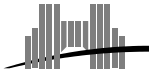


# Methodologies and Tools for exploring Transport Protocols in the context of Highspeed Networks

Romaric GUILLIER, PhD Student  
under the supervision of Pascale VICAT-BLANC PRIMET

LIP, École Normale Supérieure de Lyon, INRIA, UMR 5668, France

IGTMD 2008 – June 30<sup>th</sup>, 2008



# Outline

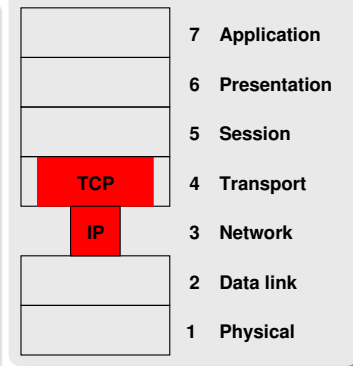
1 Problem definition

2 Proposal

3 Results

# What is TCP ?

- Vital part of the networking stack, providing **reliable** data transfer, flow control and error control [TCP 81, Cerf 74]
- Fully **distributed** algorithm in the end-hosts (scalability)
- Allows **fair** sharing of links
- **Stable** [Chiu 89]
- 80% to 95% of Internet traffic is TCP



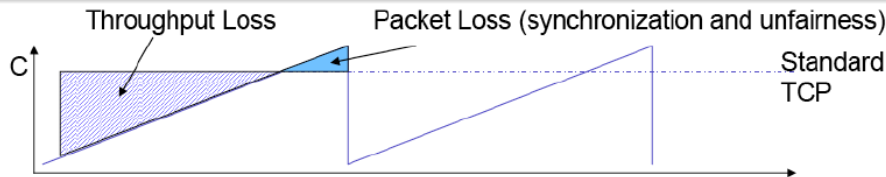
## Future of the Internet

- Technology driven: wireless networks, Fiber To The Home (DSL 5 Mbps → 100Mbps )
- Application driven: multimedia (VoD), large scale computing (low aggregation level, low multiplexing factor)
- Will TCP still be “useful” in the future ?

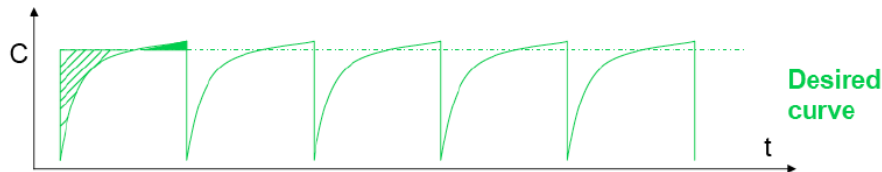
# How does TCP Congestion Control work?

## TCP Congestion window evolution (AIMD) [Jacobson 88]

- ACK :  $cwnd \leftarrow cwnd + \frac{\alpha}{cwnd}$
- Drop :  $cwnd \leftarrow cwnd - \beta * cwnd$
- **Reno**[Jacobson88] :  $\alpha = 1; \beta = \frac{1}{2}$



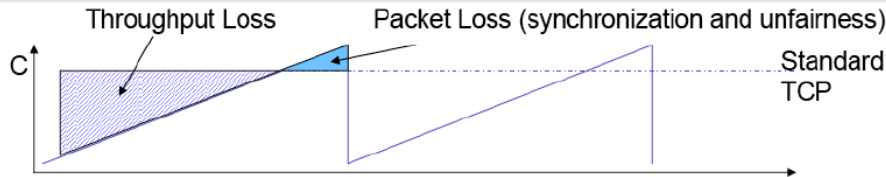
[Shrikant2007]



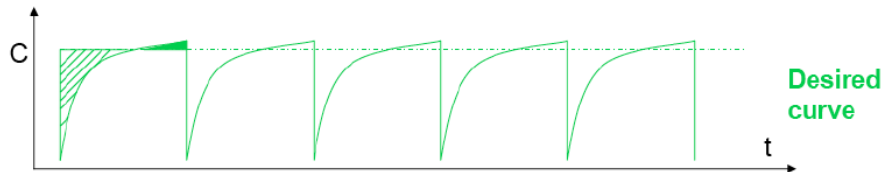
# How does TCP Congestion Control work?

## TCP limits in specific contexts

- TCP and multimedia applications (retransmissions adding delay for the application)
- TCP and wireless networks (loss not due to congestion)
- TCP and high Bandwidth Delay Product (BDP), especially with large RTT



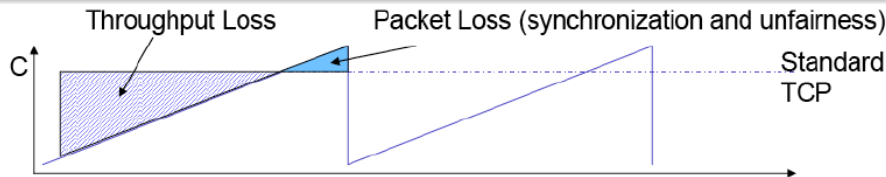
[Shrikant2007]



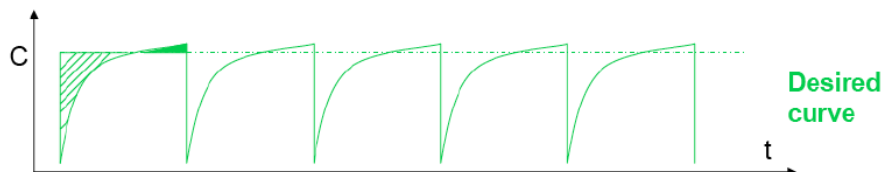
# How does TCP Congestion Control work?

## TCP limits in high BDP

- Simplified TCP model:  $Rate = \frac{MSS}{RTT} \sqrt{\frac{3}{2p}}$  [Padhye 98]
- 1 packet drop every 5e9 packets for 10 Gbps steady-state throughput on 100 ms RTT with 1500-byte packets
- “Only” operates at 75% of the capacity in average (and 25% of 10 Gbps is a lot)



[Shrikant2007]



Changing TCP ?

# Known Transport Protocol solutions

- Parallel streams
- UDP streams
- TCP variants

TCP variant	$\alpha$	$\beta$
TCP Reno [Jacobson 88]	1	$\frac{1}{2}$
BIC [Xu 04]	1 or <i>bin.search</i>	$\frac{1}{8}$
CUBIC [Rhee 05]	<i>cub(cwnd, history)</i>	$\frac{1}{5}$
HighSpeed TCP [Floyd 03]	<i>inc(cwnd)</i>	<i>decr(cwnd)</i>
Hamilton TCP [Shorten 04]	<i>f(last<sub>loss</sub>)</i>	$1 - \frac{RTT_{min}}{RTT_{max}}$
Scalable TCP [Kelly 03]	$0.01 * cwnd$	$\frac{1}{8}$

AIMD constants of several TCP variants

TCP variant	c	d
TCP Reno	1.22	0.5
BIC	15.5	0.5
HighSpeed TCP	0.12	0.835
Hamilton TCP	0.12	0.835
Scalable	0.08	1.0

Response function parameters of several TCP variants  $R = \frac{MSS}{RTT} \frac{c}{p^d}$

## TCP variants

- Since 2002, more than 10 TCP variants proposed
- Changing the AIMD  $\alpha$  and  $\beta$  to improve the response function
- But some have shown severe fairness/convergence problem
- **Need to define the good properties they should have**



# TCP Performance Metrics [Flo 07]

- TCP is a very complex protocol with a lot of requirements
- TMRG workgroup is current working on these aspects [Andrew 08, Flo 07, Flo 06]

Metric of	User Perspective	Network Perspective
<b>Throughput</b>	Goodput <b>G</b> , Completion time <b>T</b> , Cong. window <b>cwnd</b>	Throughput <b>X</b> , Link utilisation <b>U</b> , Efficiency <b>E</b>
<b>Delay</b>	RTT	Queueing delay <b>q</b>
<b>Packet loss rates</b>	Retransmission <b>r</b> Timeouts events <b>t</b>	Packet loss rate <b>p</b>
<b>Response to sudden changes</b>	Responsiveness <b>R</b> , Aggressiveness <b>A</b>	Smoothness <b>S</b>
<b>Minimizing oscillations</b>	Variance $\sigma$	Coeff. of Variation <b>CoV</b>
<b>Fairness and convergence times</b>	Jain Index <b>J</b> Delta-fair convergence $\delta_f$	Max-min, Proportional, Epsilon fairness
<b>Robustness</b>		
<b>Deployability</b>		Code complexity

Need for methodologies to study its behaviour

# Existing Methodologies

Methodology	Wang [NS2 07]	Mascolo [Mascolo 06]	Rhee [Ha 06]	Leith [Li 06]	Kumazoe [Kumazoe 07]
Type	Simulation	Simulation	Sw. emul.	Sw. emul.	Real
Topology	Dumbbell, Parking Lot, 4 Domain Network	Dumbbell	Dumbbell	Dumbbell	Dumbbell
Number of sources	n/a	6	4	2	2
Rate max (Mbps)	n/a	250	400	250	10000
RTT range (ms)	n/a	40,80,160	16,64,162, 324	16,22,42, 82, 162	18,180
Traffic model	FTP, Web, Voice Video streaming	FTP, Web	FTP, Web	FTP, Web	FTP
Metrics	$X, q, \sigma_{RTT},$ $p, J, R,$ $\delta_f, \text{robustness}$	cwnd, t	$p, J, \delta_f$	$U, G, \text{cwnd},$ $p, J$	X

- No consensus on chosen parameters value (RTT, Rate max)
- No consensus on chosen metrics
- No consensus on the scenarios
- Small number of sources used
- What tool should be used?

# Simulation vs Emulation vs Real experiment

	<b>Simulation</b>	<b>Sw. Emul.</b>	<b>Hw. Emulation</b>	<b>Real</b>
Examples	NS-2, OMNeT++	Dummynet, NISTNet	AIST-GtrcNet	WanInLab, Grid'5000, PlanetLab
<b>Pros</b>	Simple models Parameter decoupling Fine grained control	Easy to setup Coarse grained control	Easy to setup Fine grained control	Real equipment Real behavior
<b>Cons</b>	CPU intensive Memory intensive Disk intensive Phase effect Limited models	CPU intensive Memory intensive Software overhead Precision limitation	Cost Limited parameters Black boxes	Cost Limited range Limited topologies Black boxes Bugs

- [Wei 06] shows exponential simulation time in NS-2 with the bandwidth
- At 1 Gbps, the max packet rate is 83333 packets/s (MTU 1500 bytes), too much to be handled by current hardware at wire speed (Software emulation)
- Presence of black boxes in real networks
- Tools are **complementary**
- Open question: Comparaison possible between each approach ?

# Outline

- 1 Problem definition
- 2 Proposal
  - Methodology
  - Network eXperiment Engine
- 3 Results

# Steps for a performance evaluation study [Jain 91]

- ① State the goals and define the system boundaries
- ② List system system services
- ③ Select performance metrics
- ④ List system and workload parameters
- ⑤ Select factors and their values
- ⑥ Select evaluation techniques
- ⑦ Select the workload
- ⑧ Design the experiments
- ⑨ Analyze and interpret the data
- ⑩ Present the results

# Scenario example

Study how a transport protocol adapts to abrupt changes in traffic conditions (heavy congestion event).

Metrics: responsiveness

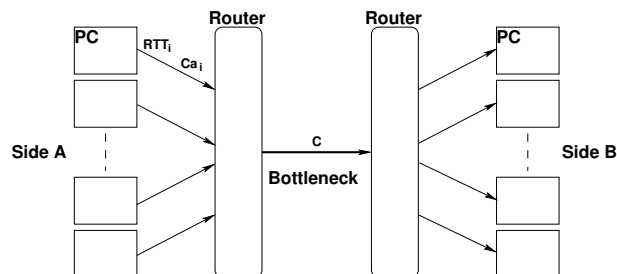


**[0-T1]** Stable situation, light congestion level (0.5)

**[T1-T2]** Major change, high congestion level/change in the mix of transport protocols

**[T2-T3]** Stable situation, light congestion level (0.5)

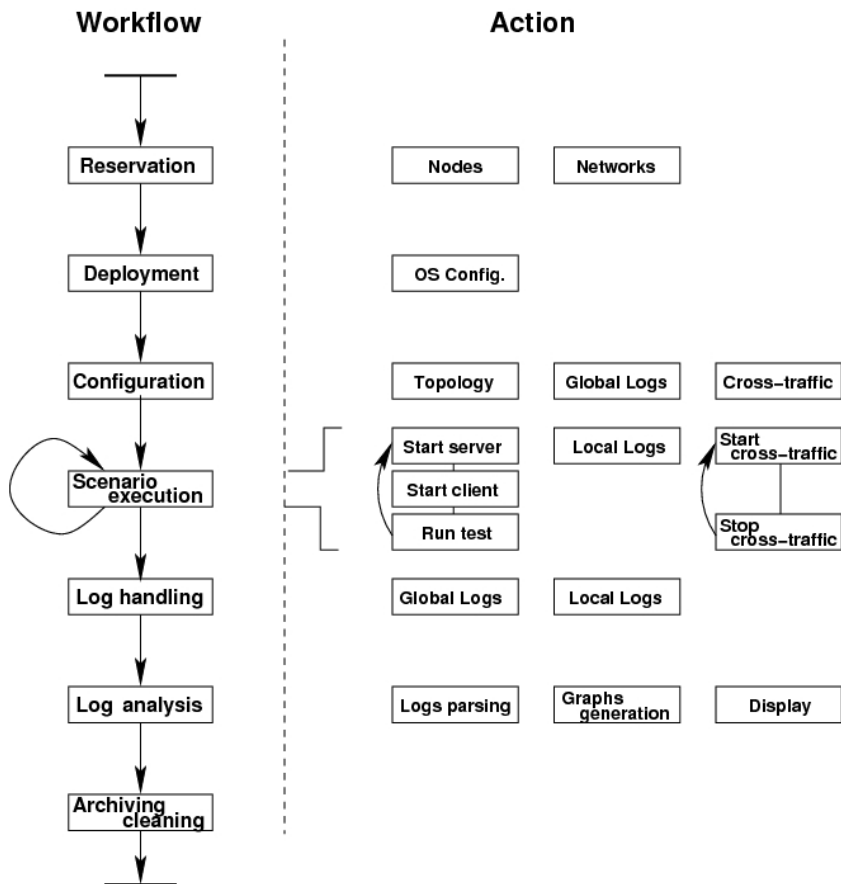
# Parameter space



	Parameter	Description	Range
<b>Infrastructure</b>	RTT	Round Trip Time	0 to 200 ms
	$C$	Bottleneck capacity	1 or 10 Gbps
	$K = \frac{C}{C_a}$	Aggregation lvl	1 or 10
<b>Workload</b>	$M$	Multiplexing factor	1 to 20
	$N_s$	Parallel streams	1 to 10
	$C_g$	Congestion factor	0 to 2.0
	$R$	Reverse traffic factor	0 to 2.0

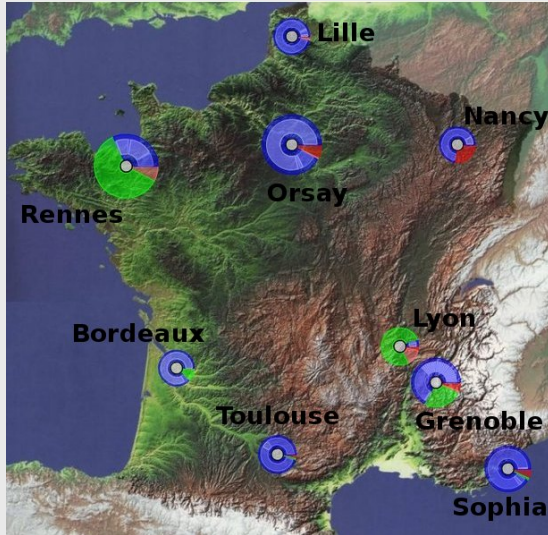
- Huge parametric space
- Experiments must be repeated several times to have a good statistical sample
- Need a tool to automatise this process

# Workflow





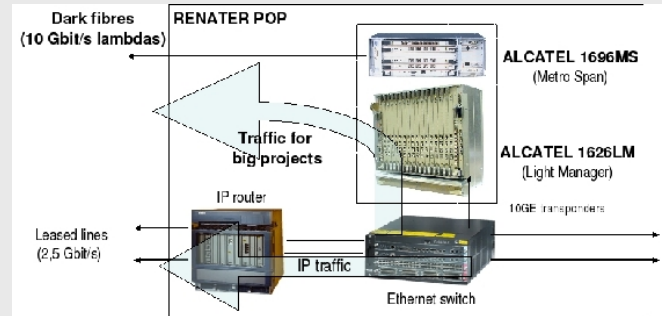
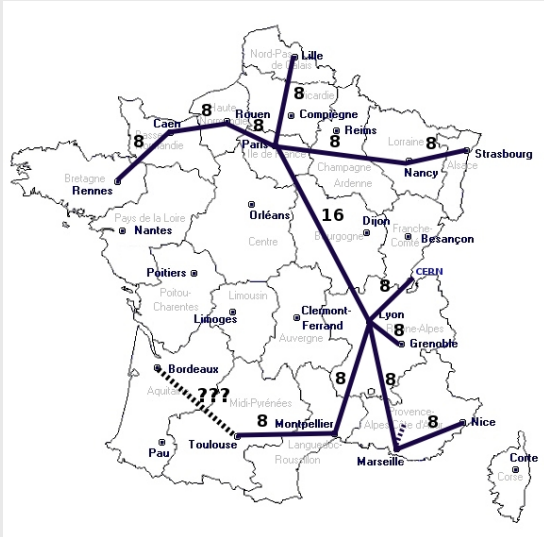
# Grid5000 [Bolze 06]: Description



Site	CPU available	CPU scheduled
Bordeaux	424	500
Grenoble	270	500
Lille	198	500
Lyon	260	500
Nancy	334	500
Orsay	684	1000
Rennes	524	522
Sophia	356	500
Toulouse	276	500
Total	3326	5022



- 9 sites in France, 17 laboratories involved
- 5000 CPUs (currently 3300)
- Private 10Gbps Ethernet over DWDM network
- Experimental testbed for Networking to Application layers.

# Grid5000 [Bolze 06]: Description



- 9 sites in France, 17 laboratories involved
- 5000 CPUs (currently 3300)
- Private 10Gbps Ethernet over DWDM network
- Experimental testbed for Networking to Application layers.

# Grid5000 [Bolze 06]: Special Features

- A high security for Grid'5000 and the Internet, despite the deep reconfiguration feature
  - ↪ Grid'5000 is confined: communications between sites are isolated from the Internet and Vice versa (level2 MPLS, Dedicated lambda).
- A software infrastructure allowing users to access Grid'5000 from any Grid'5000 site and have simple view of the system
  - ↪ A user has a single account on Grid'5000, Grid'5000 is seen as a cluster of clusters, 9 (1 per site) unsynchronized home directories
- A reservation/scheduling tools allowing users to select nodes and schedule experiments
  - ↪  Reservation engine + batch scheduler (1 per site) + OAR Grid (a co-reservation scheduling system)
- A user toolkit to reconfigure the nodes
  - ↪  Software image deployment and node reconfiguration tool

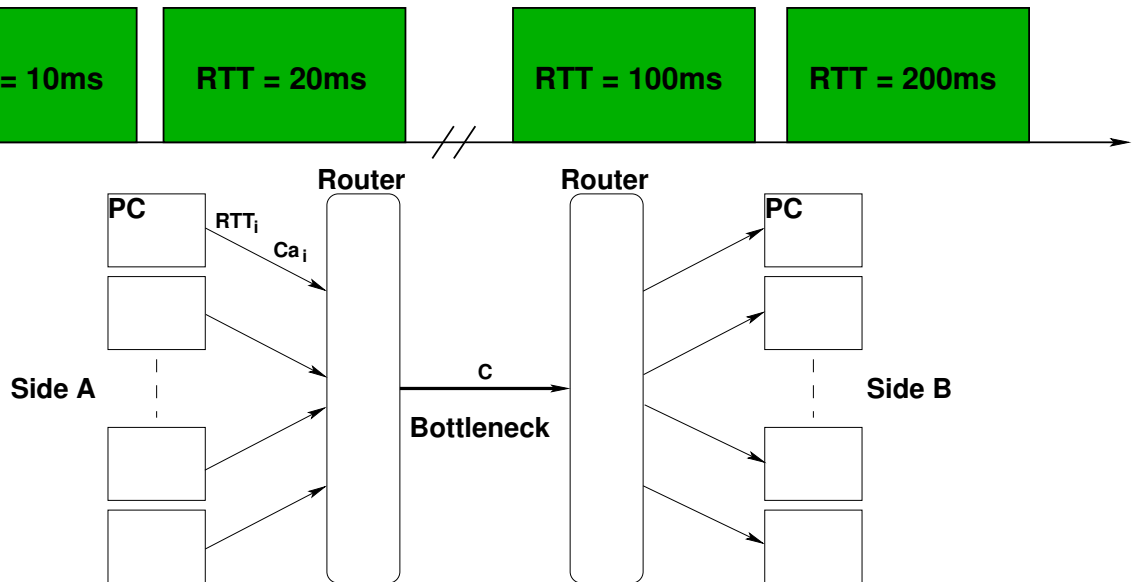
# Outline

- 1 Problem definition
- 2 Proposal
- 3 Results
  - Influence of latency
  - Influence of the multiplexing factor
  - Influence of traffic conditions
  - Influence of reverse traffic level

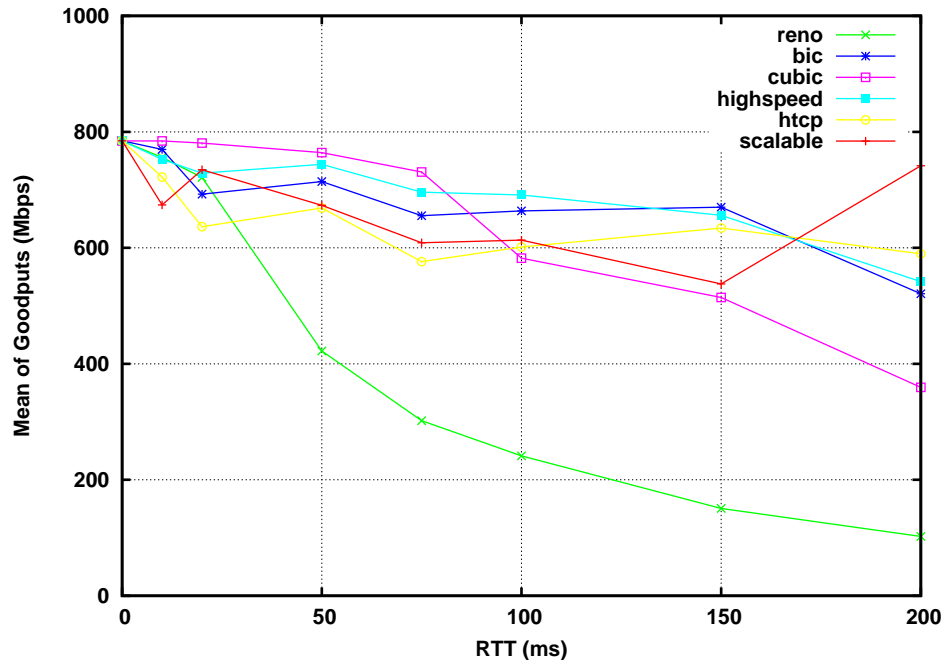
# Experimental setting

Study of the **impact of latency** on the performance of TCP variants [Guillier 07a]

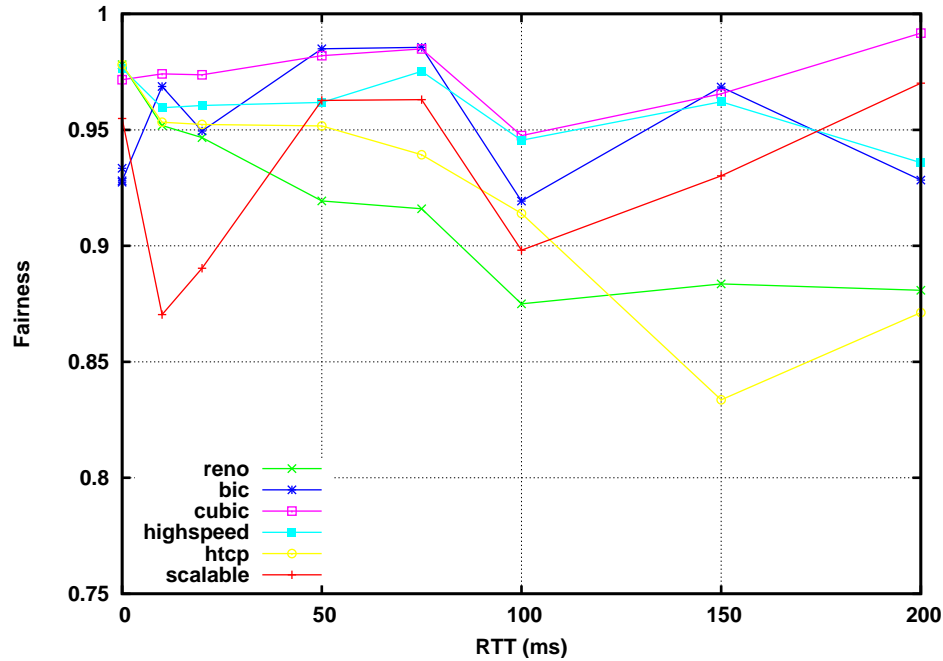
12 independant sources, transmitting continuously. A new source starts every 200 s.



# Impact of the latency on mean goodput



# Impact of the latency on fairness



# Recap table

	Flow mean goodput		Mean fairness		Normalised standard deviation	
	11.5 ms	100 ms	11.5 ms	100 ms	11.5 ms	100 ms
Reno	756.0	234.3	0.951	0.918	0.222	0.232
BIC	781.1	653.7	0.969	0.919	0.176	0.306
CUBIC	784.5	534.3	0.974	0.961	0.144	0.140
HS-TCP	753.6	671.9	0.960	0.962	0.069	0.233
H-TCP	722.2	686.1	0.953	0.926	0.230	0.256
Scalable	674.0	540.4	0.870	0.955	0.337	0.317

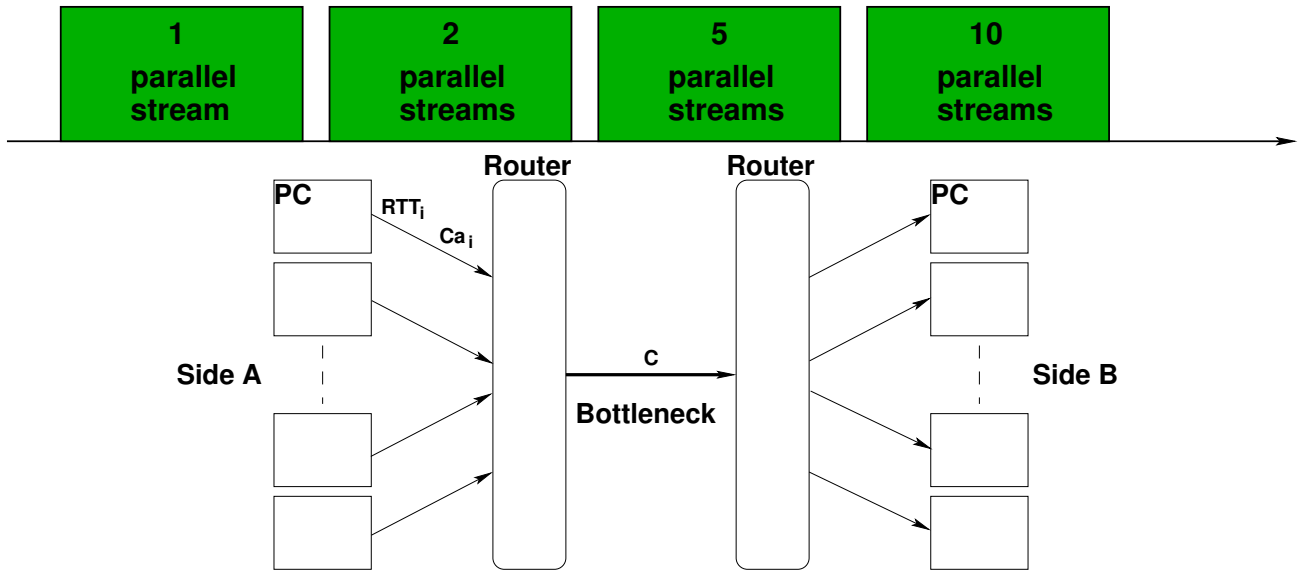
- “best” value, “worse” value.
- No universal solution.



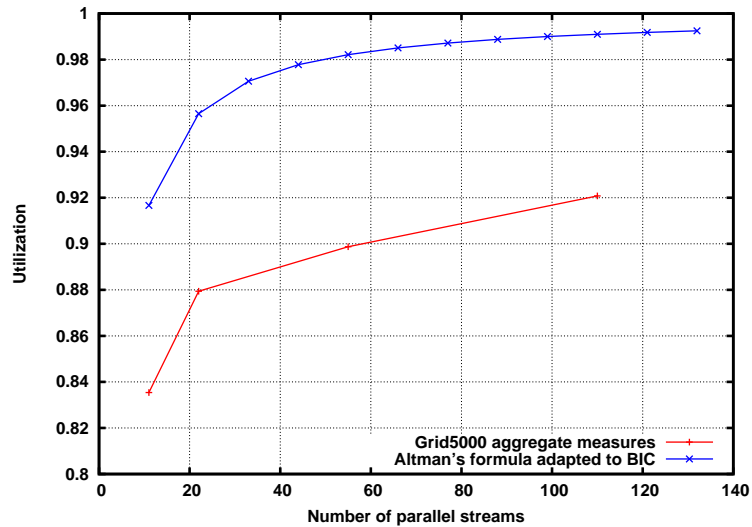
# Experimental setting

Study of the **impact of the number of parallel streams** on the global throughput [Guillier 07a]

11 independant sources, transmitting continously for 600 s



# BIC-TCP and Altman's model [Altman 06]



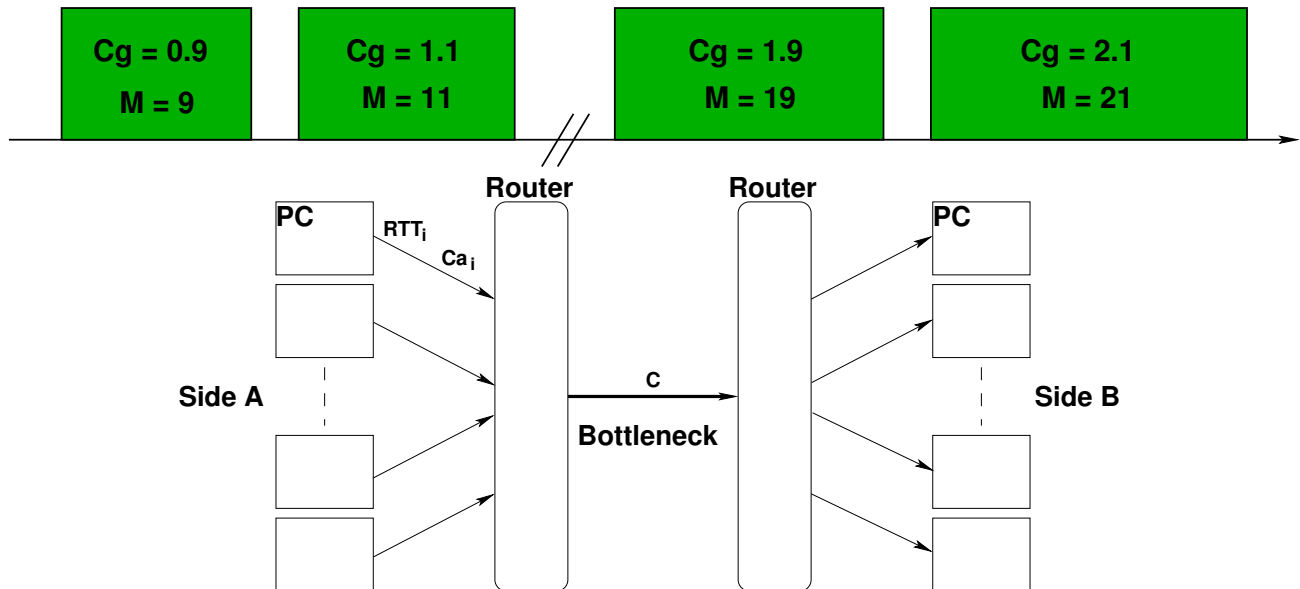
$$\text{Altman's formula } \bar{x}(N) = C \left( 1 - \frac{1}{1 + \frac{1+\beta}{1-\beta} N} \right)$$

Nb of flows by node	1	2	5	10
Mean total goodput (Mbps)	8353.66	8793.92	8987.49	9207.78
Flow mean (Mbps)	761.70	399.83	163.53	83.71
Jain Index	0.9993	0.9979	0.9960	0.9973
Gain	/	4.9%	7.3%	9.8%

# Experimental setting

Study of the **impact of traffic conditions** (congestion factor, reverse traffic factor) on the completion time of 3000 MB file transfers [Guillier 07c].

Up to 42 independant sources, emitting simultaneously.



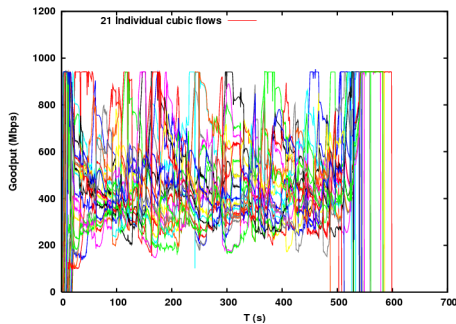
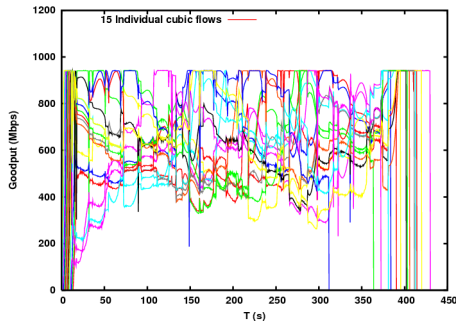
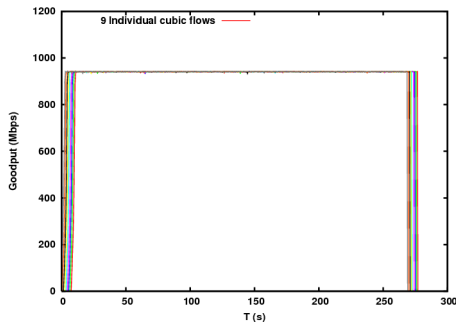
Cong.  
lvl:  $\frac{\sum C_a}{C}$

90 %:  
280 s/272 s

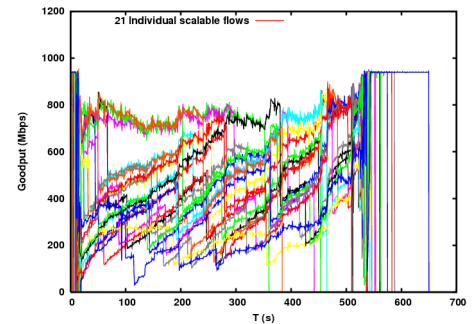
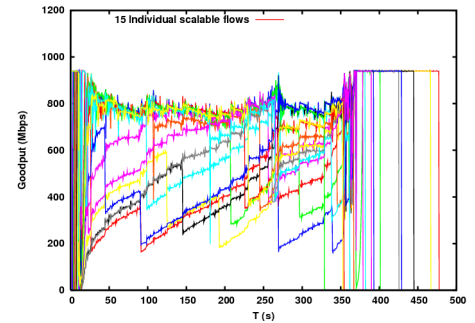
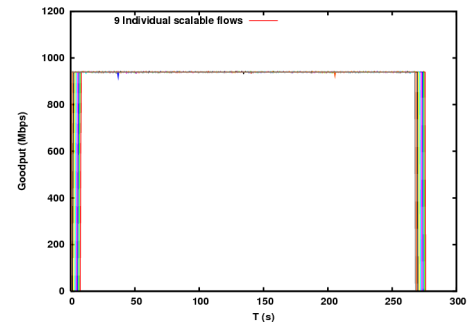
150 %:  
395 s/398 s

210 %:  
545 s/535 s

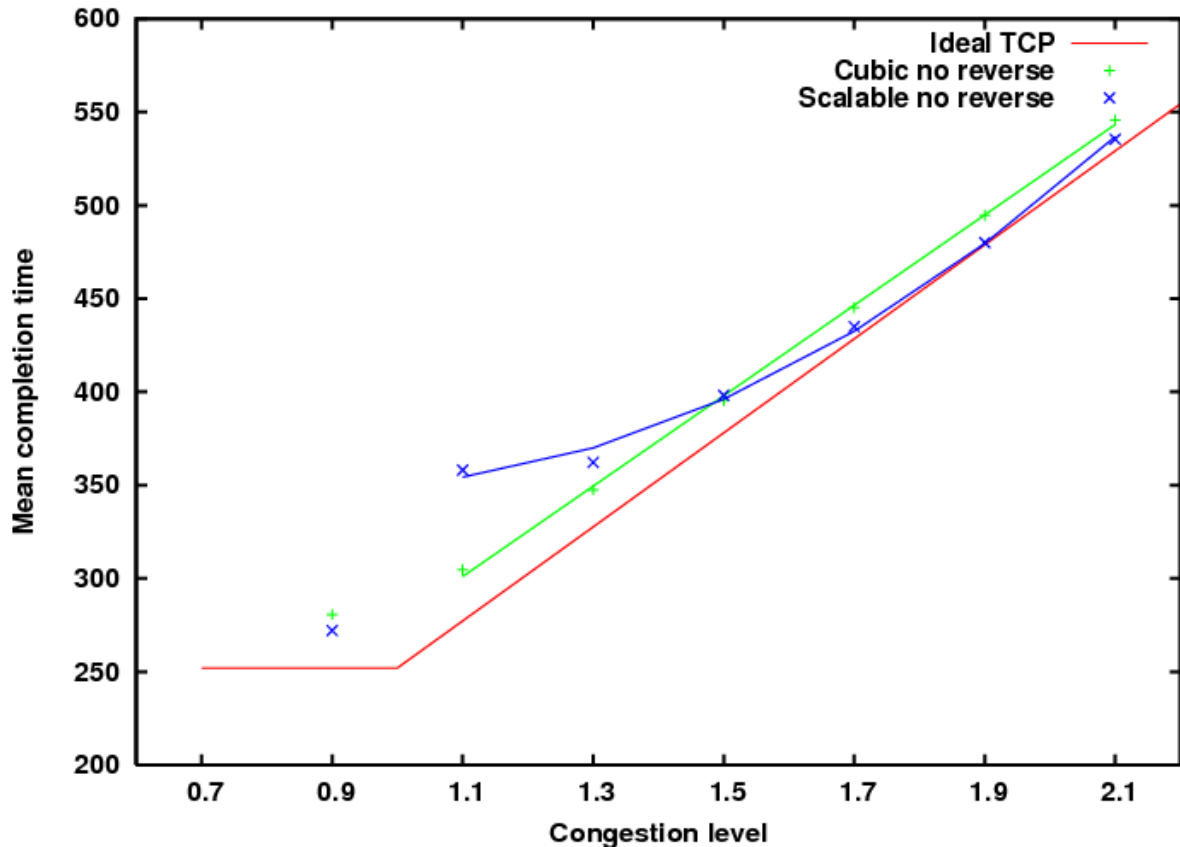
## Cubic



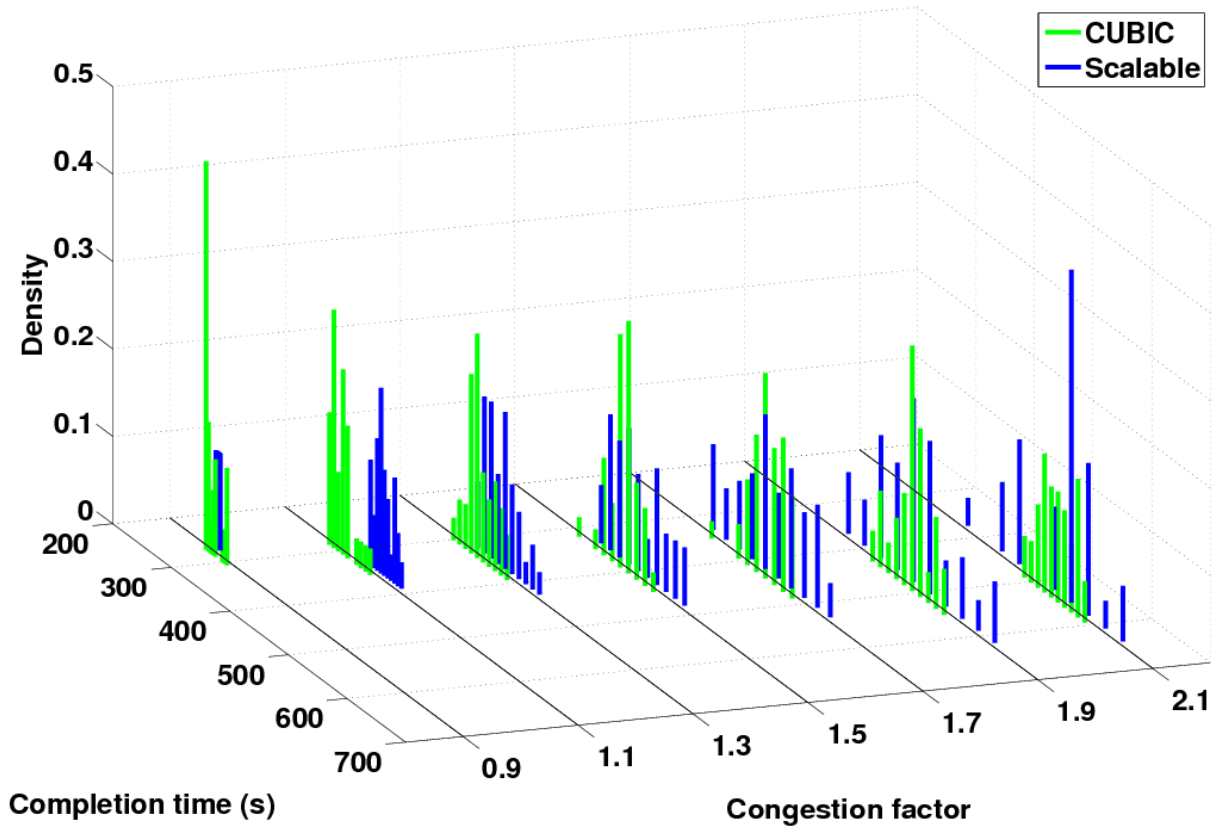
## Scalable



# Mean completion time of Cubic and Scalable

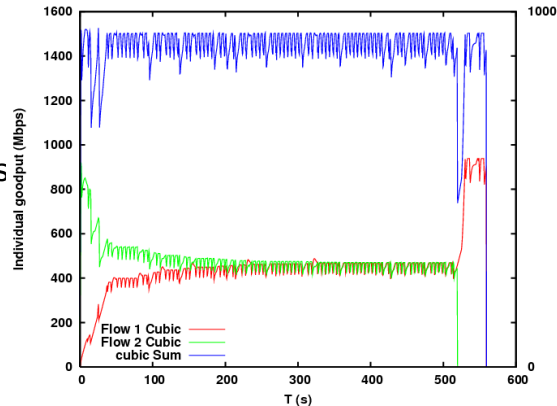


# Completion Time distribution

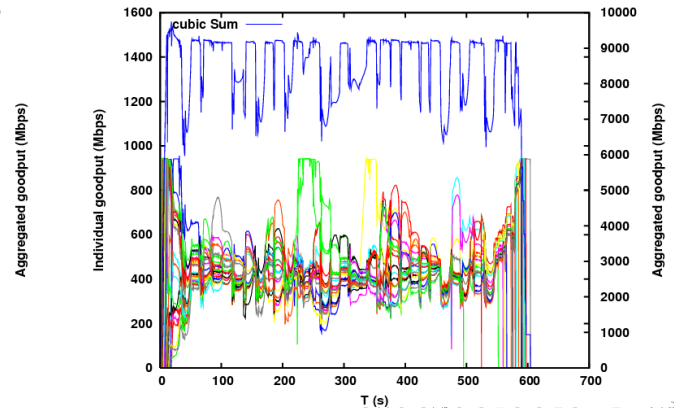
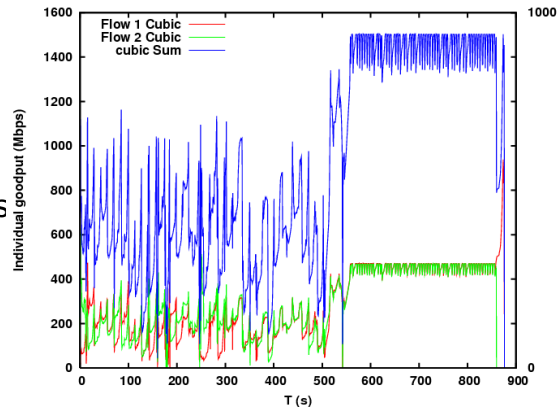
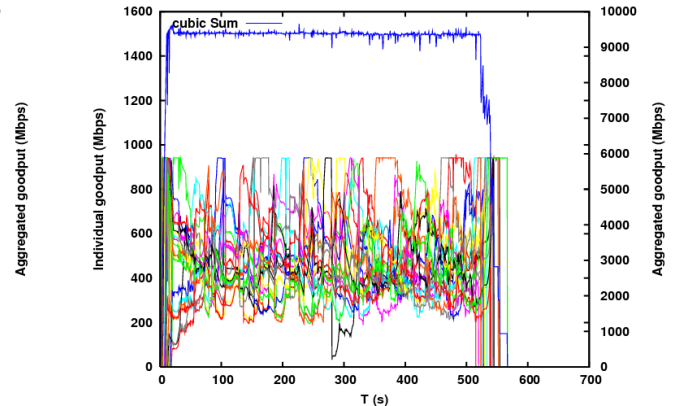


# The multiplexing factor (200 % congestion level)

2x Cubic 1 Gbps flows,  
1Gbps bottleneck

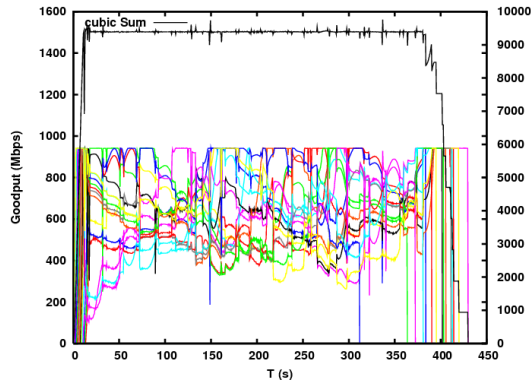


20x Cubic 1 Gbps flows,  
10Gbps bottleneck

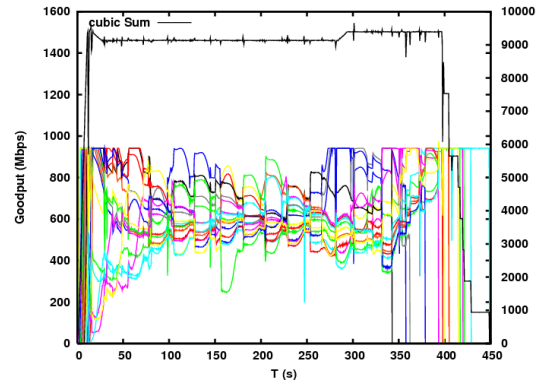


# Influence of reverse traffic on Cubic (150 % cong. lvl)

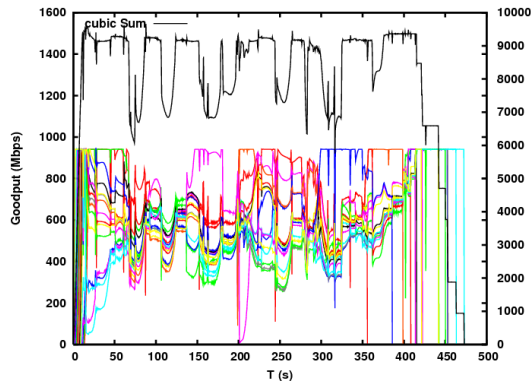
No reverse (395 s)



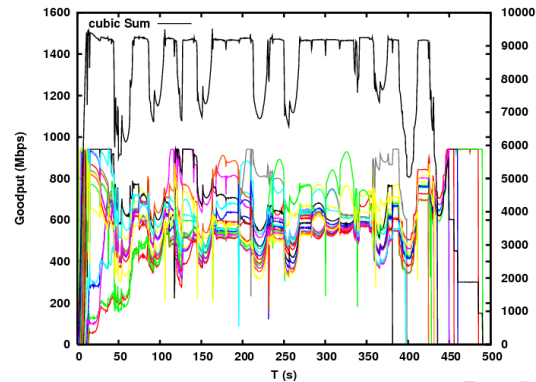
90 % reverse (400 s)



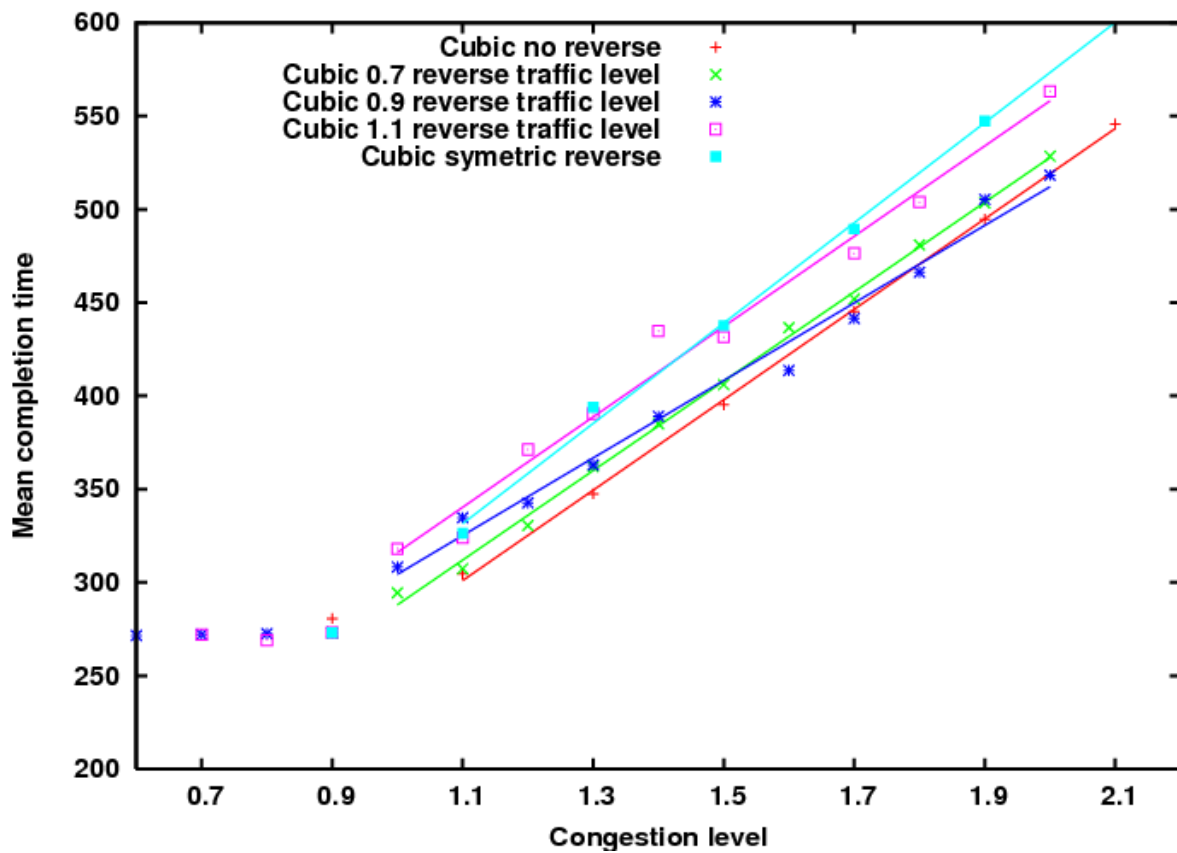
110 % reverse (432 s)



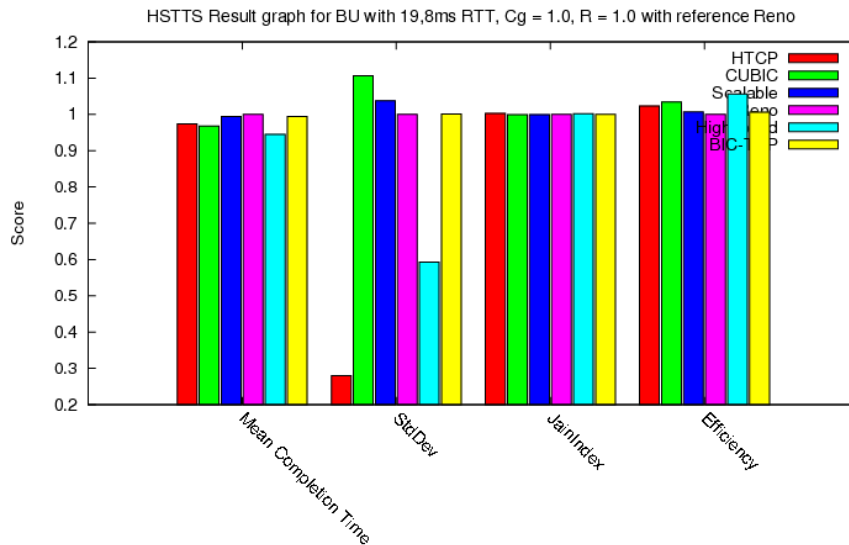
150 % reverse (438 s)







# Towards a Transport Protocol Benchmark [Guillier 07b]



# Conclusion

- Evaluation of transport protocols in high-speed networks
- NXE, a tool to automate real experiments
- Some experimental results in high speed networks

## Future and Current Works

- Contribution to TMRG transport protocol benchmark design [Andrew 08]
- Building bridges between simulation and real experiment worlds (automatic scenarios converter)
- Validation of the NXE tool
- Creating realistic scenarios of user usage
- Helping users assess their networking environment and optimize its usage
- Comparaison of the results from several tools.

# Questions?

Thanks for your attention...

# References I



Eitan Altman, Dhiman Barman, Bruno Tuffin & Milan Vojnovic.  
*Parallel TCP Sockets: Simple Model, Throughput and Validation.*  
 In Proceedings of the IEEE INFOCOM, 2006.



Lachlan Andrew, Cesar Marcondes, Sally Floyd, Lawrence Dunn, Romaric Guillier, Wang Gang, Lars Eggert, Sangtae Ha & Injong Rhee.  
*Towards a Common TCP Evaluation Suite.*  
 In PFLDNet, march 2008.



Raphaël Bolze, Franck Cappello, Eddy Caron, Michel Daydé , Frederic Desprez, Emmanuel Jeannot, Yvon Jégou, Stéphane Lanteri, Julien Leduc, Noredine Melab, Guillaume Mornet, Raymond Namyst, Pascale Vicat-Blanc Primet, Benjamin Quetier, Olivier Richard, El-Ghazali Talbi & Touché Irena.  
*Grid'5000: a large scale and highly reconfigurable experimental Grid testbed.*  
 International Journal of High Performance Computing Applications, vol. 20, no. 4, pages 481–494, November 2006.



V. Cerf & R. Kahn.  
*A Protocol for Packet Network Intercommunication.*  
 In IEEE Transactions on Communications, volume 22, pages 637–648, may 1974.



D. Chiu & R. Jain.  
*"Analysis of the Increase/Decrease Algorithms for Congestion Avoidance in Computer Networks.*  
 Journal of Computer Networks and ISDN, vol. 17, no. 1, pages 1–14, June 1989.




*Tools for the Evaluation of Simulation and Testbed Scenarios.*  
 In Sally Floyd & E Kohler, editeurs, <http://www.icir.org/tmrg/draft-irtf-tmrg-tools-03.txt>, December 2006.





*Metrics for the evaluation of Congestion Control Mechanisms.*  
 In Sally Floyd, editeur, <http://www.icir.org/tmrg/draft-irtf-tmrg-metrics-11.txt>, October 2007.


# References II


 Sally Floyd.  
*RFC 3649: HighSpeed TCP for Large Congestion Windows.*  
RFC 3649, December 2003.  
experimental.

 Romaric Guillier, Ludovic Hablot, Yuetsu Kodama, Tomohiro Kudoh, Fumihiko Okazaki, Ryousei Takano, Pascale Vicat-Blanc Primet & Sebastien Soudan.  
*A study of large flow interactions in high-speed shared networks with Grid5000 and GtrcNET-1.*  
In PFLDnet'07, February 2007.

 Romaric Guillier, Ludovic Hablot & Pascale Vicat-Blanc Primet.  
*Towards a User-Oriented Benchmark for Transport Protocols Comparison in very High Speed Networks.*  
Research Report 6244, INRIA, 07 2007.  
Also available as LIP Research Report RR2007-35.

 Romaric Guillier, Sebastien Soudan & Pascale Vicat-Blanc Primet.  
*TCP variants and transfer time predictability in very high speed networks.*  
In Infocom 2007 High Speed Networks Workshop, May 2007.

 Sangtae Ha, Long Le, Injong Rhee & Lisong Xu.  
*A Step toward Realistic Performance Evaluation of High-Speed TCP Variants.*  
Elsevier Computer Networks (COMNET) Journal, Special issue on "Hot topics in transport protocols for very fast and very long distance networks", 2006.

 Van Jacobson.  
*Congestion Avoidance and Control.*  
In SIGCOMM'88, 1988.

# References III



R. Jain.

The art of computer systems performance analysis: Techniques for experimental design, measurement, simulation, and modeling.

Wiley- Interscience, April 1991.



Tom Kelly.

*Scalable TCP: Improving Performance in Highspeed Wide Area Networks.*

In Computer Communication Review, volume 32, April 2003.



Kazumi Kumazoe, Masato Tsuru & Yuji Oie.

*Performance of high-speed transport protocols coexisting on a long distance 10Gbps testbed network.*

In GridNets, october 2007.



Yee-Ting Li, Douglas Leith & Robert N. Shorten.

*Experimental Evaluation of TCP Protocols for High-Speed Networks.*

In Transactions on Networking, June 2006.



Saverio Mascolo & Francesco Vacirca.

*The effect of reverse traffic on the performance of new TCP congestion control algorithm.*

In PFLDnet'06, February 2006.



*An NS2 TCP Evaluation Tool Suite.*

In G. Wang, Y. Xia & D. Harrison, editors, <http://www.icir.org/tmrg/draft-irtf-tmrg-ns2-tcp-tool-00.txt>, April 2007.



J. Padhye, V. Firoiu, D. Towsley & J. Kurose.

*Modeling TCP Throughput: A Simple Model and its Empirical Validation.*

In ACM SIGCOMM '98, 1998.





Injong Rhee & Lisong Xu.


*CUBIC: A New TCP-Friendly High-Speed TCP Variants.*


In PFLDnet, 2005.

# References IV

 R.N. Shorten & Doug Leith.  
*H-TCP: TCP for high-speed and long-distance networks.*  
In PFLDnet'04, Argonne, Illinois USA, February 2004.

 *Transmission Control Protocol.*  
RFC 793, september 1981.

 David X. Wei & Pei Cao.  
*NS-2 TCP-Linux: an NS-2 TCP implementation with congestion control algorithms from Linux.*  
In WNS2 '06: Proceeding from the 2006 workshop on ns-2: the IP network simulator, page 9, New York, NY, USA, 2006. ACM Press.

 Lisong Xu, Khaled Harfoush & Injong Rhee.  
*Binary Increase congestion Control for Fast Long-Distance Networks.*  
In INFOCOM, 2004.