An initiative for a classified bibliography on TCP/IP congestion control

Saleem-ullah Lar a,*, Xiaofeng Liao b

a College of Computer Science, Chongqing University, Chongqing 400044, China
b Faculty of Computer Science and Engineering, Senior Member IEEE, Chongqing University, Chongqing 400044, China

ARTICLE INFO
Article history:
Received 11 October 2011
Received in revised form 4 March 2012
Accepted 10 April 2012

Keywords:
TCP
Congestion control
Slow start
Congestion avoidance
Fast retransmit
Fast recovery

abstract
During last two decades researchers, scholars and students are continuously embracing and improving TCP congestion performance both in wired and wireless networks by focusing on four modules of congestion control algorithms i.e., slow start, congestion avoidance, fast recovery, and fast retransmit, which are considered to be the integrated models for network congestion. This paper presents the creativity to collect and classify bibliography on different flavors TCP/IP congestion control during these two decades. We have extracted some core results from the bibliography provided here which are described in the form of tables and diagrams.

© 2012 Elsevier Ltd. All rights reserved.

Contents
1. Introduction .................................................................1
2. Why congestion control ..................................................1
4. Conclusion ...............................................................5
References ........................................................................6

1. Introduction
Practically all Internet applications rely on the Transmission Control Protocol (TCP) (Postel, 1981a, 1981b) to deliver data reliably across the network. Although it was not part of its primary design, the most vital element of TCP is congestion control, it defines TCP’s performance characteristics. In this paper we have present a bibliography survey of the congestion control proposals for TCP that preserve its necessary host-to-host congestion control approaches. After careful study, observation and discussion I have extracted some core results from the bibliography provided in this article, which I have showed in the form of figures and tables. From these one can get the precise and exact information about the required paper with in short time. The survey highlighted the fact that the research focus has changed with the development of the Internet, from the basic problem of eliminating the congestion collapse (Gerla and Kleinrock, 1980; Nagle, 1984; Kent and Mogul, 1987; Floyd and Fall, 1999) phenomenon to problems of using available network resources effectively in different types of environments (wired, wireless, high-speed, long-delay, etc.).

2. Why congestion control
Congestion is a problem that occurs on shared networks when multiple users contend for access to the same resources (bandwidth, buffers, and queues). It concerns controlling traffic entry into a network, so as to avoid congestive collapse by attempting to avoid oversubscription of any of the processing or link capabilities of the intermediate nodes and networks and taking resource reducing steps, such as reducing the rate of sending packets.

Congestion occurs when there is too much traffic in the network routers has queuing capability. If a router cannot transmit packets at a given instance, it stores packets in the
queue and waits for the next chance to transmit. Queue has limited size, if queue data exceeds limit, packet will be discarded.

If congestion occurs then packet transfers are delayed and discarded, due to this some protocols/applications try to retransmit data. Users try to retransmit the data or request the same data again and again. In this case ratio of valid data is decreasing and in the end congestion collapse occurs and difficult to use network resources. Therefore we need to control this congestion to improve network quality of service.

But still congestion control is difficult due to the following reasons:

a. Internet is designed to be autonomous.
b. Internet is very huge and still expanding.
c. No centralized control.
d. No way to control each user behavior.
e. It is difficult to determine how many user/application share the network exactly.
f. It is difficult to determine the source of the congestion exactly.
g. It is difficult to determine the capacity of the network exactly.
h. It is difficult to determine how much networks are congested exactly.
i. It is difficult to determine why packets are lost exactly.

This is the reason it is still critical issue for researchers. However various efforts have been done in the last twenty years to solve this network problem which I have collected and present in the next section.


In this section we have describe the most important work done in the last two decades. Extracting core results from the papers and presenting in the form of tables and figures will be helpful for students and researchers. This section is devoted to congestion control bibliography that builds a foundation for all presently known host-to-host algorithms.

Objective of Fig. 1 is to show work done on various TCP flavors which discuss many solutions of several congestion control problems on three major categories, i.e., loss based, delay based and loss based with bandwidth estimation. Somehow this Fig. 1 has almost same objective with Table 5 to design an ideal algorithm for long delayed and high speed networks.

Regrettably Protocol standards that remain unaware of the network resources have created various unexpected results on the Internet, including the appearance of congestion collapse. The problem of congestion control, meaning intelligent (i.e., network resource-aware) and yet effective use of resources available in packet-switched networks, is not at rival problem, but the efficient explanation to it is highly desirable. As a result, congestion control is one of the extensively studied as in the Internet research conducted over the last 20 years, and a number of proposals aimed at improving various aspects of the congestion-responsive data flows is very large. Several groups of these proposals have been studied in AlHanbali et al. (2005) (congestion control in adhoc networks), Lochert et al. (2007) (congestion control for mobile adhoc networks), Widmer et al. (2001) (congestion control for non-TCP protocols), Balakrishnan et al. (1997) (congestion control for wireless networks), Leung et al. (2007) (congestion control for networks with high levels of packet reordering), Low et al. (2002) (current up to 2002 TCP variants and their analytical models), Hasegawa and Murata (2001) (fairness issues in congestion control), and others researchers. Unlike previous studies, in this survey we tried to collect, classify, and analyze major congestion control algorithms that optimize various parameters of TCP data transfer without relying on any explicit notifications from the network. In other words, they preserve the host-to-host principle of TCP, where by the network is seen as a black box. See Fig. 2 for an evolutionary graph of variants of TCP congestion control.

Table 1 describes the summary of the features of various algorithms, moreover Fig. 3 shows the evolutionary graph of these algorithms.

Algorithm in Paxson (1997) describe that if receiver can detect and report packet loss, the acknowledgment will arrive at the sender exactly one RTT after the arrival of loss packet. If we want an
immediate reply from TCP receiver with loss reports as last in order packet (Przybylski et al., 2005), then the loss can be detected by the fast retransmit algorithm exactly within the interval of RTT (Nichols et al., 1998), and packets reordering issues are mitigate in Blake et al. (1998), Karn and Partridge (1987), Braden (1989), Stevens (1997), Chiu and Jain (1989), Floyd (1998), Jacobson et al. (1992) with different scenarios. Due to the resource sharing nature of IP networks effectiveness is not only the important parameter for congestion control, TCP should also enforce efficient resource sharing as described in Sing and Soh (2005).

Table 2 summarize the following ideas:

i. They allow nonzero probability of packet reordering.

ii. They can detect out-of-order events and respond with an increase in flow rate (optimistic reaction). Nonetheless, these proposals have fundamental differences due to arrange of acceptable degrees of packet reordering, from moderate in TD-FR (time delayed fast recovery) to extreme in TCP PR (packet reordering), and different baseline congestion control approaches.

Fig. 2. Evolutionary graph of TCP congestion control modifications.
Features of tcp modifications/variants that solve the congestion collapse problem.

<table>
<thead>
<tr>
<th>TCP variant</th>
<th>Base</th>
<th>Year</th>
<th>Update features*</th>
<th>Modification</th>
<th>Status*</th>
<th>Implementation (version)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tahoe (Jacobson, 1988)</td>
<td>RFC793</td>
<td>1988</td>
<td>Queuing delay as a supplemental congestion Prediction parameter for CA</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
</tr>
<tr>
<td>DUAL (Wang and Crowcroft, 1992)</td>
<td>Tahoe</td>
<td>1992</td>
<td></td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
</tr>
<tr>
<td>Reno (Jacobson, 1990; Allman et al., 1999)</td>
<td>Reno</td>
<td>1990</td>
<td>FR</td>
<td>Sender</td>
<td>Std.</td>
<td>&gt;4.3³</td>
</tr>
<tr>
<td>New Reno (Floyd and Henderson, 1999; Floyd et al., 2004)</td>
<td>Reno</td>
<td>1999</td>
<td>FR resistant to multiple losses.</td>
<td>Sender</td>
<td>Std.</td>
<td>&gt;F4</td>
</tr>
<tr>
<td>SACK (Mathis et al., 1996)</td>
<td>RFC793</td>
<td>1996</td>
<td>Extended information in feedback messages.</td>
<td>Sender</td>
<td>Exp.</td>
<td>&gt;1.3.90</td>
</tr>
<tr>
<td>FACK (Mathis and Mahdavi, 1996)</td>
<td>Reno, SACK</td>
<td>1996</td>
<td>SACK-based loss recovery algorithm.</td>
<td>Sender</td>
<td>Exp.</td>
<td>&gt;95/NT</td>
</tr>
<tr>
<td>Vegas (Brakmo and Peterson, 1995)</td>
<td>Reno</td>
<td>1995</td>
<td>Bottleneck buffer utilization as a primary feedback for the CA and secondary for the SS.</td>
<td>Sender</td>
<td>Exp.</td>
<td>&gt;10.4.6</td>
</tr>
<tr>
<td>Vegas+(Hasegawa et al., 2000)</td>
<td>New Reno/Vegas</td>
<td>2000</td>
<td>Reno/Vegas CA mode switching based on RTT dynamics.</td>
<td>Sender</td>
<td>Exp.</td>
<td>&gt;2.1.36</td>
</tr>
<tr>
<td>Veno (Fu and Liew, 2003)</td>
<td>New Reno/Vegas</td>
<td>2002</td>
<td>Reno-type CA and FR increase/ decrease coefficient adaptation based on bottleneck buffer state estimation.</td>
<td>Sender</td>
<td>Exp.</td>
<td>&gt;2.1.92</td>
</tr>
<tr>
<td>Vegas-A (Srijith et al., 2005)</td>
<td>Vegas</td>
<td>2005</td>
<td>Adaptive bottleneck buffer state aware CA.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
</tr>
</tbody>
</table>

* SS=Slow Start, CA=Congestion Avoidance, FR=Fast Recovery.
* Exp=Experimental, Std=Standard.
* Barkley Software Distribution [S=Sun, F= FreeBSD, N=NetBSD].
* > (greater than) represents Kernel/Major release version for BSD, Linux and Mac.

Quality of service is an important factor in TCP performance, but due to high internet heterogeneity, many attempts (Clark and Fang, 1998; Xiao and Ni, 1999; Davie, 2003) have done to improve QoS functionality on the network. To mitigate the deployment issues with providing some level of QoS host to host based prioritization have been proposed in Table 3, while Table 4 shows characteristic features of refinements in Westwood that try to mitigate discovered problems.

The rapid increase of wireless networks has enlightened the need for TCP amendment. Originally designed for wired networks where congestion is the major cause of packet losses, TCP is not capable to respond effectively to packet losses not related to congestion. Indeed, if a data packet is lost due to short-term radio frequency interference, then there are no router buffers over flows and TCP’s decision to reduce the congestion window is erroneous. Instead, it should just recover from the loss and continue the transmission as if nothing had happened. Numerous techniques have been proposed to resolve this problem. One group gives up the idea of a pure peer to peer data transfer either by (a) requiring routers to reveal the network state (e.g., using ECN (Rama Krishnan et al., 2001)), by (b) relying on network channels to recover from the non-congestion-related losses (e.g., link-layer retransmission (Gast and Loukides, 2002)) or TCP packet inquisitive and loss recovery by intermediate routers (Balakrishnan et al., 1995), or by (c) separating the wireless error- level and wired error-safe transmission paths using an intermediate host (Bakre and Badrinath, 1994; Brown and Singh, 1997). However these approaches have been thoroughly discussed in Lochert et al. (2007).

TCP was originally designed for wired networks, but the increase in wireless networks force to modify TCP protocol. Packet loss is major issue in wireless networks due to short term radio frequency interference with no buffering issue, (while up to date buffer issues mention in End-to-end). In this case reducing congestion window size is nor a proactive approach. Several solutions (Rama Krishnan...
et al., 2001; Gast and Loukides, 2002; Balakrishnan et al., 1995; Bakre and Badrinath, 1994; Brown and Singh, 1997; Zhang et al., 1991; Samaraweera, 1999; Cen et al., 2003; Wei et al., 2006) have been proposed to resolve this issue in coordinating with Lochert et al. (2007). I have discussed various solutions (Table 5) that address several congestion control problems. Although these solutions rest on different assumptions and approaches (see Fig. 1), they have the same objective to create an ideal algorithm for high-speed (e.g., Optical) or large delay (e.g., Satellite) links.

Similarly many works on high speed and long delay network (Baiocchi et al., 2007; Floyd, 2004; Aggarwal et al., 2000; Ha et al., 2006; Belhaj 2008; Kapoor et al., 2004; Wei et al., 2006; Caro et al., 2003) have been proposed to solve different issues regarding slow start, packet loss, delay, stepping the transmission of data packets, RTT fairness, RTT inter fairness, design.

4. Conclusion

In this work we have presented a survey of various approaches to TCP congestion control that do not rely on any obvious

---

### Table 2
Features of tcp modifications/variants that solve the packet reordering problem.

<table>
<thead>
<tr>
<th>TCP variant</th>
<th>Base Year</th>
<th>Update features</th>
<th>Modification Status</th>
<th>Implementation (version)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TD-FR (Bu and Towsley, 2001; Hollot et al., 2001)</td>
<td>Reno 1997</td>
<td>TD-FR</td>
<td>Receiver Exp.</td>
<td>BSD Linux Sim*</td>
</tr>
<tr>
<td>Eifel (Ludwig and Katz 2000; Ludwig and Gurtov, 2005)</td>
<td>NewReno 2000</td>
<td>Differentiate between transmitted and retransmitted data packets.</td>
<td>Sender Std. i3.0F 2.2.10</td>
<td>NS2</td>
</tr>
<tr>
<td>TCP PR (Bohacek et al., 2003)</td>
<td>NewReno 2003</td>
<td>Fine-grained retransmission timeouts, no reaction to DUPACKs.</td>
<td>Sender Receiver Exp.</td>
<td>NS2</td>
</tr>
<tr>
<td>DSACK (Floyd et al., 2000)</td>
<td>SACK 2000</td>
<td>Reporting duplicate segments.</td>
<td>Sender Std.</td>
<td>NS2</td>
</tr>
<tr>
<td>RR-TCP (Zhang et al., 2002; Zhang et al., 2003)</td>
<td>DSACK 2002</td>
<td>Duplicate ACK threshold adaptation</td>
<td>Receiver Exp.</td>
<td>NS2</td>
</tr>
</tbody>
</table>

* Sim = Simulator.

### Table 3
Features of tcp modifications/variants that implement a low-priority data service transfer.

<table>
<thead>
<tr>
<th>TCP variant</th>
<th>Base Year</th>
<th>Update features</th>
<th>Modification Status</th>
<th>Implementation (version)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nice (Venkataramani et al., 2002)</td>
<td>Vegas 2002</td>
<td>Delay threshold as a secondary congestion indicator. Early congestion detection.</td>
<td>Sender Exp.</td>
<td>2.3.15</td>
</tr>
<tr>
<td>LP (Kuzmanovic and Knightly, 2006; Kuzmanovic and Knightly, 2003; Choi and Yoo, 2005; Kuzmanovic and Knightly Les Cottrell)</td>
<td>NewReno 2002</td>
<td></td>
<td>Sender Exp.</td>
<td>&gt; 2.6.18 NS2</td>
</tr>
</tbody>
</table>

### Table 4
Features of tcp modifications/variants that enable resistance to random losses.

<table>
<thead>
<tr>
<th>TCP variant</th>
<th>Base Year</th>
<th>Update features</th>
<th>Modification Status</th>
<th>Implementation (version)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP Westwood (Mascolo et al., 2001; Gerla and Sanadidi, 2007)</td>
<td>NewReno 2001</td>
<td>Estimate of available bandwidth (ACK granularity), FR.</td>
<td>Sender Exp.</td>
<td>NS2</td>
</tr>
<tr>
<td>TCPW-CRB (Wang et al., 2002a)</td>
<td>Westwood 2002</td>
<td>Available bandwidth estimate (combination of ACK and long-term granularity), identifying pre dominant cause of packet loss.</td>
<td>Sender Exp.</td>
<td>NS2</td>
</tr>
<tr>
<td>TCPW-ABSE (Wang et al., 2002b)</td>
<td>CRB 2002</td>
<td>Available bandwidth estimate (continuously varied sampling interval), varied exponential smoothing coefficient</td>
<td>Sender Exp.</td>
<td>NS2</td>
</tr>
<tr>
<td>TCPW BR (Yang et al., 2003)</td>
<td>Westwood 2003</td>
<td>Loss type estimation technique (queuing delay estimation threshold, rate gap threshold), retransmit sign of all outstanding data packets, limiting retransmission timer back off.</td>
<td>Sender Exp.</td>
<td>NS2</td>
</tr>
<tr>
<td>TCPW BBE (Shimonishi et al., 2005)</td>
<td>Westwood 2003</td>
<td>Effective bottleneck buffer capacity estimation, reduction coefficient adaptation, congestion Window boosting.</td>
<td>Sender Exp.</td>
<td>NS2</td>
</tr>
</tbody>
</table>

---

Please cite this article as: Lar S.-u., Liao X. An initiative for a classified bibliography on TCP/IP congestion control. Journal of Network and Computer Applications (2012), http://dx.doi.org/10.1016/j.jnca.2012.04.003
signaling from the focus on the network. The survey emphasized the fact that the research focus has changed with the development of the Internet, from the basic problem of eliminating the congestion collapse phenomenon to problems of using available network resources effectively in different types of environments (wired, wireless, high-speed, long-delay, etc.).

Congestion control has attracted significant attention over the past decade. This paper provides an overview of the TCP congestion control. We have focused and categorized the most important work done in the last two decades. We have extracted core work done and presented in the form of diagrams, which I hope is more convenient for the researcher and students to get the core points of the papers.

### References


### Table 5

Features of tcp modifications/variants aimed at improving efficiency in high-speed or long-delay networks.

<table>
<thead>
<tr>
<th>TCP variant</th>
<th>Base</th>
<th>Year</th>
<th>Update features</th>
<th>Modification</th>
<th>Status</th>
<th>Implementation (version)</th>
<th>Win</th>
<th>Linux</th>
<th>Sim</th>
</tr>
</thead>
<tbody>
<tr>
<td>HS-TCP</td>
<td>NewReno</td>
<td>2003</td>
<td>Additive increase steps and multiplicative decrease factors as functions of the congestion window size, limited SS.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>&gt; 2.6.13</td>
<td>NS2</td>
<td></td>
</tr>
<tr>
<td>STCP</td>
<td>NewReno</td>
<td>2003</td>
<td>MIMD congestion avoidance policy.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>&gt; 2.6.13</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>H-TCP</td>
<td>NewReno</td>
<td>2004</td>
<td>Cwnd increase steps as a Function of time elapsed since the last packet Loss detection, scaling increase step to a reference RTT, multiplicative decrease coefficient adaptation.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>&gt; 2.6.13</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>TCP Hybla</td>
<td>NewReno</td>
<td>2004</td>
<td>Scaling the increase steps in SS and CA to the reference RTT, Data packet pacing, initial slow start estimation.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>&gt; 2.6.13</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>YeAH-TCP</td>
<td>HS-TCP</td>
<td>2004</td>
<td>Binary cwnd search, limited SS.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>&gt; 2.6.12</td>
<td>NS2.6</td>
<td></td>
</tr>
<tr>
<td>TCPW-A</td>
<td>Westwood</td>
<td>2005</td>
<td>Agile probing, persistent non-congestion detection.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>NS2</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>Log Westwood</td>
<td>Westwood</td>
<td>2008</td>
<td>Logarithmic cwnd increase.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>NS2</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>TCP cubic</td>
<td>BIC</td>
<td>2008</td>
<td>The cwnd control as a cubic function of time elapsed since a last congestion event.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>&gt; 2.6.16</td>
<td>NS2.6</td>
<td></td>
</tr>
<tr>
<td>FAST TCP</td>
<td>Vegas</td>
<td>2003</td>
<td>Constant-rate cwnd equation-based update.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>NS2.9</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>TCP Libra</td>
<td>NewReno</td>
<td>2005</td>
<td>Adaptation of the packet pairs to estimate the Bottleneck link capacity, scale the cwnd increase step by the bottleneck link capacity and queuing delay.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>NS2</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>TCP New Vegas</td>
<td>Vegas</td>
<td>2005</td>
<td>Rapid window convergence, packet pacing, packet pairing.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>TCP AR</td>
<td>Westwood</td>
<td>2005</td>
<td>Cwnd increase steps as a function of the achievable rate and queuing delay estimates.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>TCP fusion</td>
<td>Vegas</td>
<td>2007</td>
<td>Cwnd increase steps as a function of the achievable rate and queuing delay estimates.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>TCP Africa</td>
<td>HS-TCP</td>
<td>2005</td>
<td>Switching between fast (HS-TCP) and slow (New Reno) mode depending on the Vegas-type network state estimation.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>NS2</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>Compound TCP</td>
<td>HS-TCP</td>
<td>2005</td>
<td>Two components (slow and scalable) in the cwnd calculation.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>Vista, S08, XP, S03</td>
<td>2.6.14–2.6.25</td>
<td></td>
</tr>
<tr>
<td>TCP Illinois</td>
<td>Newly</td>
<td>2006</td>
<td>AIMD factors as functions of the queuing delay.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>&gt; 2.6.22</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>YeAH TCP</td>
<td>STCP</td>
<td>2007</td>
<td>Switching b/w fast (STCP) and slow (New Reno) mode depending on a combined Vegas-type and DUAL-type estimate, precautionary decongestion.</td>
<td>Sender</td>
<td>Exp.</td>
<td>–</td>
<td>&gt; 2.6.22</td>
<td>–</td>
<td></td>
</tr>
</tbody>
</table>


Please cite this article as: Lar S-, Liao X. An initiative for a classified bibliography on TCP/IP congestion control. Journal of Network and Computer Applications (2012), http://dx.doi.org/10.1016/j.jnca.2012.04.003