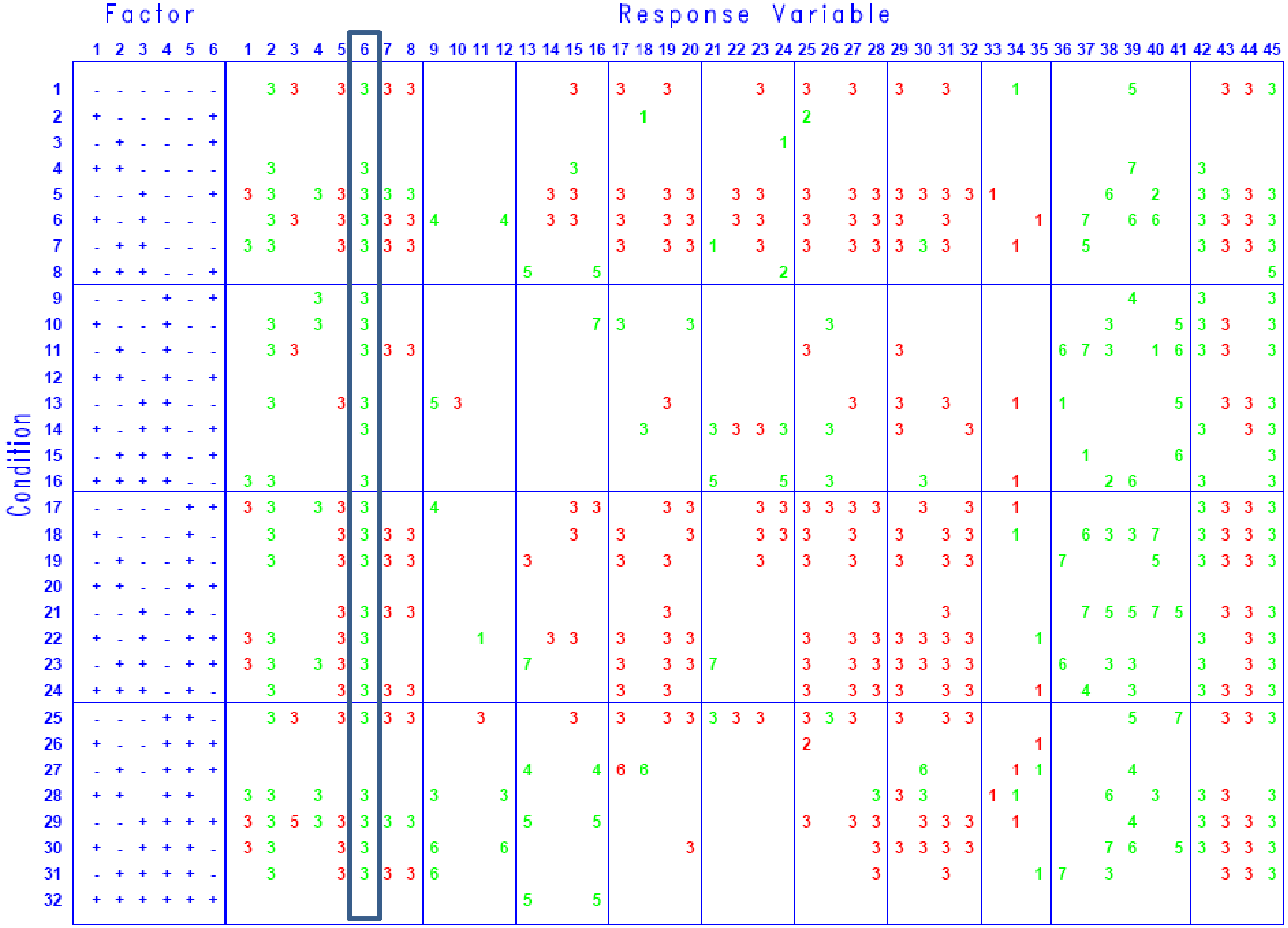
Plot shows that retransmission rate for congestion-control algorithm 3 is an outlier under conditions with high congestion, and the higher the level of congestion, the more the algorithm is an outlier.



Grubbs' test of outliers $G = \frac{\max(|Y_i - \text{mean}(Y)|)}{s}$

Sample plot analyzing the influence of condition and congestion control algorithm on the segment retransmission rate (in this experiment retransmission rate was designated as y6) – y axis gives residuals around the mean value for each condition and x axis gives conditions ordered by increasing range of residuals

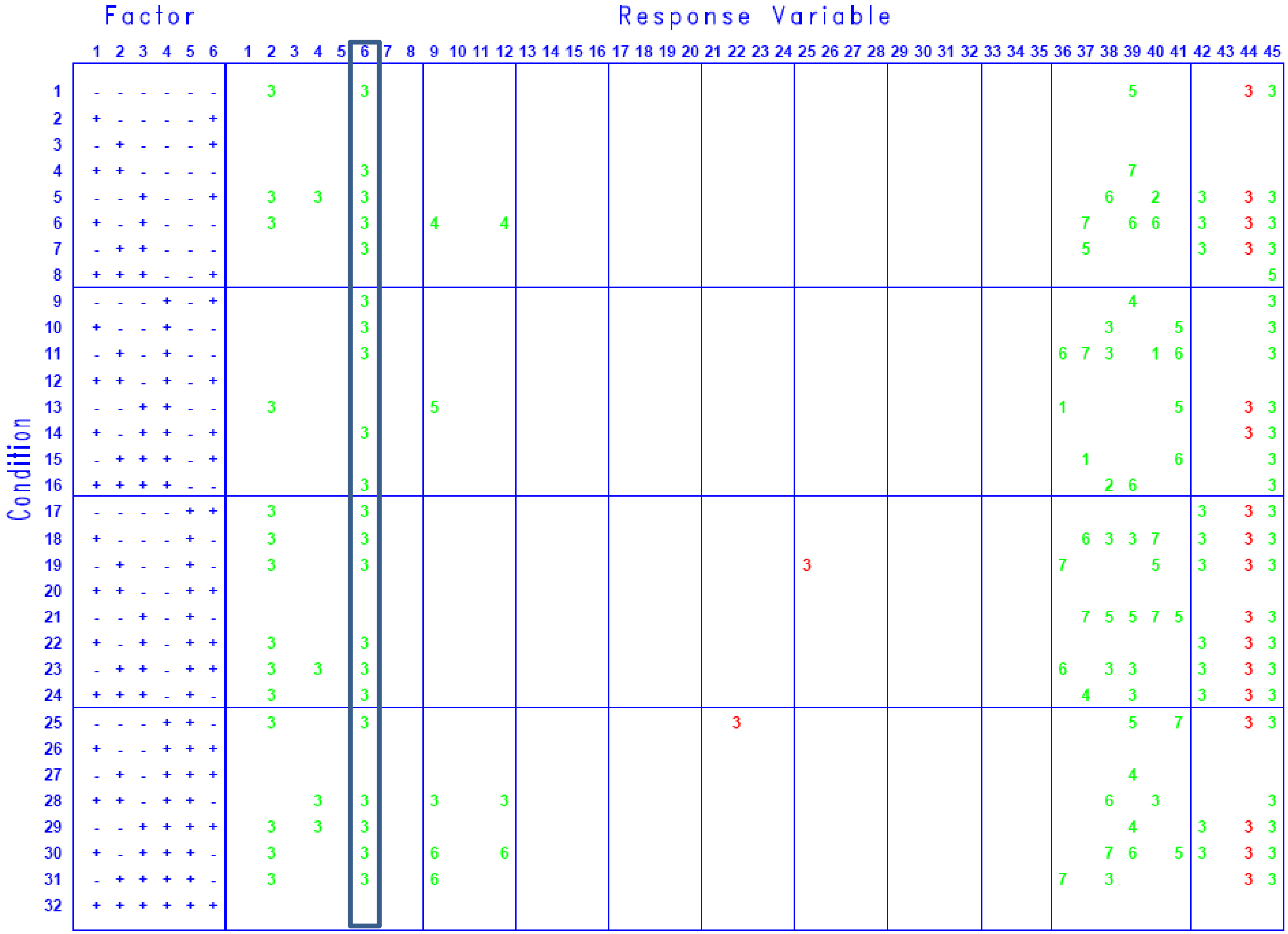
4 (b) Condition-Response Summaries: Rollup Outliers from Underlying Analyses



Condition	Factor						Response Variable																																																	
	1	2	3	4	5	6	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
1	-	-	-	-	-	-		3	3		3	3	3	3							3		3		3				3		3		3		3		3			1					5				3	3	3
2	+	-	-	-	-	+																		1							2																				
3	-	+	-	-	-	+																								1																					
4	+	+	-	-	-	-		3				3									3																								7			3			
5	-	-	+	-	-	+	3	3		3	3	3	3	3						3	3		3		3	3		3	3		3		3	3	3	3	3	3	1					6		2		3	3	3	3
6	+	-	+	-	-	-		3	3		3	3	3	3	4			4		3	3		3		3	3		3	3		3		3	3	3		3				1		7		6	6		3	3	3	3
7	-	+	+	-	-	-	3	3			3	3	3	3									3		3	3	1		3		3		3	3	3	3	3			1			5					3	3	3	3
8	+	+	+	-	-	+													5			5								2																					5
9	-	-	-	+	-	+				3		3																																	4			3			3
10	+	-	-	+	-	-		3		3		3										7	3			3						3												3			5	3	3		3
11	-	+	-	+	-	-		3	3			3	3	3																	3				3							6	7	3		1	6	3	3		3
12	+	+	-	+	-	+																																													
13	-	-	+	+	-	-		3			3	3			5	3									3								3		3		3			1		1					5		3	3	3
14	+	-	+	+	-	+						3												3			3	3	3	3		3			3			3										3		3	3
15	-	+	+	+	-	+																																					1				6				3
16	+	+	+	+	-	-	3	3				3															5			5		3				3				1				2	6			3			3
17	-	-	-	-	+	+	3	3		3	3	3			4						3	3			3	3			3	3	3	3	3	3		3		3		1								3	3	3	3
18	+	-	-	-	+	-		3			3	3	3	3							3		3			3			3	3	3		3		3		3	3		1			6	3	3	7		3	3	3	3
19	-	+	-	-	+	-		3			3	3	3	3					3				3		3				3		3		3		3		3	3				7				5		3	3	3	3
20	+	+	-	-	+	+																																													
21	-	-	+	-	+	-					3	3	3	3											3												3						7	5	5	7	5		3	3	3
22	+	-	+	-	+	+	3	3			3	3					1			3	3		3		3	3					3		3	3	3	3	3	3			1							3		3	3
23	-	+	+	-	+	+	3	3		3	3	3							7				3		3	3	7				3		3	3	3	3	3	3				6		3	3			3		3	3
24	+	+	+	-	+	-		3			3	3	3	3									3		3						3		3	3	3		3	3			1		4		3			3	3	3	3
25	-	-	-	+	+	-		3	3		3	3	3	3			3				3		3		3	3	3	3	3		3	3	3		3		3	3							5		7		3	3	3
26	+	-	-	+	+	+																									2										1										
27	-	+	-	+	+	+													4			4	6	6												6				1	1				4						
28	+	+	-	+	+	-	3	3		3		3			3			3																3	3	3			1	1				6		3		3	3		3
29	-	-	+	+	+	+	3	3	5	3	3	3	3	3					5			5									3		3	3		3	3	3		1					4			3	3	3	3
30	+	-	+	+	+	-	3	3			3	3			6			6								3								3	3	3	3	3						7	6		5	3	3	3	3
31	-	+	+	+	+	-		3			3	3	3	3	6																			3			3				1	7		3					3	3	3
32	+	+	+	+	+	+													5			5																													

Condition-response summary identifying any statistically significant outliers among congestion control algorithms for each of 45 responses measured over 32 different conditions in this particular experiment

4 (c) Condition-Response Summaries: Apply Filters



Condition	Factor						Response Variable																																													
	1	2	3	4	5	6	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
1	-	-	-	-	-	-		3				3																																	5					3	3
2	+	-	-	-	-	+																																													
3	-	+	-	-	-	+																																													
4	+	+	-	-	-	-						3																																	7						
5	-	-	+	-	-	+		3		3		3																																6		2		3		3	3
6	+	-	+	-	-	-		3				3			4			4																									7		6	6		3		3	3
7	-	+	+	-	-	-						3																															5					3		3	3
8	+	+	+	-	-	+																																													5
9	-	-	-	+	-	+						3																																	4						3
10	+	-	-	+	-	-						3																																3			5				3
11	-	+	-	+	-	-						3																														6	7	3		1	6				3
12	+	+	-	+	-	+																																													
13	-	-	+	+	-	-		3							5																											1					5			3	3
14	+	-	+	+	-	+						3																																						3	3
15	-	+	+	+	-	+																																					1				6				3
16	+	+	+	+	-	-						3																																2	6						3
17	-	-	-	-	+	+		3				3																																				3		3	3
18	+	-	-	-	+	-		3				3																															6	3	3	7		3		3	3
19	-	+	-	-	+	-		3				3																			3											7				5		3		3	3
20	+	+	-	-	+	+																																													
21	-	-	+	-	+	-																																					7	5	5	7	5			3	3
22	+	-	+	-	+	+		3				3																																				3		3	3
23	-	+	+	-	+	+		3		3		3																														6		3	3			3		3	3
24	+	+	+	-	+	-		3				3																															4		3			3		3	3
25	-	-	-	+	+	-		3				3																3																	5		7			3	3
26	+	-	-	+	+	+																																													
27	-	+	-	+	+	+																																							4						
28	+	+	-	+	+	-				3		3			3			3																										6		3					3
29	-	-	+	+	+	+		3		3		3																																	4			3		3	3
30	+	-	+	+	+	-		3				3			6			6																										7	6		5	3		3	3
31	-	+	+	+	+	-		3				3			6																											7		3						3	3
32	+	+	+	+	+	+																																													

Filtered condition-response summary identifying any congestion control algorithms that are statistically significant outliers and that exhibit at least a 10 % relative effect for each of 45 responses measures over 32 different conditions

Application to Predict Macroscopic Dynamics of Proposed TCP Replacements

Experiments Conducted

How do the proposed TCP replacements react to and recover from spatiotemporal congestion?

Experiment #1a	**Compared seven proposed TCP replacements (excluded FAST-AT) in a large (up to 278,000 sources), fast (up to 192 Gbps backbone) network; Web browsing users; 25 minutes simulated; three 5-minute time periods; large (> 1 billion segments) initial *sst*; all sources use the same congestion control algorithm**
Experiment #1b	**Same as #1a, except; smaller (up to 27,800 sources) and slower (up to 28.8 Gpbs backbone) network and low (100 segments) initial *sst*. (Added FAST-AT)**

How do the proposed TCP replacements improve flow throughputs and affect competing TCP flows?

Experiment #2a	**Compared eight proposed TCP replacements in a small (up to 26,085 sources), slow (up to 38.4 Gpbs backbone) network; Web browsing users and interspersed users who download software and movies; 60 minutes simulated; large (> 1 billion segments) initial *sst*; some sources use standard TCP congestion control procedures and some sources use one of the proposed TCP replacements**
Experiment #2b	**Same as #2a except for low (100 segments) initial *sst*.**
Experiment #2c	**Same as #2a except for a larger (up to 261,792 sources) and faster (up to 384 Gbps backbone) network.**

What distinguishes among proposed TCP replacements? Will they be useful? Will they be safe?

1. ***Increase rate*: How quickly can the maximum transmission rate be achieved?**

2. ***Loss/Recovery processing*:**
   a. **How much does the protocol reduce transmission rate upon a loss?**
   b. **How quickly does the protocol increase transmission rate after a reduction?**

3. ***TCP Fairness*: How well do standard TCP flows do when competing with proposed TCP replacements?**

4. ***Utility bounds*: Under what circumstances can proposed TCP replacements provide improved user throughputs?**

5. ***Safety*: Will widespread deployment of proposed TCP replacements induce undesirable macroscopic dynamics into the Internet?**

Increase Rate

- Assuming low congestion, setting of initial *sst* is a key factor
  - High initial *sst* – all algorithms (standard TCP included) achieved maximum transmission rate with the same (exponential) quickness
  - Low initial *sst* – proposed TCP replacements achieved maximum transmission rate more quickly than (linear) increase of standard TCP

- Under heavy congestion, setting of initial *sst* matters little because initial slow start terminates upon first packet loss and a flow enters congestion avoidance, where loss/recovery processing determines throughput

- On real TCP flows receivers may convey a window (*rwnd*) that can restrict throughput because sources pace transmission based on **min**(*cwnd*, *rwnd*). Typically, *rwnd* < *cwnd*. In our studies, we assume an infinite *rwnd* in order to compare effects of congestion control algorithms. Throughput on many TCP flows in a real network might be constrained by *rwnd*, so that proposed TCP replacements would provide little advantage over standard TCP. In fact, even TCP congestion control does not have much influence when *rwnd* < *cwnd.*

Loss/Recovery Processing

- One group of algorithms (Scalable TCP, BIC[1] and HSTCP) reduce transmission rate less than standard TCP after a packet loss
  - Unfair to TCP flows and to new flows using alternate algorithms
- Another group of algorithms (CTCP, FAST and FAST-AT) reduce transmission rate by ½ following a loss (HTCP is a hybrid with reduction between 20 and 50 %)
  - These algorithms seek to obtain higher throughput by increasing transmission rate more quickly than standard TCP (the rate of increase varies with the algorithm)
  - HTCP reverts to TCP congestion avoidance for 1 s after each loss, which can lead to lower throughputs than other alternate algorithms
- Under extreme spatiotemporal congestion, most proposed TCP replacements have a low-window threshold and revert to standard TCP congestion avoidance procedures (giving no advantage to alternate procedures)
  - FAST and FAST-AT do not use TCP congestion avoidance under any conditions, which can lead to oscillatory behavior and increased loss rates

[1] Note that on repeated losses occurring close in time, BIC can reduce *cwnd* substantially more than standard TCP – thus, on paths with very severe congestion TCP can provide higher throughput than BIC

TCP Fairness

- All proposed TCP replacements take steps to provide improved throughput over TCP – thus comparing fairness must consider relative performance of TCP flows when competing with flows using each proposed replacement

- We found CTCP, HTCP and FAST-AT to be most fair to TCP flows
  - Under low initial *sst* FAST-AT is more unfair because of its quick increase in rate
  - Injecting more FAST-AT packets induced more losses in TCP flows, which could recover only linearly

- We found Scalable TCP, BIC and FAST to be most unfair to TCP flows
  - Established Scalable and BIC flows (on large files) tended to maintain higher transmission rates than TCP flows after losses, while FAST recovered more quickly, and these proposed TCP replacements induced more losses in TCP flows

- HSTCP appeared moderately fair to TCP flows, especially under conditions of lower congestion and under low initial *sst* – HSTCP appeared unfair under conditions of heavy congestion

- We found that Scalable TCP, BIC and HSTCP are also unfair to competing flows that are newly arriving

Utility Bounds

- We found that proposed TCP replacements could provide increased utility (throughput) for users – however, this utility would arise only under a specific combination of circumstances
  - Flow's *rwnd* must not be constraining flow transmission rate
  - Flow's initial *sst* must be relatively low
  - Flow must be transferring a large file
  - Flow's packets must be transiting a relatively uncongested path (i.e., experiencing only sporadic losses) or else users must be willing to tolerate marked unfairness in trade for increased throughput

- How likely is this combination of circumstances on a given Internet flow?
  - Certainly possible to engineer a network, or segments of a network, to provide specific users with improved throughput compared with TCP
  - We suspect a rather low probability for such circumstances to arise generally in the Internet

- We conclude that proposed TCP replacements can provide improved user throughput – however, most users seem unlikely to benefit very often

Safety

- We can answer this only in part – additional cautionary findings may be possible
  - We simulated either homogeneous networks where all flows used one congestion control algorithm or mixes of TCP flows competing with flows using one proposed TCP replacement at a time
  - The real Internet could contain a mix of many different types of congestion algorithm

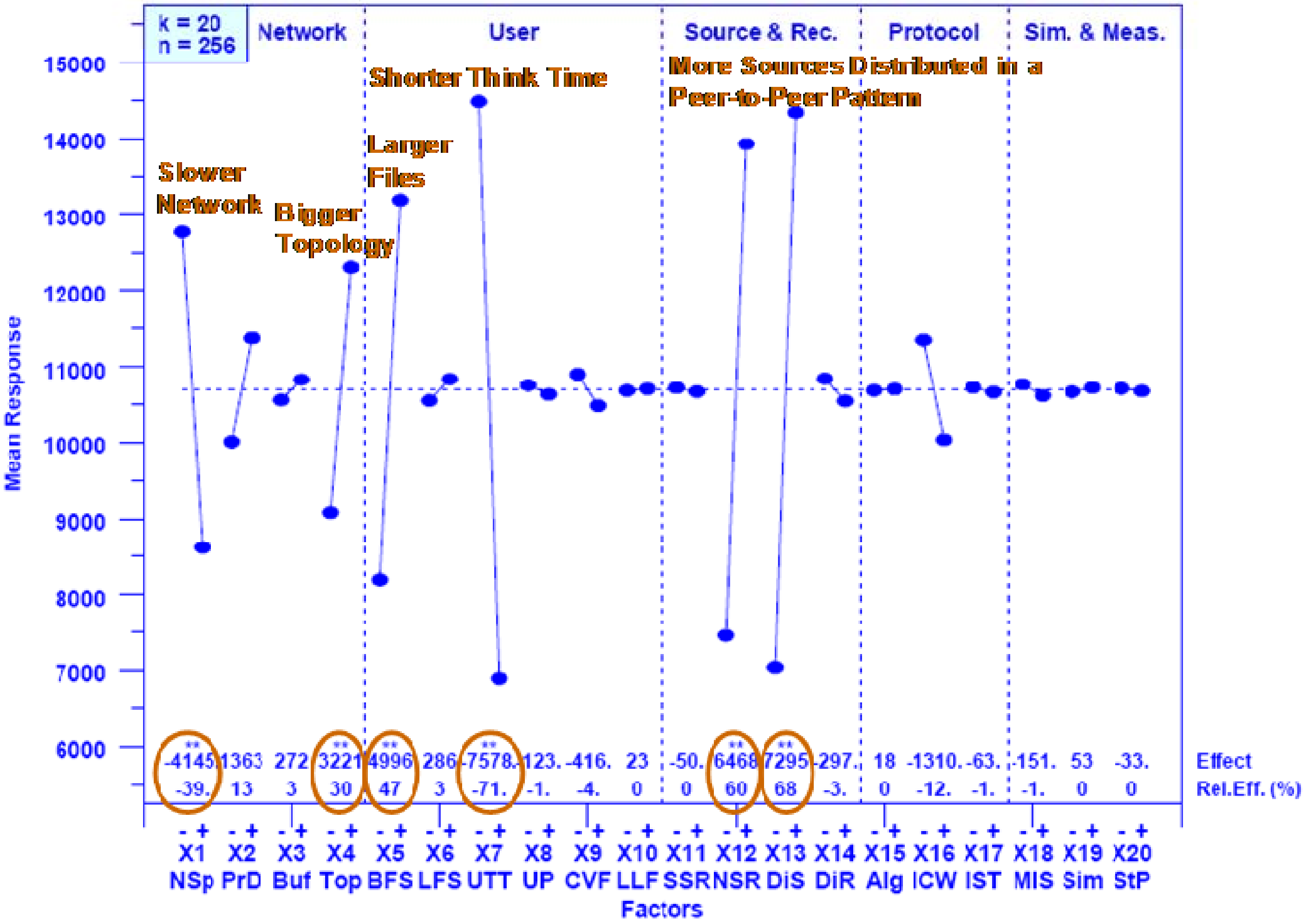
- For most algorithms we studied, under most conditions, we found little significant change in macroscopic network characteristics

- FAST and FAST-AT are exceptions to this general finding
  - Under high spatiotemporal congestion, where there were insufficient buffers to support flows transiting specific routers, FAST and FAST-AT entered an oscillatory behavior where the flow *cwnd* increased and decreased rapidly with large amplitude
  - Under such conditions the network showed increased loss and retransmission rates, a higher number of flows pending in the connecting state and a lower number of flows completed over time

- We recommend the need for additional study of FAST and FAST-AT prior to widespread deployment and use on the Internet

Omitted Slides

1(a) Sample Main-Effects Analysis on One Response



Main effects plot showing absolute and relative influence of each *MesoNet* parameter (x axis) on mean (y axis) number of sources transmitting (y1) and identifying six statistically significant parameters – network speed (X1), topology (X4), file size (X5), think time (X7) and number (X12) and distribution (X13) of sources

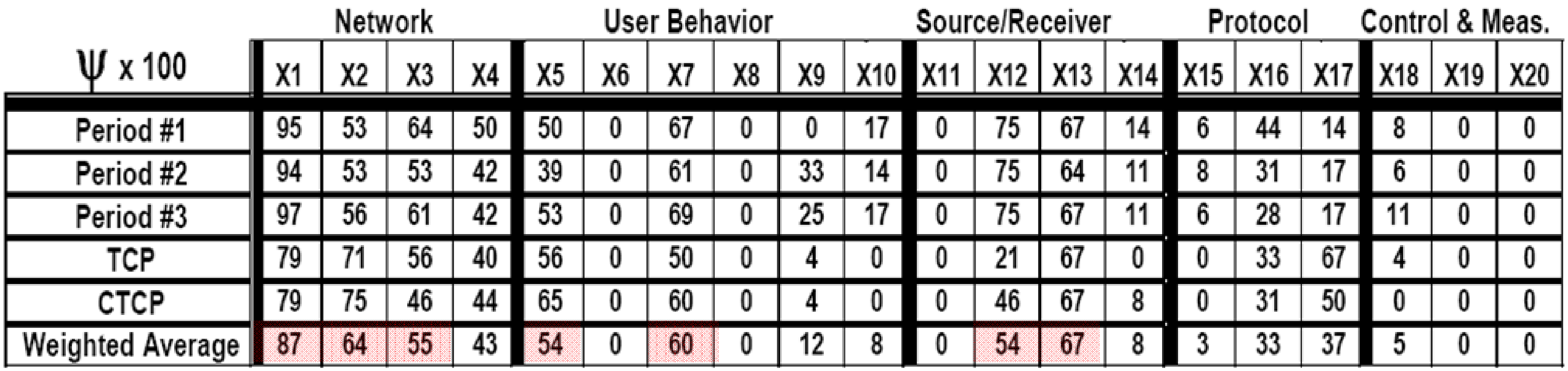
1 (b) Summary of Main-Effects Analyses Across 18 Responses

Significance of influence of 20 MesoNet parameters (columns) on 18 macroscopic responses (rows) during 2nd time period: blue cells indicate significance *p* < 0.01 and orange cells indicate significance *p* < 0.05, where a – or + in highlighted cells indicates the parameter setting that causes an increase in the corresponding response

Metric Class		Network				User Behavior						Source/Receiver				Protocol			Sim. Control & Meas.		
	Y#	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	X18	X19	X20
Flows	Y1	-**		+**	+*			-**					+**	+**			+**				
	Y2	-**			+**	+**		-**					+**	+**							
	Y3	+**	+**	+**		-**		+**					-**	-**				+**			
	Y4	-**	-**	-**		+**		-**					+**	+**		-*					
	Y5	+**		+**				+**					-**	-**		+**	+*	-**			
Congestion	Y6	-**	-**	-**		+**		-**					+**	+**			+**				
	Y7	+*																			
	Y8	-**	-**	-**		+**							+**	+**			+**				
Delay	Y9	-**	+**	+**									+*	+*							
	Y10	-**	+**	+**				-*					+**	+**							
Aggregate TP	Y11	+**	-*		+**	-**		-**					+**	+**							
	Y12	+**		+**	+**	+**		-**		-**			+**								
Long-Lived Flow TP	Y13	+**		+*	+*					+**	-**										
	Y14	+*			+**					+**	-**										
	Y15	+**								+**	-*				-*						
Other Flow TP	Y16	+**	-**		-**			+*		+**			-**		-*			+**	-*		
	Y17	+**	-**		+*			+**		+**			-**	+**	-**		+**		-*		
	Y18	+**	-**		-**			+**					-**	-**			+**				

1 (c) Summary of Parameter Influence on Responses for Three Time Periods and Two Congestion Control Algorithms

$$\Psi = (|\{y \mid p < 0.01\}| + \tfrac{1}{2}\,|\{y \mid p < 0.05\}|) \,/\, |\{y\}|$$

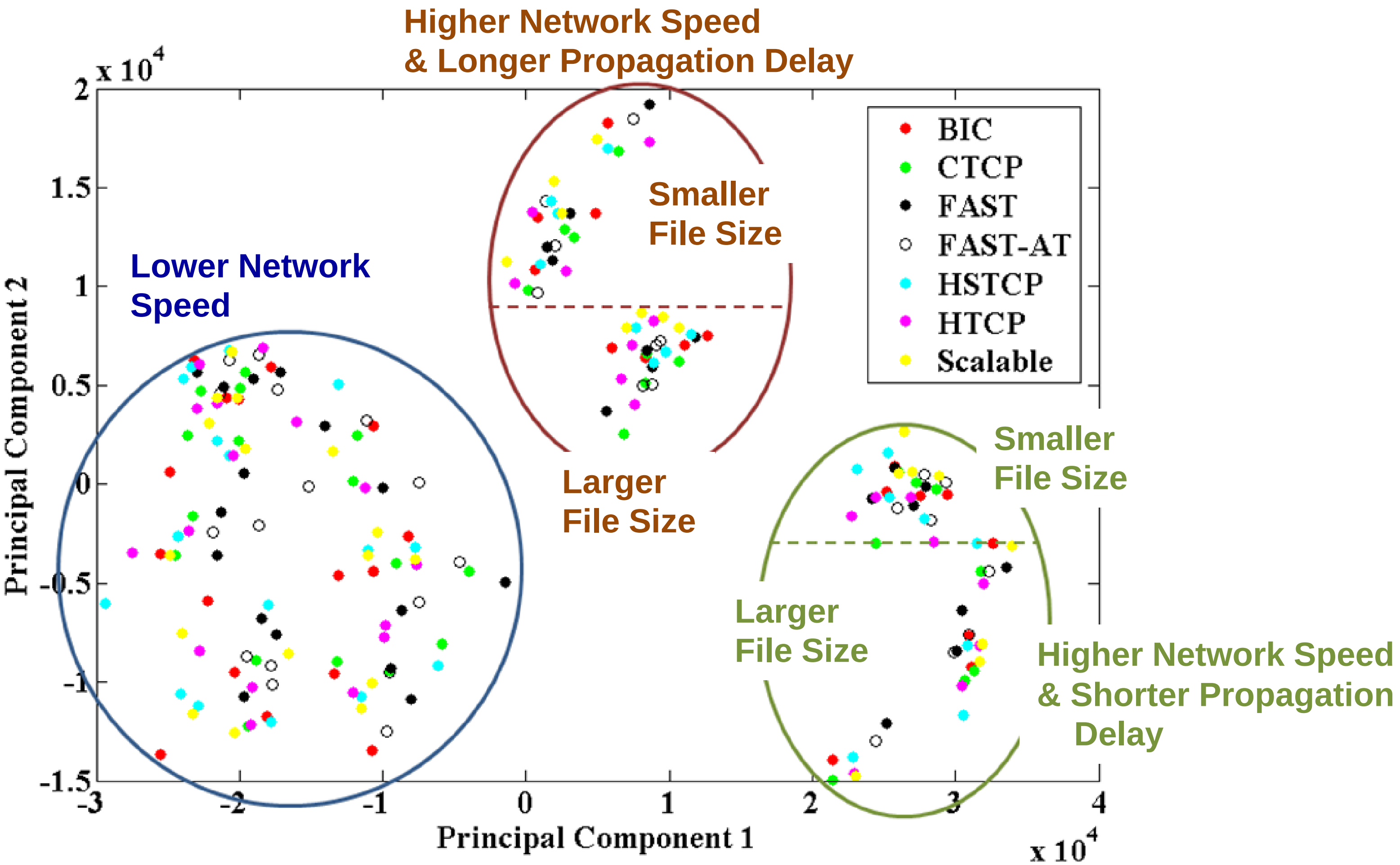


	Network				User Behavior						Source/Receiver				Protocol			Control & Meas.		
Ψ x 100	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	X18	X19	X20
Period #1	95	53	64	50	50	0	67	0	0	17	0	75	67	14	6	44	14	8	0	0
Period #2	94	53	53	42	39	0	61	0	33	14	0	75	64	11	8	31	17	6	0	0
Period #3	97	56	61	42	53	0	69	0	25	17	0	75	67	11	6	28	17	11	0	0
TCP	79	71	56	40	56	0	50	0	4	0	0	21	67	0	0	33	67	4	0	0
CTCP	79	75	46	44	65	0	60	0	4	0	0	46	67	8	0	31	50	0	0	0
Weighted Average	87	64	55	43	54	0	60	0	12	8	0	54	67	8	3	33	37	5	0	0

Parameters with largest influence: network speed (x1), distribution of sources (x13), propagation delay (x2), user think time (x7), buffer size (x3) and file size (x5) and number of sources (x12)

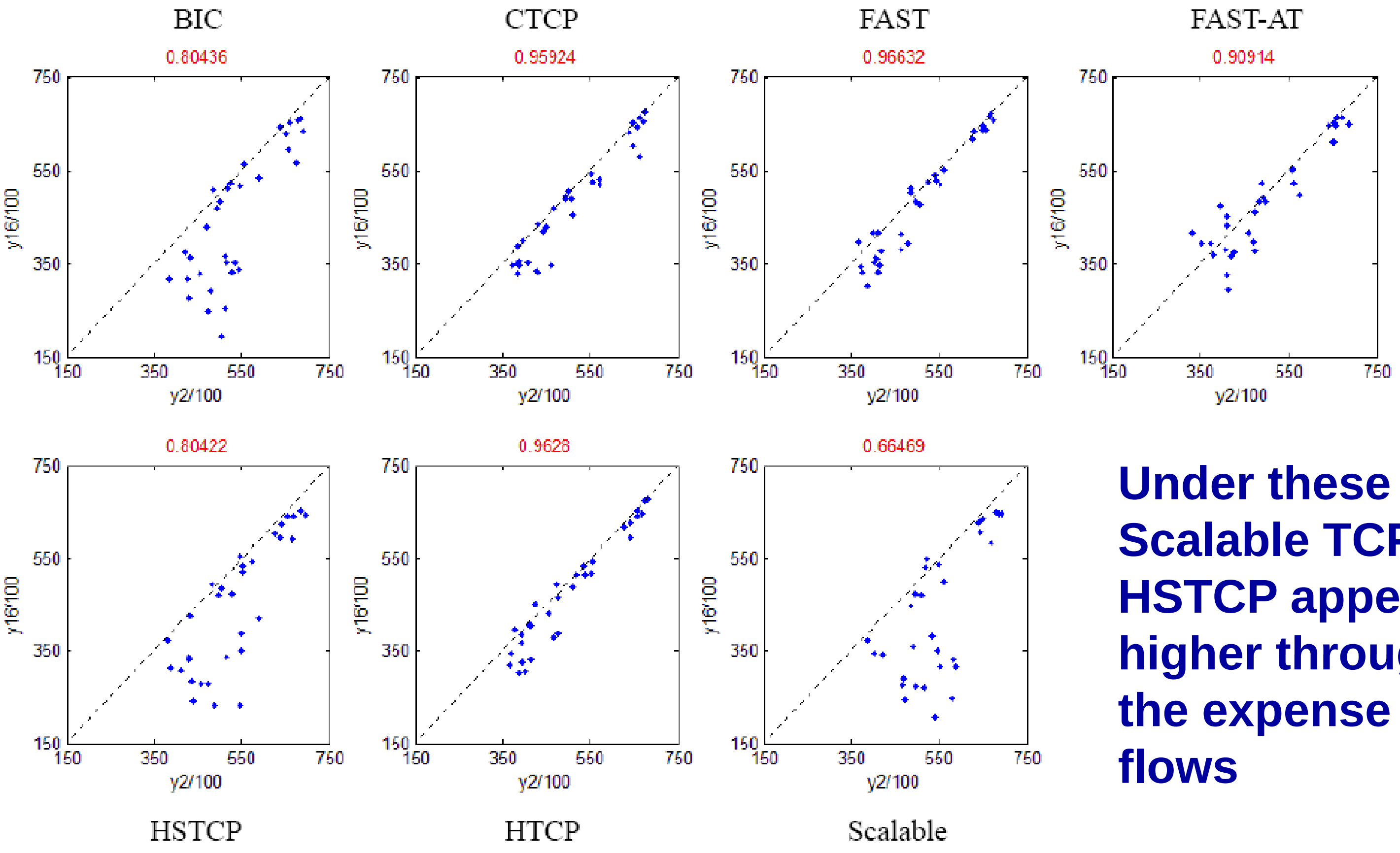
3 Primary Principal Components Plots

Here, congestion-control algorithm does not make a difference



PC1 (x axis) vs. PC2 (y axis) from a multivariate dataset containing average throughput achieved on network connections using seven different congestion control algorithms under 32 different parameter combinations. Circles identify groupings of points and lines within circles distinguish additional subgroups.

5 Biplots

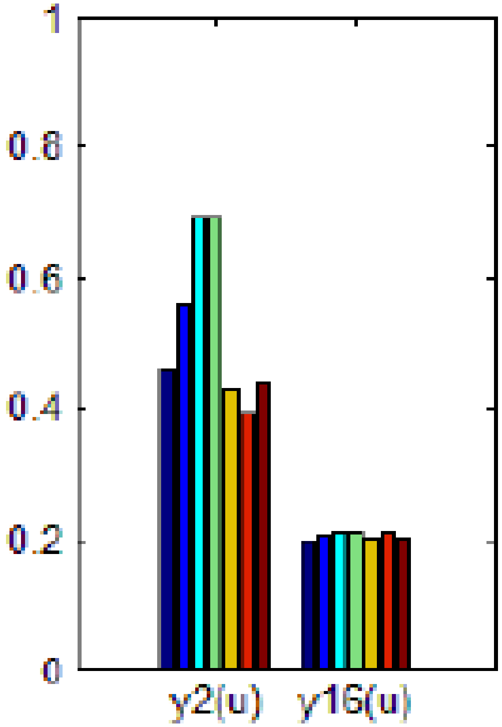


Under these conditions, Scalable TCP, BIC and HSTCP appear to give higher throughputs at the expense of TCP flows

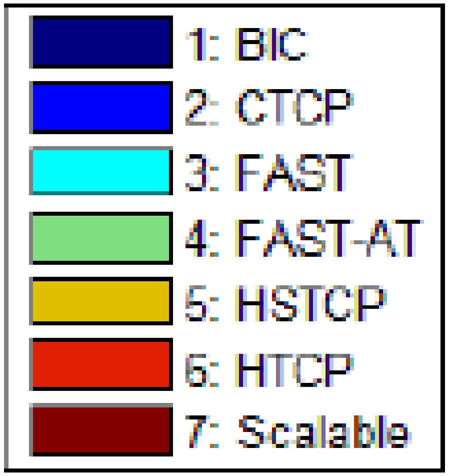
Seven biplots, where each biplot compares throughput (in units of 100 segments per second) for large flows transferred over very fast paths with fast interface speeds using a proposed TCP replacement (x axis) with throughput on identical competing flows using standard TCP – here a high initial slow-start threshold is used

6 Colored Bar Graphs

FAST & FAST-AT provide best throughput, while CTCP provides second best



Legend

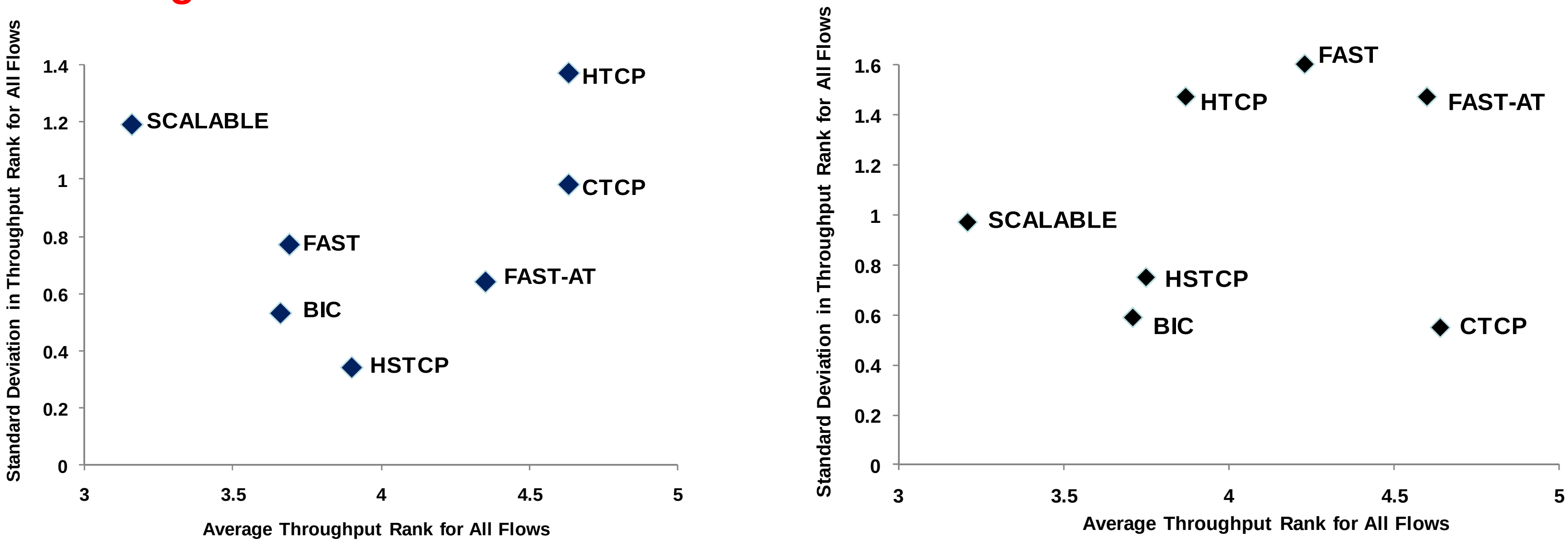


Note that TCP flows achieve only 20 % of available throughput, as was measured empirically by other researchers

Fraction (y axis) of maximum available throughput achieved when transferring movies over uncongested paths with a maximum rate of 1 Gbps – each of the leftmost seven bars – y2(u) – represents flows using one of the proposed replacements for TCP (see legend), while each of the rightmost seven bars – y16(u) – represents flows using standard TCP and competing with flows using one of the proposed TCP replacements (see legend) – here a low initial slow-start threshold is used.

7 Rank Analyses

CTCP leads high ranking throughputs under a wide range of conditions



high initial slow-start threshold

low initial slow-start threshold

Average throughput rank (x axis) vs. standard deviation in throughput rank (y axis) for flows using each proposed TCP replacement and for competing flows using standard TCP