Implementation and Evaluation of a New TCP Loss Recovery Architecture

2.1 Two Lists Structure

We propose a new loss recovery algorithm based on a new data structure - **two packet transmission order lists**. The important difference between our proposal and the existing algorithms is the basic data structure. The data structure of the previous algorithms is sequential number based packet list which has difficulty remembering packet transmission order. However, our proposal is able to reflect the packet transmission order.

As shown in Figure 1(b), LE manages two lists, **WAITLIST** and **RTXLIST**. When a packet is newly sent or resent, it is inserted at the end of WAITLIST to record the transmission order. Therefore, the list naturally represents all of the currently outstanding packets. When a packet is determined to be lost, the packet is transferred to the end of RTXLIST. Each entry in the lists contains three variables dupCnt, timeStamp, and seqNum.

2.2 New Loss Recovery Algorithms

**Per Packet Acking Process:** Whenever an ACK, including time stamp and SACK options, arrives, the timeStamp in the packet is compared to the time stamp of the ACK. If the timeStamp is less than the time stamp of the ACK and the seqNum is less than or equal to the cumulative number of the ACK, the packet is acked. If the timeStamp is equal to the time stamp of the ACK, SACK blocks are used for acking the packet. If the packet is not acked, the dupCnt increases by one. If dupCnt reaches the dupThresh (usually 3), the packet is considered as a loss and is moved to the end of RTXLIST for retransmission. If the timeStamp is larger