Evaluation of Advanced TCP stacks on Fast Long-Distance production Networks

Prepared by Les Cottrell & Hadrien Bullot, Richard Hughes-Jones
EPFL, SLAC and Manchester University for the
Protocols for Fast Long Distance Networks, ANL
February, 2004
www.slac.stanford.edu/grp/scs/net/talk03/pfld-feb04d.ppt

Partially funded by DOE/MICS Field Work Proposal
on Internet End-to-end Performance Monitoring
(IEPM), also supported by IUPAP
and UK e-Science via PPARC

1

Project goals

- Test new advanced TCP stacks, see how they perform on short and long-distance **real production** WAN links
- Compare & contrast: ease of configuration, throughput, convergence, fairness, stability etc.
- For different RTTs, windows, txqueuelen
- Recommend "optimum" stacks for data intensive science:
  (BaBar) transfers using bbftp, bbcp, GridFTP
- Validate simulator & emulator findings & provide feedback

2

Protocol selection

- Focus on TCP only
  - No Rate based transport protocols (e.g. SABUL, UDT, RBUDP) at the moment
  - No iSCSI or FC over IP
- Sender mods only, HENP model is few big senders, lots of smaller receivers
  - Simplifies deployment, only a few hosts at a few sending sites
  - No DRS
- Runs made on production networks so:
  - No router mods (XCP/ECN), no jumbos,

3

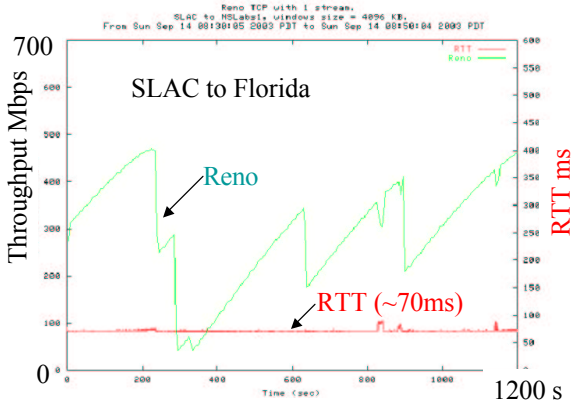
Protocols Evaluated

- Linux 2.4 New Reno with SACK: single (Reno) and parallel streams (P-TCP)
- Scalable TCP (S-TCP)
- Fast TCP
- HighSpeed TCP (HS-TCP)
- HighSpeed TCP Low Priority (HSTCP-LP)
- Binary Increase Control TCP (Bic-TCP)
- Hamilton TCP (H-TCP)

4

Reno single stream

- Low performance on fast long distance paths
  - AIMD (add *a=1* pkt to *cwnd* / RTT, decrease *cwnd* by factor *b=0.5* in congestion)

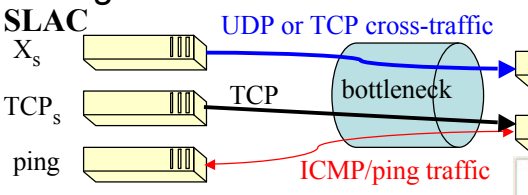


5

Measurements

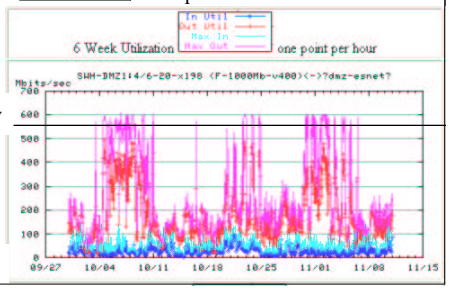
- 20 minute tests, long enough to see stable patterns
- Iperf reports incremental and cumulative throughputs at 5 second intervals
- Ping interval about 100ms

Ping traffic goes to TCP_r when also running cross-traffic
Otherwise goes to X_r



Over a thousand 20 minute measurements or 300 hours

600 Mbps capacity



Utilization of SLAC ESnet link Sep-Nov '03

Networks

- 3 main network paths
  - Short distance:
    SLAC-Caltech (RTT~10ms)
  - Middle distance:
    U. Florida (UFl) & DataTAG Chicago(RTT~70ms)
  - Long distance:
    CERN & University of Manchester (RTT ~ 170ms)
  - Tests during nights and weekends to avoid unacceptable impacts on production traffic

7

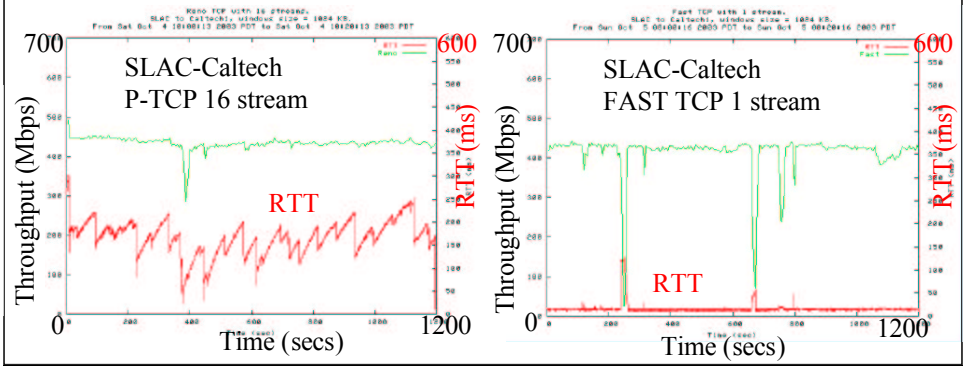
Windows

- Set large maximum windows (typically 32MB) on all hosts
- Used 3 different windows with iperf:
  - Small window size, factor 2-4 below optimal
  - Roughly optimal window size (~BDP)
  - Oversized window

8

RTT

- Only P-TCP appears to dramatically affect the RTT
  - E.g. increases by RTT by 200ms (factor 20 for short distances)
  - Implication: P-TCP would impact apps. like Voice/IP



txqueuelen

- Regulates the size of the queue between the IP layer and the Ethernet layer
- May increase the throughput if we find optimal values
- But may increase duplicate ACKs (Y. T Li)

Txqueuelen vs TCP for UFl 4MB window	Reno 16	S-TCP	Fast	HS	Bic	H TCP	HS LP	avg
tqueuelen=100	428	301	340	431	387	348	383	374
tqueuelen=2000	434	437	400	224	396	310	380	368.71
tqueuelen=10000	429	281	385	243	407	337	386	352.57
Avg	430.33	339.67	375	299.33	396.67	331.67	383	

- All stacks except S-TCP use txqueuelen=100 as default
- S-TCP uses txqueuelen=2000 by default
- Tests showed these were reasonable choices

10

Throughput (Mbps)

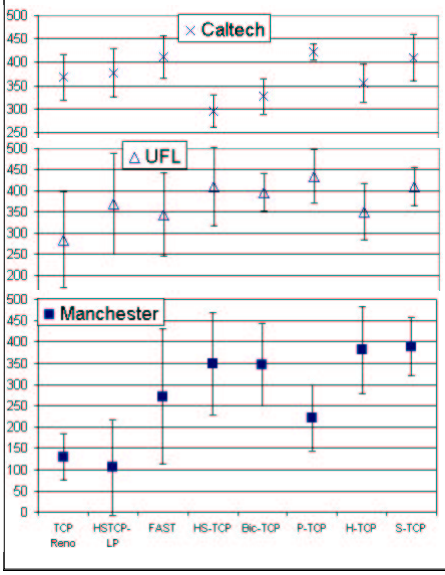
Windows too small (worse for longer distance)

Throughput SLAC to Remote	Reno 16	Sc	Bic	Fast	HS LP	H	HS	Reno 1	Avg
Caltech 256 KB	395	226	238	233	236	233	225	239	253
UFl 1 MB	451	110	133	136	141	140	136	129	172
Caltech 512 KB	413	377	372	408	374	339	307	362	369
UFl 4 MB	428	437	387	340	383	348	431	294	381
Caltech 1 MB	434	429	382	413	381	374	284	374	384
UFl 8 MB	442	383	404	348	357	351	387	278	369
Average	427	327	319	313	312	298	295	279	321
Rank	1	2	2	2	2	4	4	4	

Poor performance
Reasonable performance
Better performance
Best performance

Reno with 1 stream has problems on Medium distance link (70ms) Window size ?

Throughput



Avg throughput for optimal & large window sizes from SLAC to CalTech, UFl & Manchester

Stack more important for long RTTs

Single stream Reno & HSTCP-LP poorer on large RTTs

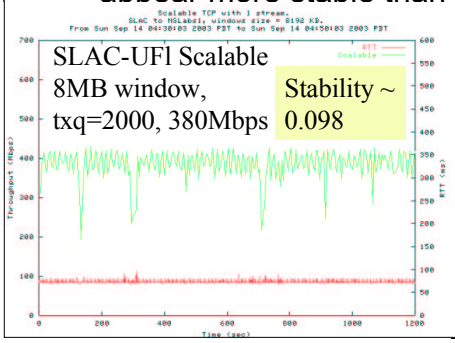
12

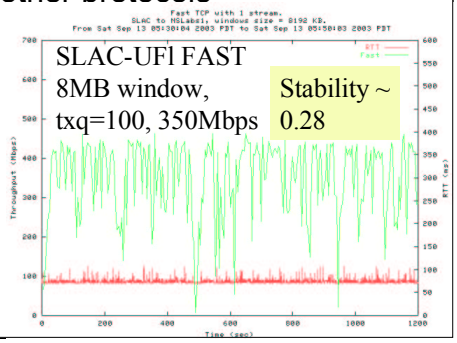
Stability

- Definition: standard deviation normalized by the average throughput

Stability for optimal txq window & stack for SLAC to UFl	Reno TCP	Reno TCP 1	S-TCP	Fast TCP	HS-TCP	Bic-TCP	H TCP	HSTCP
1 MB	0.2065	0.0713	0.0988	0.0897	0.1100	0.0955	0.0985	0.1281
4 MB	0.3754	0.1660	0.1167	0.2985	0.2115	0.1335	0.2181	0.3132
8 MB	0.4149	0.1179	0.0986	0.2772	0.2471	0.0850	0.1595	0.3333

  - At short RTT (10ms) stability is usually good (<=12%)
  - At medium RTT (70ms) P-TCP, Scalable & Bic-TCP appear more stable than the other protocols





Sinusoidal UDP

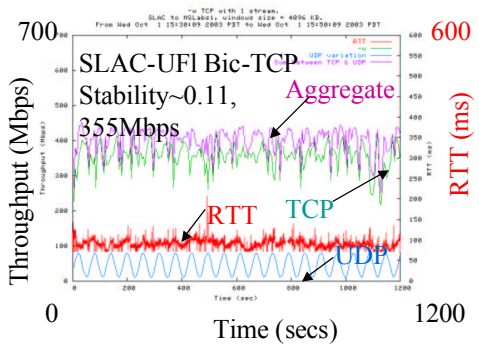
- UDP does not back off in face of congestion, it has a "stiff" behavior
- We modified iperf to allow it to create UDP traffic with a sinusoidal time behavior, following an idea from Tom Hacker
  - See how TCP responds to varying cross-traffic
- Used 2 periods of 30 and 60 seconds and amplitude varying from 20 to 80 Mbps
- Sent from 2nd sending host to 2nd receiving host while sending TCP from 1st sending host to 1st receiving host
- As long as the window size was large enough all protocols converged quickly and maintain a roughly constant aggregate throughput
- Especially for P-TCP & Bic-TCP

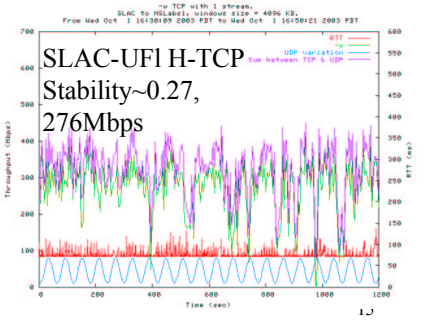
14

TCP Stability against UDP

- Stability better at short distances
- P-TCP & Bic more stable

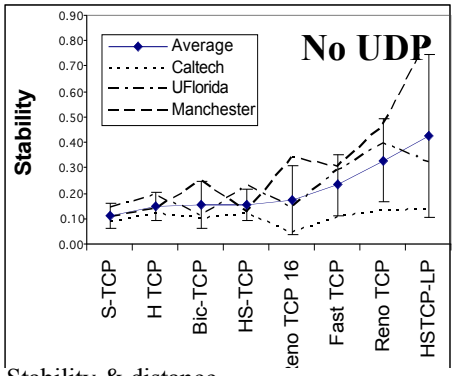
Stability to UFl vs window & UDP freq.	Reno 16	Scal	Fast	HS	Bic	H	HS LP
UDP 60s + 1 MB	0.13	0.13	0.09	0.10	0.10	0.11	0.17
UDP 60s + 4 MB	0.12	0.26	0.35	0.18	0.11	0.27	0.25
UDP 60s + 8 MB	0.13	0.14	0.36	0.20	0.14	0.14	0.23
UDP 30s + 1 MB	0.12	0.11	0.07	0.11	0.09	0.21	0.17
UDP 30s + 4 MB	0.16	0.38	0.29	0.21	0.12	0.27	0.30

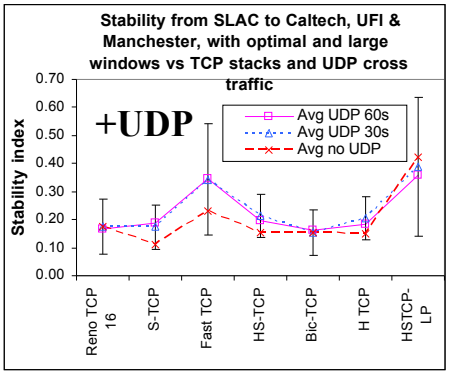




Stability

Stability from SLAC to Caltech, U Florida & Manchester





Stability & distance
Short RTT is more stable

Little difference between periodicity of UDP (30 & 60 secs)
HSTCP-LP & FAST have larger stability indices (less stability)

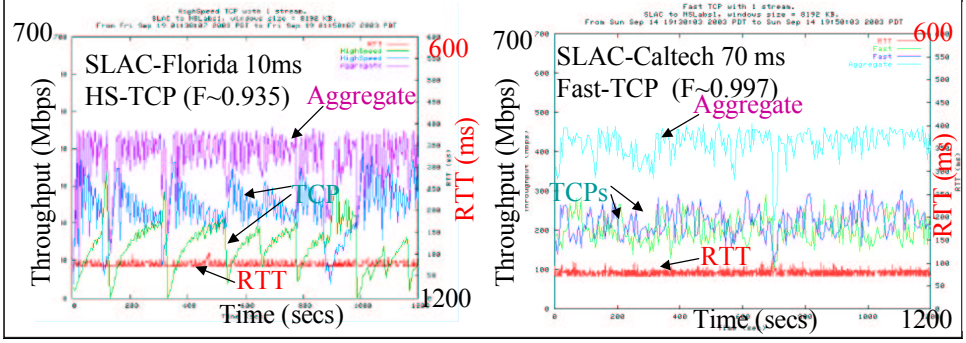
16

Cross TCP Traffic

- Important to understand how fair a protocol is
- For one protocol competing against the same protocol (**intra-protocol**) we define the fairness for a single bottleneck as:

$$F = \frac{(\sum_{i=1}^{n} \overline{x}_i)^2}{n \sum_{i=1}^{n} \overline{x}_i^2}$$

- All protocols have good intra-protocol Fairness (F>0.98)
- Except HS-TCP (F<0.94) when the window size > optimal



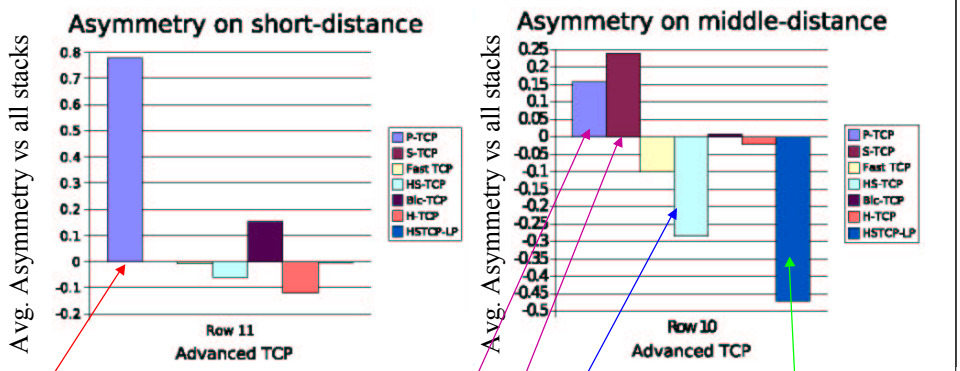
Fairness (F)

Avg Fairness from SLAC to UFl. Cross-traffic=> Source	Reno TCP 1 6	S-TCP	Fast TCP	HS-TCP	Bic-TCP	H TCP	HSTCP-LP	Avg
P-TCP	1.00	0.92	0.89	0.90	0.95	0.94	0.69	0.9
S-TCP	0.92	1.00	0.87	0.90	0.91	0.92	0.73	0.9
Fast TCP	0.89	0.87	1.00	0.92	0.93	0.99	0.73	0.9
HS-TCP	0.90	0.90	0.92	0.97	0.95	0.94	0.95	0.9
Bic-TCP	0.95	0.91	0.93	0.95	1.00	0.99	0.93	0.9
H-TCP	0.94	0.92	0.99	0.94	0.99	1.00	0.95	0.9
HSTCP-LP	0.69	0.73	0.73	0.95	0.93	0.95	1.00	0.8

- Most have good intra-protocol fairness (diagonal elements), except HS-TCP
- Worse for larger RTT (Caltech F~0.999+-0.004, U Florida F~0.995+-0.14, Manchester F~0.95+-0.05)
- Inter protocol Bic & H appear more fair against others
- Worst fairness are: P-TCP, S-TCP, Fast, HSTCP-LP (backoff early)
- But cannot tell who is aggressive and who is timid

Inter protocol Fairness

- For inter-protocol fairness we introduce the asymmetry between the two throughputs: $A = \frac{\overline{x}_1 - \overline{x}_2}{\overline{x}_1 + \overline{x}_2}$
  - Where x_1 and x_2 are the throughput averages of TCP stack 1 competing with TCP stack 2 +ve TCP1 agressive



Reno 16 v. aggressive at short RTT, Reno 16 & Scalable aggressive at medium distance on medium RTT: HS-TCP timid HSTCP-LP very timid

Inter Fairness – UFI (A)

$A=(x_m-x_c)/(x_m+x_c)$

Aggressive

Fair

Timid

Cross traffic=> Major source	Reno 16	Sca	Fast	HS	Bic	H	HS LP	Avg
Reno 16 + 4 MB	0.00	0.38	0.26	0.45	0.05	0.12	0.66	0.27
Reno 16 + 8 MB	0.00	-0.16	0.25	0.35	0.10	0.09	0.61	0.18
S-TCP + 4 MB	-0.38	0.00	0.33	0.07	0.19	0.12	0.65	0.14
S-TCP + 8 MB	0.16	0.00	0.63	0.65	0.56	0.54	0.70	0.46
Fast TCP + 4 MB	-0.26	-0.33	0.00	0.26	-0.29	0.11	0.68	0.03
Fast TCP + 8 MB	-0.25	-0.63	0.00	0.48	-0.38	0.11	0.68	0.00
HS-TCP + 4 MB	-0.45	-0.07	-0.26	0.00	-0.25	-0.17	0.37	-0.12
HS-TCP + 8 MB	-0.35	-0.65	-0.48	0.00	-0.33	-0.41	0.13	-0.30
Bic-TCP + 4 MB	-0.05	-0.19	0.29	0.25	0.00	-0.10	0.29	0.07
Bic-TCP + 8 MB	-0.10	-0.56	0.38	0.33	0.00	-0.15	0.31	0.03
H TCP + 4 MB	-0.12	-0.12	-0.11	0.17	0.10	0.00	0.19	0.01
H TCP + 8 MB	-0.09	-0.54	-0.11	0.41	0.15	0.00	0.37	0.03
Average	-0.16	-0.24	0.10	0.28	-0.01	0.02	0.47	0.07

Diagonal = 0 by definition
Symmetric off diagonal
Down how does X traffic behave

Scalable & Reno 16 streams are aggressive
HS LP is very timid

Fast more aggressive than HS & H
HS is timid