Internet Draft
draft-fpeng-wecn-04.txt

Fei Peng
Beijing University of
posts and telecomm.
Jian Ma
Nokia Research Center

June 2001

An Effective way for Enhancement of TCP Performance in Wireless and Mobile Networks

Status of this Memo

1. Abstract

TCP congestion control has been developed on the assumption that congestion in the network to be the only cause for packet loss. Thus, it drops its transmit window upon detecting a packet loss. In the presence of high error rates and intermittent connective characteristic of wireless link, these results in an unnecessary reduction in link bandwidth utilization for packet losses are not mainly due to congestion.

This paper provides an effective way to improve TCP performance in wireless and mobile networks. Current ECN proposal used in traditional wired networks with the purpose of avoiding unnecessary packet losses can obtain separation of loss recovery from congestion control if packet losses are disabled to initiate source window reduction. However, to implement ECN, complex and inconsistent buffer management in the networks is required; in the cases where ECN-Capability is falsely indicated or the ECN congestion

Notification is erased, the congested router's queue could continue to build, resulting in packet loss at the congested router. A flow that is intentionally avoiding end-to-end congestion control by packet losses at the end nodes can avoid end-to-end congestion control even when the congested queue is in packet-dropping mode.

By refusing to reduce its sending rate in response to packet drops in the network, the flow experience higher queuing delays, and may causes very severe problem on other flows. In this draft, we propose to add Wireless-ECN just upon the first packet loss occurs. In this way, it strongly achieves separation of congestion control and loss recovery mechanism by quickly informing the sender that a loss happened because of reasons related to network congestion. The unnecessary window reduction caused by lost packets due to link errors is avoided. When there is falsely indicating ECN-Capability or erasing the ECN congestion indication (in the CE-bit) in wireless and mobile networks, the addition of Wireless-ECN can represent congestion losses to initiate window reduction in time and improve ECN performance. Specially, to prevent unnecessary packet losses due to buffer overflow, it is most recommended to cooperate WECN with ECN or a better congestion avoidance mechanism to improve TCP performance in wireless and mobile networks.

2. Conventions used in this document

"WECN" indicates Wireless-ECN. "WECN-Echo flag" indicates Wireless-ECN-Echo flag.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [ ].

3. Introduction

Congestion control is one of the key mechanisms to accommodate the increasingly diverse range of services and types of traffic in the Internet. Initially the Internet was intended to support best effort service, and the TCP congestion control method that was actually implemented has been developed on the assumption that the network would be treated as a black box. This means that the end nodes do not exercise control by directly ascertaining the state of routers and transmission line, but rather regulate the traffic by inferring the network load indirectly from packet loss and response time fluctuations. In wired networks, this may not induce serious problems as packet losses are mainly due to congestion. However, in the presence of high error rates and intermittent connective characteristic of wireless links, the reliance on packet drops as indication of congestion causes a significant degradation in TCP performance, for TCP reacts to packet loss as it would in the wired environment: it drops its retransmit window size before retransmitting packets, initiates congestion control or avoidance mechanisms and resets its retransmission timer, thereby result in an unnecessary

reduction in the link bandwidth utilization, which causing poor throughput and very high interactive delays.

Recently, several schemes have been proposed to alleviate the effects of non-congestion-related losses on TCP performance over networks that have wireless or similar high-loss links.
One of the researches concerns on improvement of transport underlying protocol stacks. For example, there have been several proposals for reliable link-layer protocols as forward error correction (FEC) and retransmission of lost packets in response to automatic repeat request (ARQ) messages. However, it is the main worry about link-layer protocols that an adverse effect on certain transport-layer protocols such as TCP is very possible. Another solution of network layer protocol called Snoop protocol is proposed to cache packets at the base station and perform local retransmissions across the wireless link. Like link-layer solutions, the snoop approach could also suffer from not being able to completely shield the sender from wireless losses.

Several schemes modified at transport layer have been proposed to alleviate the effects of non-congestion-related losses on transport performance. The indirect-TCP is one of the first protocols to distinguish different losses by splitting a TCP connection between a fixed and mobile host into two separate connections at the base station, a more optimized wireless link-specific protocol tuned for better performance can be used over a one-hop wireless link. The advantage of the split connection approach is that it achieves a separation of flow and congestion control of the wireless link from that of the fixed network. However, there are some drawbacks of this approach such as loss of semantics, application re-linking and software overhead, etc. And there is no need to sacrifice the semantics of acknowledgments in order to achieve possible good performance.

As lost packets can be simply divided into congestion-related losses and non-congestion-related losses, an ELN protocol can be used to differentiate the packet loss by adding an explicit loss notification option to TCP acknowledgments when a packet is dropped on the wireless link. Future cumulative acknowledgments corresponding to each lost packet must be always marked to identify that a non-congestion-related loss has occurred, then the sender may perform retransmissions without invoking the associated congestion-control procedures. In practice, this algorithm brings burden to the implementation nodes since judgment is required for each dropped packets and each lost packet due to transmission errors needs marking otherwise it will invoke congestion control. Additionally, it might be difficult to identify which packets are lost due to errors on lossy link, for example, it may be hard to determine the connection that a corrupted packet belongs to since the header could itself be corrupted.

This paper provides a mechanism called Wireless-ECN (WECN) used in wireless and mobile networks. Like ELN, it is a mechanism by which the reason for the loss of a packet can be communicated to the TCP sender. In particular, it provides a way by which senders can be informed that a loss happened because of reasons related to network congestion, so that sender congestion control can be decoupled from retransmissions. Since it is no need to reduce the congestion window more than once in a round trip time, the burden to the implementation is greatly released for judgment is not required for each dropped packets due to congestion. Additionally, it is very easy to identify which packets are lost due to buffer overflow.

4. Motivation

Currently, a scheme concerning adding an explicit congestion notification (ECN) to TCP acknowledgment option through detecting incipient congestion in network is proposed by IETF. Upon receiving this ECN message, the sender reduces its congestion window whenever loss occurs. With the purpose of preventing unnecessary packet drops, TCP can base its congestion control not solely on packet losses. However, when ECN mechanism is being used, it makes the sense to add at the same time mechanisms to monitor the average queue size. Many current networks with routers whose main function is to route packets to the networks have no provision for the detection of incipient congestion before the queue overflows. Even with this, during periods where the average queue size exceeds an upper threshold, packet-marking rate can not catch up with its passing rate, and therefore routers drop packets rather than set the ECN in packet header. It may be a deterrent in such case if packet losses are completely kept from initiating congestion control. In spite of all these difficulties, ECN is likely to be adopted gradually, So accommodating migration is essential and it is also important to provide a method to apply ECN to wireless and mobile environment.

In the following section, we provide a mechanism called Wireless-ECN (WECN) used in wireless and mobile networks. It provides a way by which senders can be informed that a loss happened because of reasons related to network congestion, so that sender congestion control can be decoupled from retransmissions. Since it is arranged to mark a data packet with an explicit wireless congestion notification (WECN) immediately after a packet loss has occurred due to a buffer overflow, it ensures that Wireless-ECN notification can be added without worrying about slow marking rate for dropping packets give the buffer time to vacate some space to hold and transmit new data. With Wireless-ECN, the complexity and inconsistency of buffer management and more conservative TCP congestion control over a short-scale period in ECN mechanism is not necessarily needed. When apply ECN into wireless and mobile networks, it is not required that packet losses to initiate congestion control. WECN at this time can represents congestion losses to invoke window reduction in time and improve ECN performance. So, Wireless-ECN provides a simple and effective way to cooperate ECN into wireless environment.

In particular, considering WECN message is set not later than threshold duplicate ACKs, it is hoped WECN would provide a more timely reaction to loss retransmission than reactions based on drop detection via duplicate ACKs or timeout.

5. Assumptions

In this section, we describe some of the important assumptions that guided the design choices in this proposal.

(1) It is one of the premises of WECN mechanism that all the end nodes in the experiment are WECN capable. Otherwise, congestion control would be invoked in response to packet loss whenever it is due to congestion or transmit errors.
(2) It is assumed that each node in the network can add WECN bit in the packets to indicate congestion once packet losses occur.
(3) We also assume that the transport is capable and willing to participate in WECN in each transmitting packet.
(4) With the above assumption, implementing WECN need not having to resort to "islands" of WECN-capable and non-WECN-capable environments.
(5) Asymmetric routing is likely to be a normal occurrence in the Internet. The path (sequence of links and routers) followed by the acknowledgment packets in the reverse direction.

6. Wireless Explicit Congestion Notification in IP

Upon the first lost packet due to buffer overflow in the intermediate router, Wireless-ECN notification is added in the header of following passing packet for dropping packets give the buffer time to vacate some space to hold and transmit new data. This ensures that the Wireless-ECN notification can inevitably be added even with multiple packet losses and no need of mechanisms in the router to monitor the average queue size. The request for the Wireless-ECN to be added as quickly as possible after a congestion loss occurs is to make certain to alleviate network congestion in time.

It is requested that each node in the network can add Wireless-ECN signal to indicate congestion once a packet loss occurs. If multiple packet losses occur in a congestion window, multiple Wireless-ECN messages are to be set in a node. It is also allowed to set Wireless-ECN bit in the following packets belonging to the same congestion window in the successive passing routers. As it must take a round-trip time for the first added Wireless-ECN to reach to the sender, and also for the condition and buffer occupancy in each network node varies greatly, not only one Wireless-ECN are set in a transmission window during a round-trip time. Some Wireless-ECN set may belong to different transmitting network nodes and some may belong to the same node. Moreover, once the Wireless-ECN is added in one router, the following routers may not erase the Wireless-ECN bit in arriving packets to maintain the Wireless-ECN compliance in the network. As TCP receiver must set Wireless-ECN bit in TCP header in response to each IP packet with Wireless-ECN bit added, the TCP source, however, may receive multiple Wireless-ECN packets in a congestion window. Since the main function of Wireless-ECN is to reduce the congestion window only once in a transmission window in the event of network congestion, this multiple Wireless-ECN messages also provide robustness against the possibility of a dropped Wireless-ECN packet in the bi-directional transmitting paths.

To comply with current proposed ECN mechanism in IETF draft, we obey the suggestion of keeping congestion experienced information in the regular headers of an IP packet since processing the regular headers in IP packets is more efficient than processing the header information in IP options and denote it as W-CE bit to differentiate that of CE bit used for ECN. According to this, we use 'W-CE packet' to indicate a packet that has the W-CE bit set. Since the Wireless-ECN-Echo flag set in the TCP header which will directly cause window reduction at the source side is determined by W-CE bit set in the IP header, to initiate reduction of window size as ECN does, we choose bit 6 and bit 7 that are designated as the ECT bit an the CE bit of ECN packet in the IPv4 TOS octet to indicate the congestion packet losses in the intermediate routers. So, the use of Wireless-ECN field as specified in this document is the same as traditional ECN, but it provides function of indicating packet losses due to congestion.

7. Congestion Control Support from the Transport Protocol

7.1. TCP initialization

In the TCP connection setup phase, the source and destination TCPs exchange information about their desire and/or capability to use WECN. Subsequent to the completion of this negotiation indicates to the network that the transport is capable and willing to participate in WECN for each transmitting packet.

When a node sends a TCP SYN packet, it may set the WECN-Echo flag in the TCP header as an indication that the sending TCP is ECN-Capable, rather than as an indication of congestion or of response to congestion. More precisely, a SYN packet with WECN-Echo flags set indicates that the TCP implementation transmitting the SYN packet will participate in WECN as both a sender and receiver.

When a node sends a SYN-ACK packet, it may set WECN-Echo flag, the sending TCP correctly interprets a receiver's reflection of it own flags in the Reserved field as an indication that the receiver is ECN capable.

The WECN-Echo flag is at the same field location as ECN-Echo flag in TCP header.

7.2. The TCP Sender

Though our proposed Wireless-ECN mechanism can not avoid unnecessary packet drops for short or delay-sensitive TCP connections as ECN can, it has the effect on independence of TCP of packet loss as the indication of congestion. As TCP use Wireless-ECN message to invoke congestion control, whenever a packet loss occurs, only retransmission mechanism is used at that time which does not affect TCP data flow rate. This is very useful in wireless and mobile networks for the network itself has the function of distinguishing different source of packet losses. By informing the sender with Wireless-ECN packets when the congestion loss occurs, packet losses due to transmission errors could not reduce TCP flow control window, which results in significant performance improvement. The TCP congestion window can only be reduced by Wireless-ECN message that is set due to congestion losses.

The additional requirement for the TCP source to react to multiple Wireless-ECN packets is that it should not repeat the reduction of the congestion window; since the packet was probably dropped before the source reduced its window in response to the WECN. TCP should react to a WECN at most once per round-trip time. The sender would react to the subsequent WECN only after the outstanding data before the sender entered into loss recovery phase upon receiving the first W-CE bit have all been acknowledged. Like ECN, another change for TCP is a negotiation phase during setup to determine if both end nodes are WECN-capable.

TCP follows existing algorithms for sending data packets in response to incoming ACKs. Multiple duplicate acknowledgements or retransmit timeouts only in charge of loss retransmission and are separated from window reduction.

7.3. The TCP Receiver

When TCP receives a W-CE data packet at the destination end-system, the TCP data receiver sets the ECN-Echo flag in the TCP header of the subsequent ACK packet. The ECN-Echo flag in the reserved field of the TCP header are designated as Wireless-ECN-Echo flag since WECN uses the same flag as ECN. If any of the received data packets are W-CE packets, then the returning ACK has the ECN-Echo flag set. As only one Wireless-ECN is required to be added in a router, the frequency of Wireless-ECN to be set in a congestion window will not exceed the number of total network nodes where packets must travel across. This could relieve the burden of providing interface facility between IP and TCP at the receiver side.

To provide robustness against the possibility of a dropped ACK packet carrying a Wireless-ECN-Echo flag, the TCP receiver must set the WECN-Echo flag in a series of ACK packets. The TCP receiver uses the CWR (Congestion Window Reduced) flag as in ECN of reserved field of TCP header to determine when to stop setting the WECN-Echo flag.

8. Retransmission Support from the Transport Protocol

In contrast to Fast Retransmit algorithm in Reno, Tahoe TCP uses retransmit timers to detect lost packets All transmitted data is acknowledged by the receiver, and a lost packet is signaled by the expiration of a timer at the transmitter even with the arrival of duplicate acknowledgements Mutiple lost packets in a congestion window are recovered by multiple backoff of Retransmission Timeout Instead of modifying TCP behavior of detecting packet losses basing on timeout, WECN quicken the recovery phase of lost papckets by reducing the backoff of the timeout value in our simulator, that is, if a WECN message is received in a congestion window, the Retransmit Timeout set for the following packet losses in the same window will not be doubled.

We then discuss the implications of WECN for Reno and New-Reno TCP versions.

With our Wireless-ECN consideration, TCP react to packet loss without reduction of transmission window. When three duplicate ACKs inform the sender about packet loss, only retransmission is invoked. And since the sender response only the first ECN in a widow, the congestion window reduces once even with multiple packets lost in a window. So, Reno TCP with Wireless-ECN need not afraid to wait for a retransmit time-out when multiple packets are dropped from a single window of data. High TCP performance would be obtained since the congestion window is only reduced once while retransmitting lost packets. Large congestion window also ensures the arrival of adequate threshold duplicate ACKs to trigger lost packet retransmission.

The New-Reno TCP includes a small change to the Reno algorithm at the sender that eliminate the wait of Reno for a retransmit timer when multiple packets are lost from a window because it also only reduce congestion window once during loss recovery phase. To do this, partial ACKs do not take TCP out of Fast Recovery in New-Reno as Reno formally does, while WECN in Reno maintain congestion window by not response to the following WECNs in a window. So, New-Reno TCP with our WECN mechanism includes a small change to the New-Reno algorithm at the sender that eliminates the reliance of New-Reno on the threshold duplicate ACKs to invoke loss recovery. If the ACK has the WECN bit set, it can ensure that the packet with the sequence number in ACK has been lost. Using WECN to notify the reduction of congestion window can avoid complex treatment of partial ACKs in New-

Reno.

9. Cooperate ECN into Wireless networks

Current ECN mechanism is proposed in traditional wired environment to alleviate network congestion. Because of complexity of network conditions and the nature of feedback congestion control mechanism, congestion losses will not be completely averted. So, there exist a wide range of ECN conditions such as an ECN followed by another ECN, a Fast Retransmit, or a Retransmit Timeout; a Retransmit Timeout or a Fast Retransmit followed by an ECN. So, it is also required that TCP follows existing algorithm for sending data packets and reducing congestion window in response to incoming ACKs, multiple duplicate acknowledgements, or retransmit timouts when ECN is used.

To put ECN into wireless and mobile networks, we have to suggest that TCP congestion window reduction can not be initiated by packet losses. The dropping packets due to congestion may be an adequate deterrent factor affecting network performance if congestion control is not initiated by packet losses due to buffer overflow in time.

With the addition of Wireless-ECN, when some packet losses due to congestion come first before ECN messages arrive in a round-trip time, the Wireless-ECN signals added just upon the first packet loss caused by congestion will come in time to initiate general congestion control. And also, TCP only reacts to once and generally the earliest one of Wireless-ECN or ECN ACK every round-trip time. The problem of unnecessary reduction of window size by packet losse irrelavent to congestion is solved for only wireless-ECN or ECN is interpreted by the source TCP as a new instance of congestion.

10. Security Considerations

One disadvantages of Wireless-ECN mechanism is that Wireless-ECN message could be dropped by the network before reaching the TCP source. If the assumed source of data loss is not congestion, loss of Wireless-ECN will result in the failure of congestion control for that flow and a resulting increase in congestion in the network, ultimately resulting in subsequent packets dropped for that flow as the average queue size increased at the congested gateway. Since Wireless-ECN is added upon packet losses caused by congestion, subsequent packets dropped in the next congestion window may lead to another successive try to add WECN? With the possibility of additionally setting Wireless-ECN bit because of the previous lost WECN, it does not show performance problems from dropped Wireless-ECN message. In addition, the number of dropped WECN message should be small in the network since multiple WECN messages set immediately following each dropped packet provide robustness against the possibility of a dropped WECN packet in the bi-directional transmitting paths. In addition, dropping packets give the buffer time to vacate some space to hold and slow down data transmission, this ensures that the Wireless-ECN notification can be added even with multiple packet losses for transmitting rate would not exceed marking rate. The compliance of WECN in the network that ensures that once WECN is added in one router, the following routers may not erase the WECN bit in arriving packets also implicit the security of WECN.

In spite of all these properties, we still give the security mechanism of WECN, we set a threshold to indicate when WECN should not be added, that is, when the buffer occupancy reduced below the threshold and there is no packet losses due to buffer overflow occurs, WECN bit is kept from setting otherwise it is added continuously following the first WECN set immediately after buffer overflow. Considering WECN leads at least half reduction of congestion window size, we assume the threshold needed will not be lower than fifty percent of buffer occupancy and it is better for the threshold set not higher than eighty percent of buffer size to ensure enough WECN generated. Since the threshold set for WECN is to determine when to stop adding it, it has little relationship with buffer management and is simple to implement. Additionally, WECN can also use the existing threshold set for ECN or other congestion avoidance mechanism if these mechanism are supported in the routers at the moment.

11. Support from ATM networks

The investigation of WECN in this paper concerns only TCP and IP networks, we now consider a scenario of TCP traffic where part of or all of the path might consist of ATM networks. For ECN, the threshold set in ATM switch buffer to indicate congestion before buffer overflow can not get accessible way to inform the TCP connections of congestion, the only viable mechanism is for the ATM network to drop TCP or IP packets, either inside or at the boundaries of the ATM network. At the boundary of the ATM network where frame segmentation and reassembly occur, IP level of WECN mechanism could immediately add WECN to indicate congestion upon dropped TCP packet and invoke WECN mechanism for packets from WECN source.

12. Conclusions and Future Works

With the incremental deployment of IETF proposed ECN in both end systems and in routers, there exist many inconsistencies in the network and end-systems. For example, some routers may drop packets using the same RED policies for congestion detection while other routers set the CE bit for equivalent level of congestion. Some routers update its window once every two round-trip times while other sources increase the congestion window additional every round-trip time after equivalent threshold packets in the last window had the ECN bit set.

Since active queue management is another complex research area in IETF, we change the mechanism of setting CE bit basing on the average queue length exceeding a threshold and propose to add Wireless-ECN just upon the first packet loss due to buffer overflow occurs. By doing this, inconsistent threshold judgment may not happen and no RED algorithm is required in the routers. The reduction of congestion window upon receiving the first Wireless-ECN prevents the discordance of different number of threshold ECNs chosen by end-systems with the IETF ECN usage. With all these consistency in the networks and end-systems, our mechanism not only improves TCP performance in wireless and mobile networks but provide a compatible way to enhance ECN performance across wireless paths as well for it realizes distinguish different source of packet losses.

The simulation in our project shows that WECN mechanism would give even higher throughput than ECN and ECN with WECN scheme because it hasten the rate of loss recovery phase which mainly due to congestion. Though in our simulation, we only utilize it in Tahoe TCP, it may obtain advantages if apply WECN in Reno TCP and further to New Reno TCP.

In other cases, though Wireless-ECN achieve quick retransmit lost packets mainly due to buffer overflow, sometimes the retransmission is invoked by non-congestion related reasons and because the loss of WECN though rarely occur, to find a better retransmission algorithm is also important for the drawbacks in the current Fast Retransmission and Recovery algorithm.
To further analyze the benefits of Wireless-ECN for TCP and for ECN, continuous simulations must execute in the future works, which contains congestion in both directions, two-way traffic performance, and with multiple congested gateways.

13. Acknowledgments

We would like to appreciate Beijing University of posts and telecommunicaitons for offering great advocacy of our research on this mechanism. We also thank Nokia corporation for supporting this idea with applying of a patent. And I (first author) will particularly mention my professor (Mrs. Cheng Shiduan) for the encouragement of my work.

14. References

[1] FEI, P., JIAN, M. "Overload control method for a packet-switched network", Patent Application NC 18254, June 1999.
[2] FEI, P., Jian, M., etc. "TCP Performance Enhancements in Wireless and Mobile Networks", International Conference on Communication Technology (ICCT'98), Oct,1998, pp.S46-07-1:S46-07-5.
[3] K. K. Ramakrishnan, Sally Floyd, "A proposal to add Explicit Congestion Notification (ECN) to IP", Internet Draft-kksjf-ecn-03.txt, Oct, 1998.
[4] Sally Floyd, "TCP and Explicit Congestion Notification", Computer

Communication Review, V.24 N.5, October 1994, p.10-23.

[5] Sally Floyd, David Black, K.K.Ramakrishnan.,"IPsec Interactions with ECN", draft-ietf-ecn-01.txt. December 1999.

15. Author's Addresses

Fei Peng
Beijing University of Posts and Telecomm.
Phone: 010-62282007
Email: fpeng_01@hotmail.com