

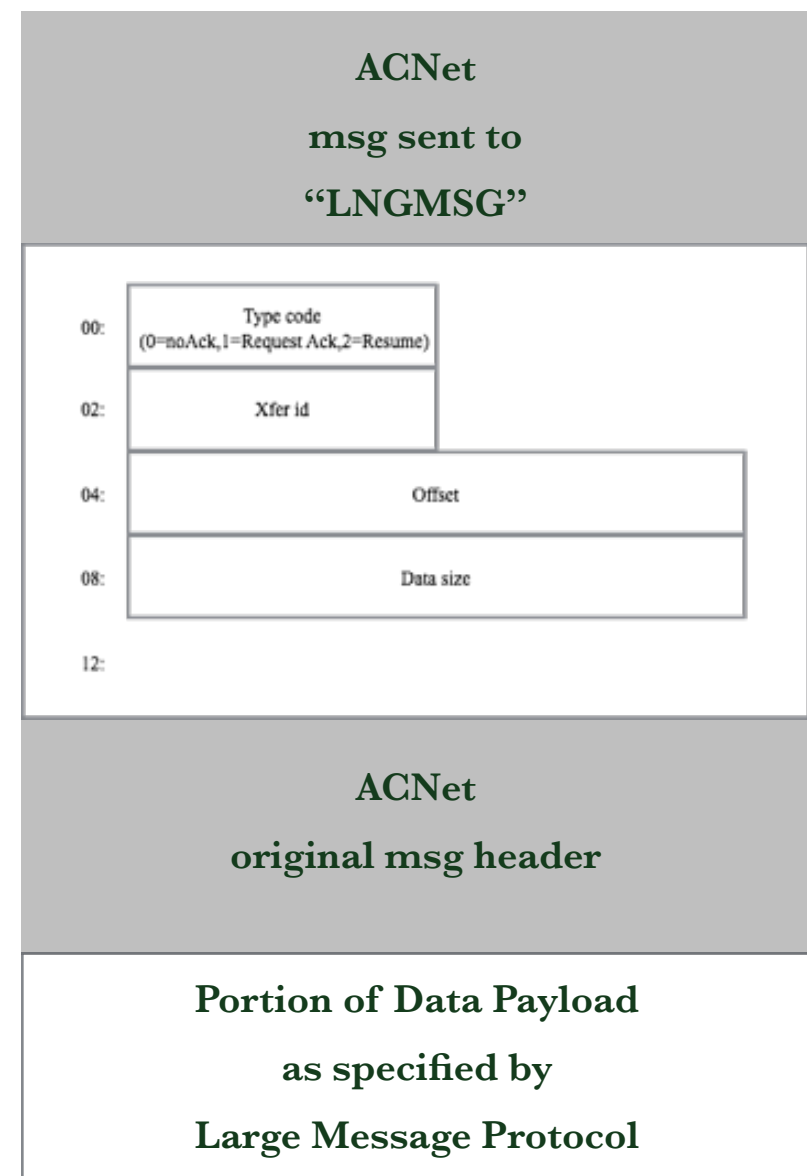
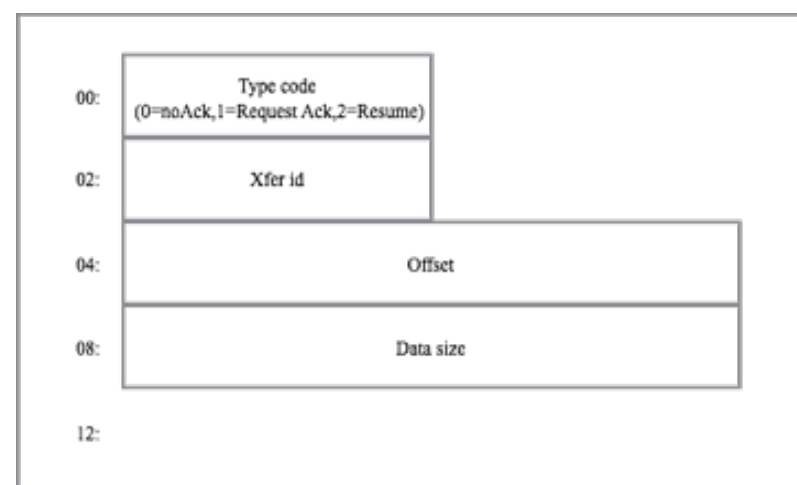
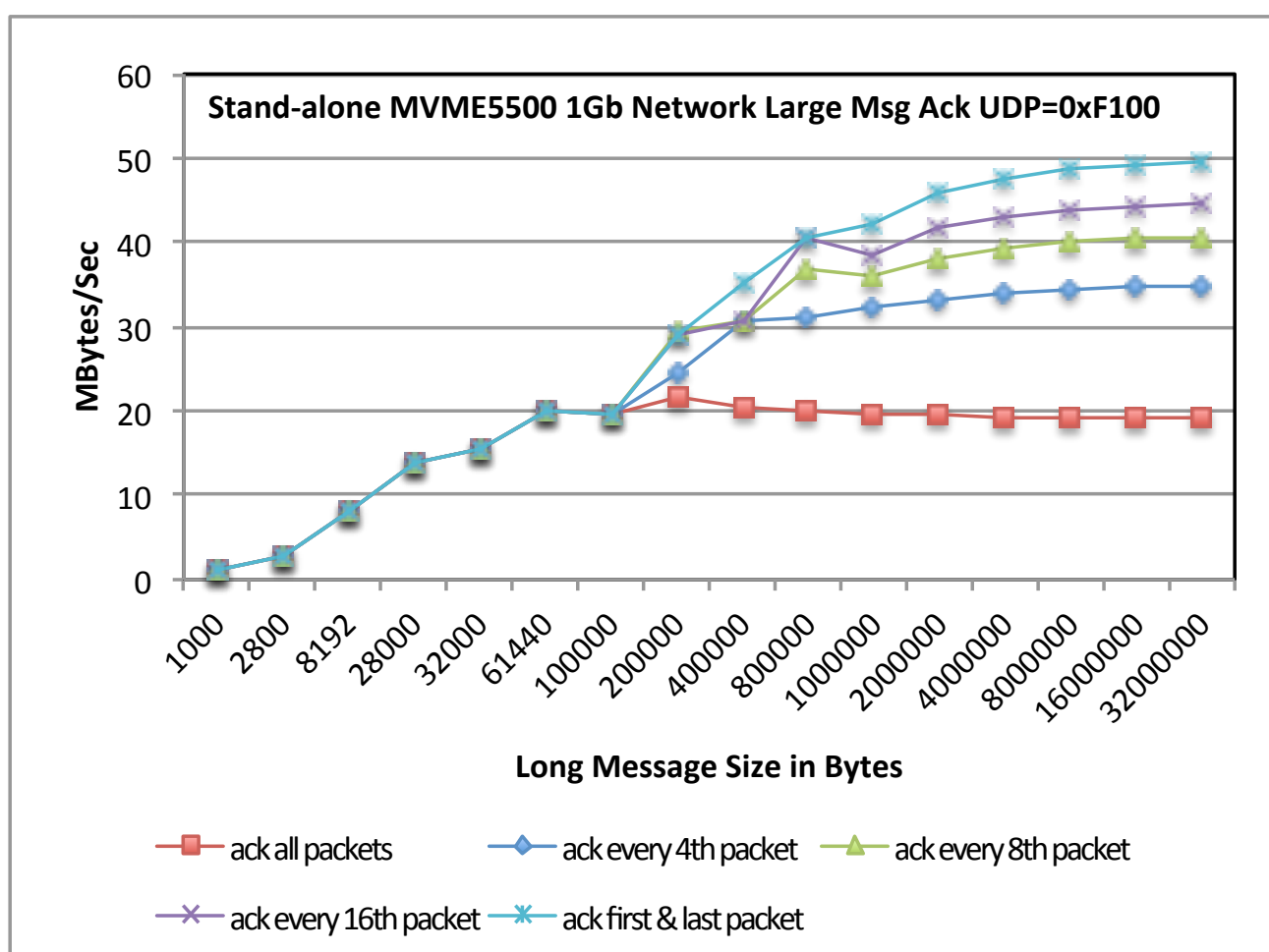
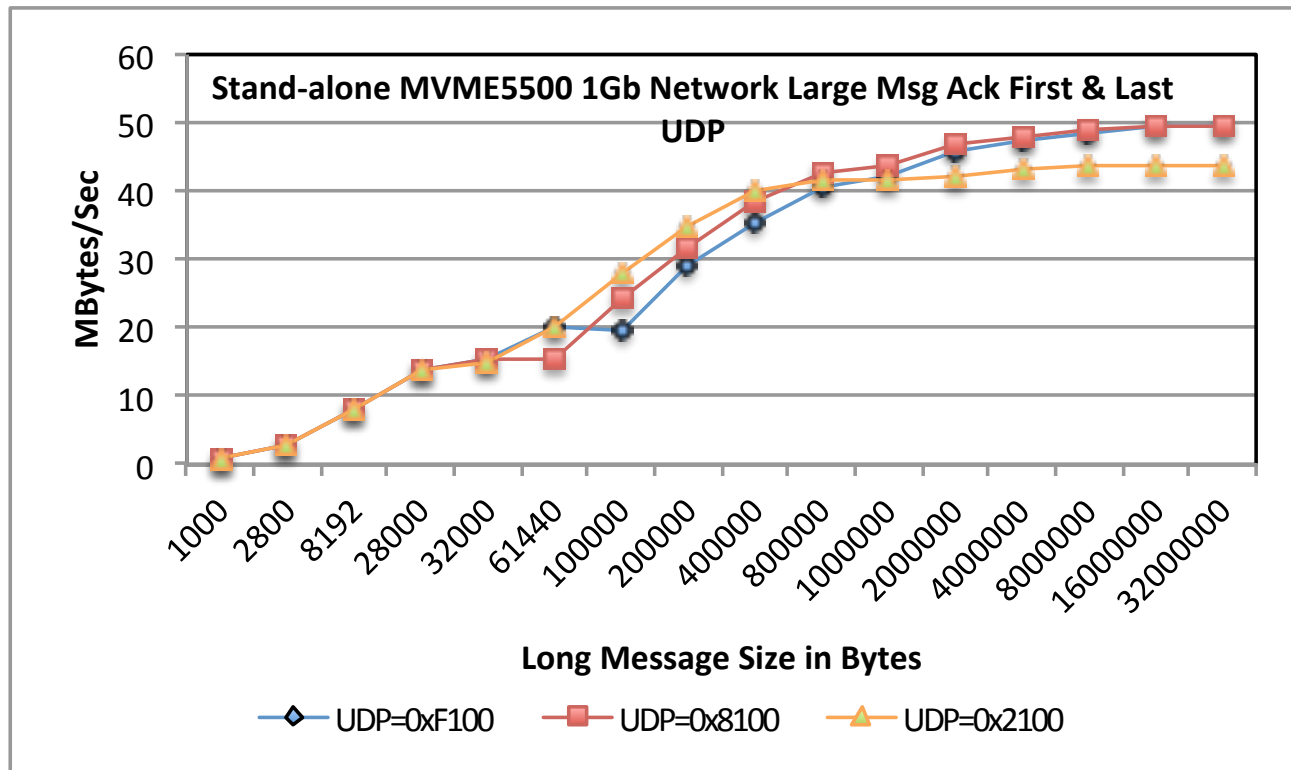
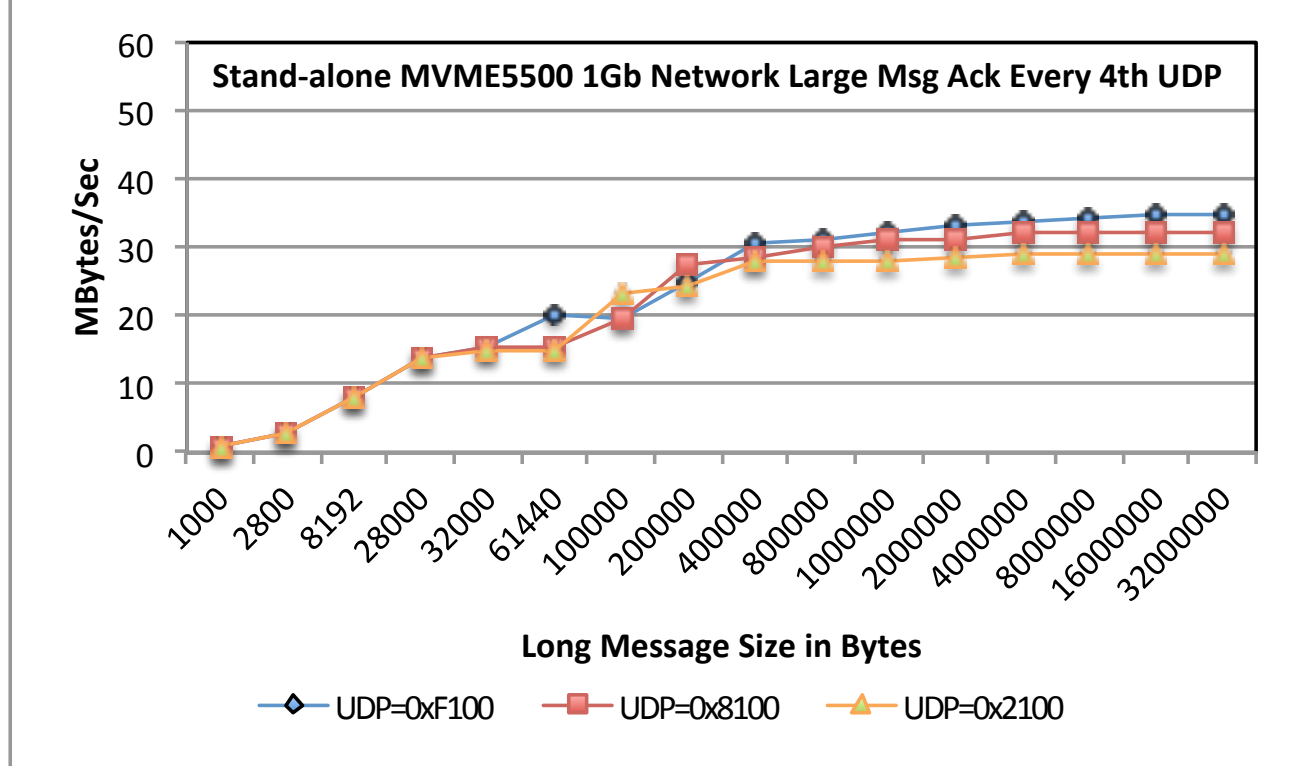
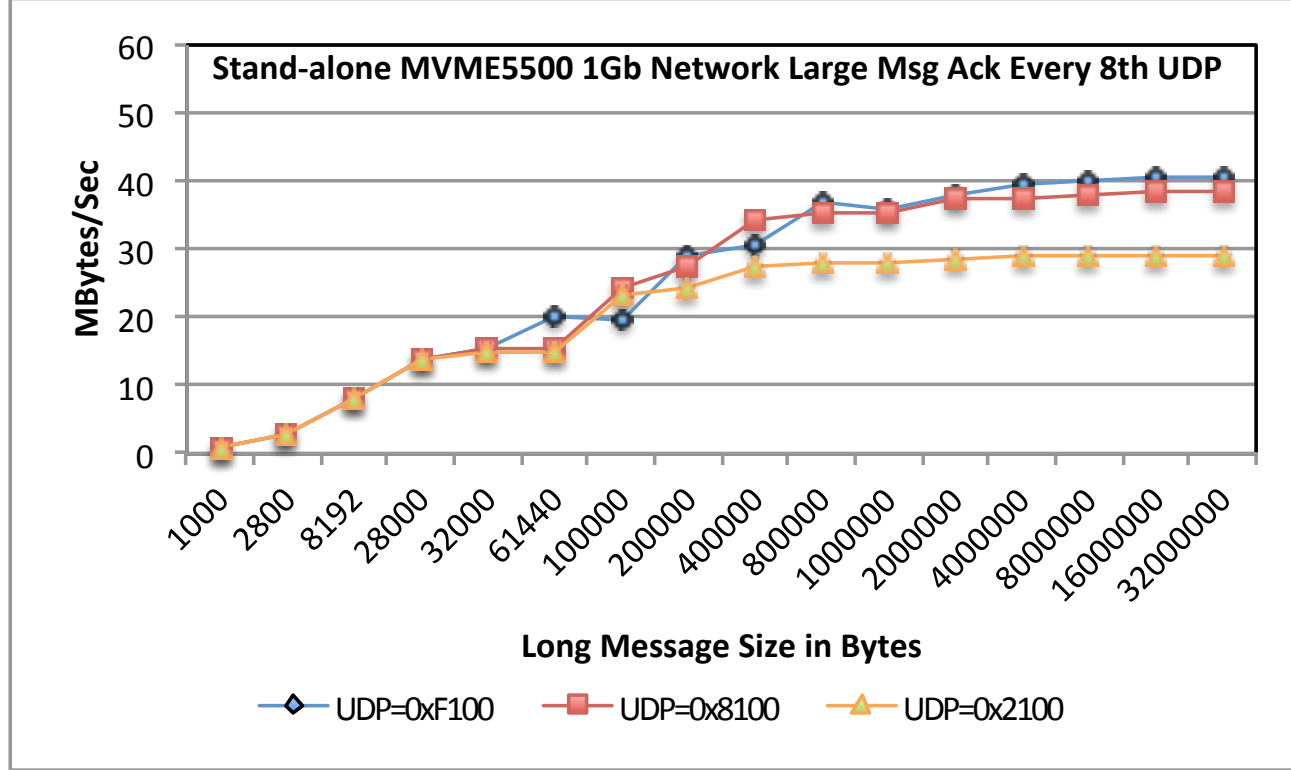
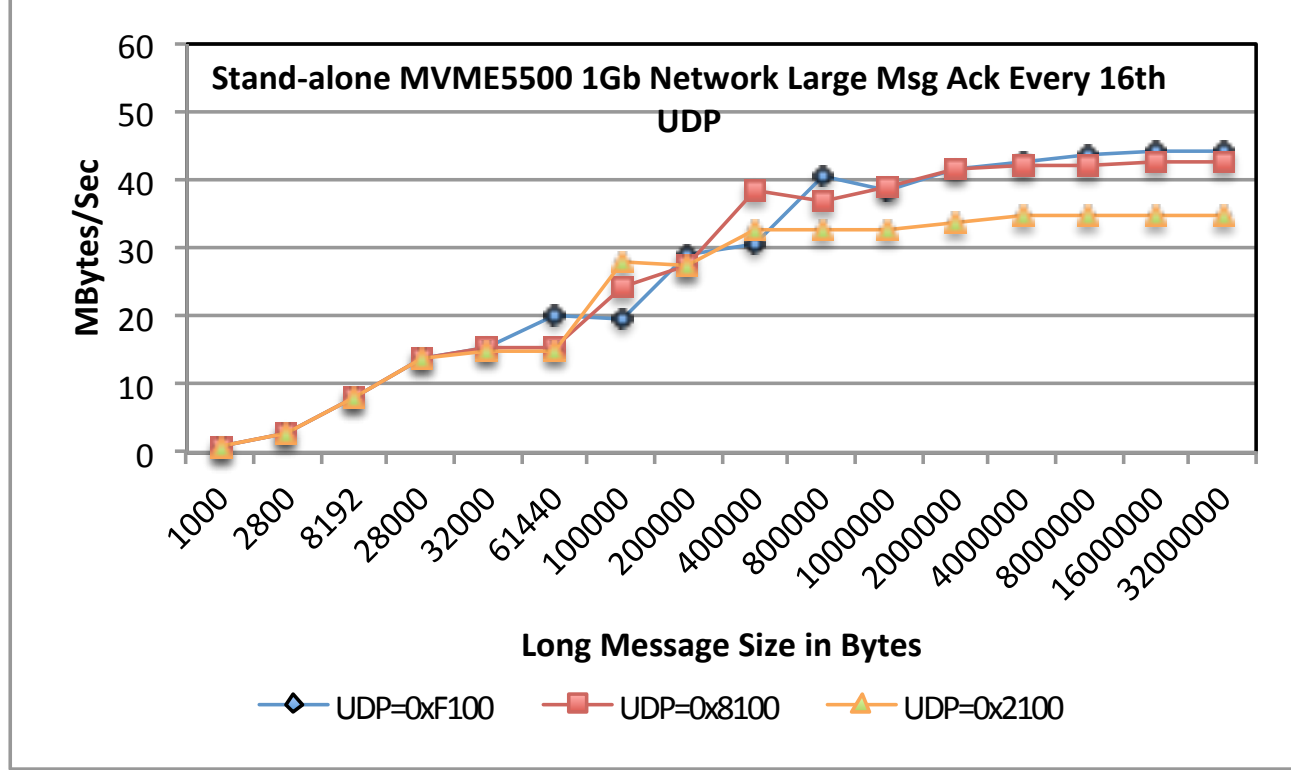
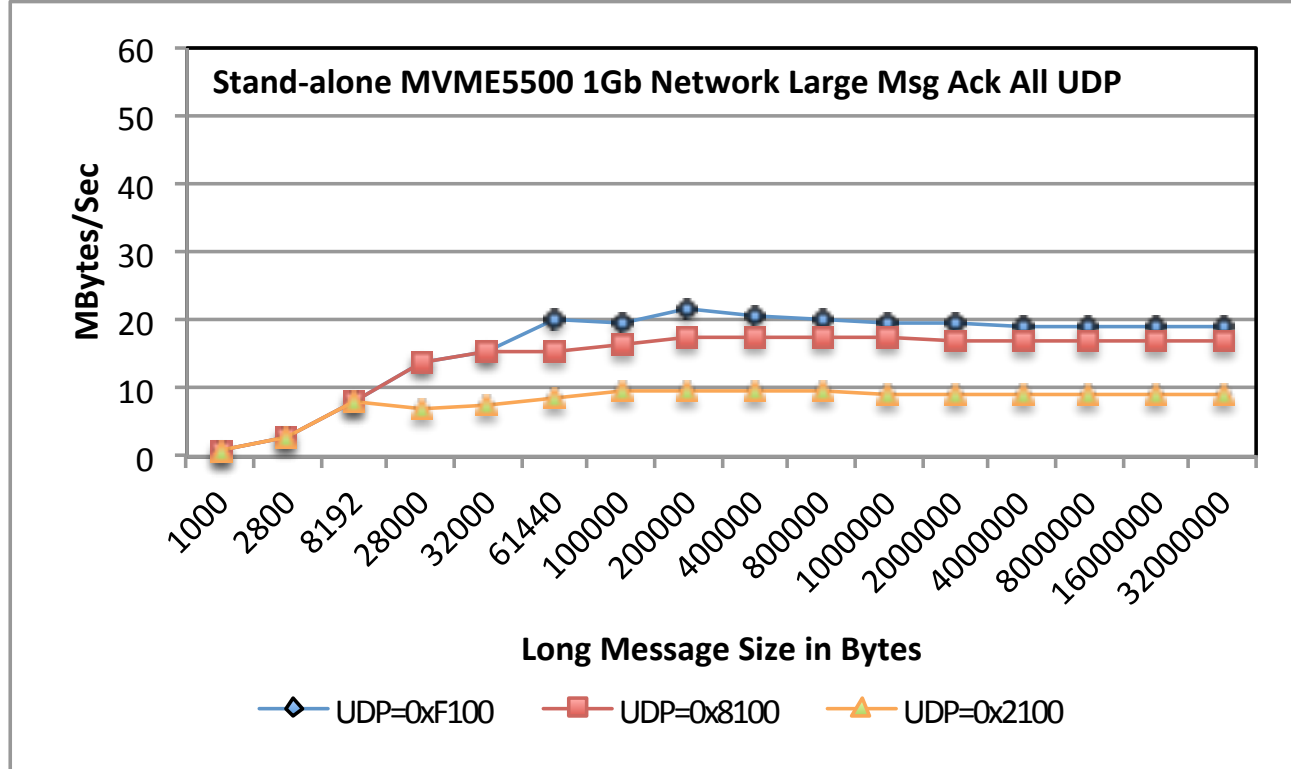
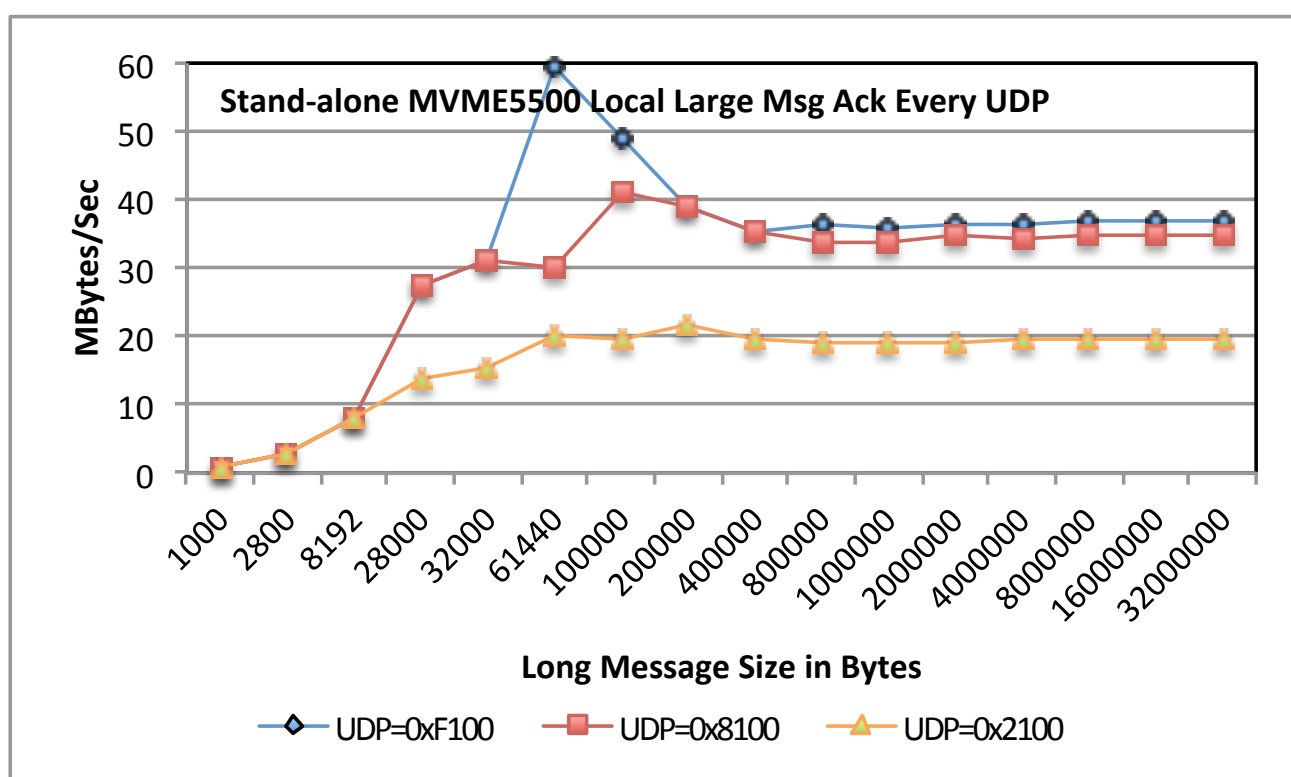
# A Protocol for Streaming Large Messages with UDP

Charlie Briegel, Rich Neswold, Mike Sliczniak  
 FNAL<sup>†</sup>, Batavia, IL 60510, U.S.A.

## Stand-alone Implementation

## Protocol

## ACNet Implementation

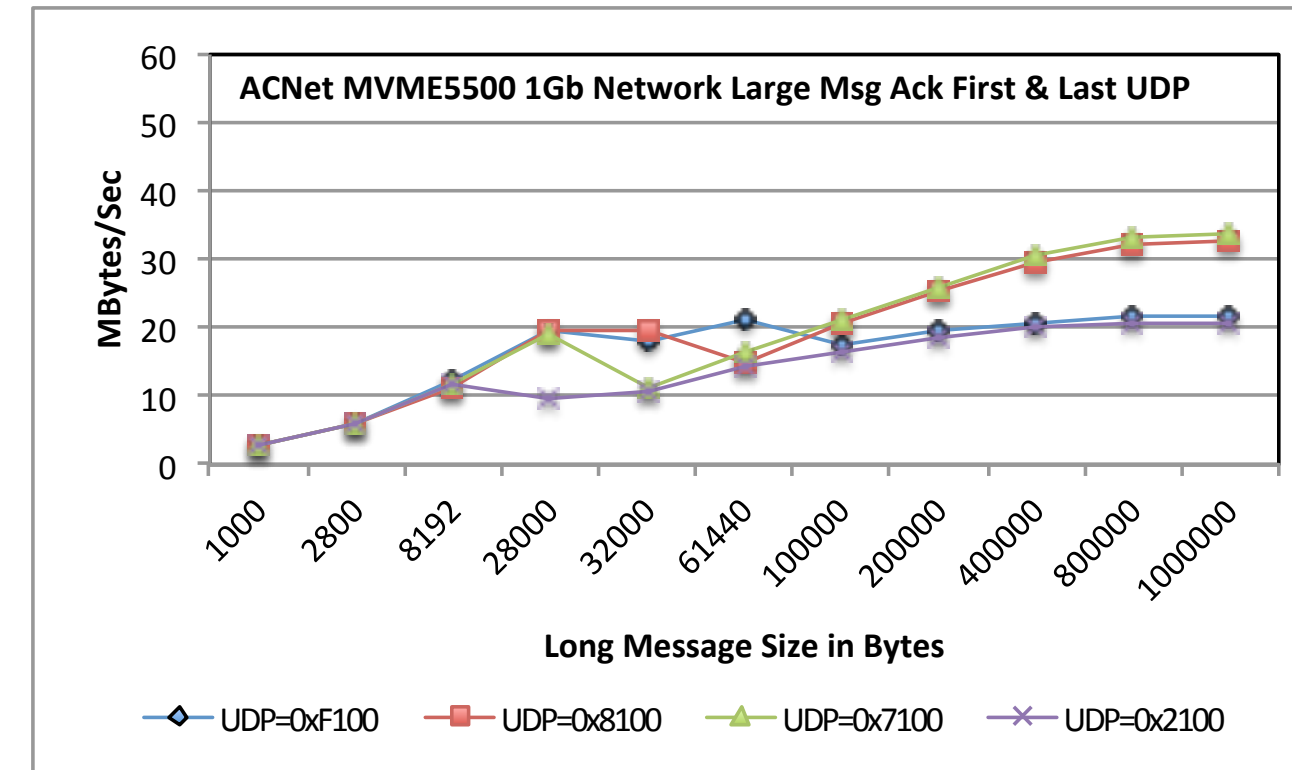
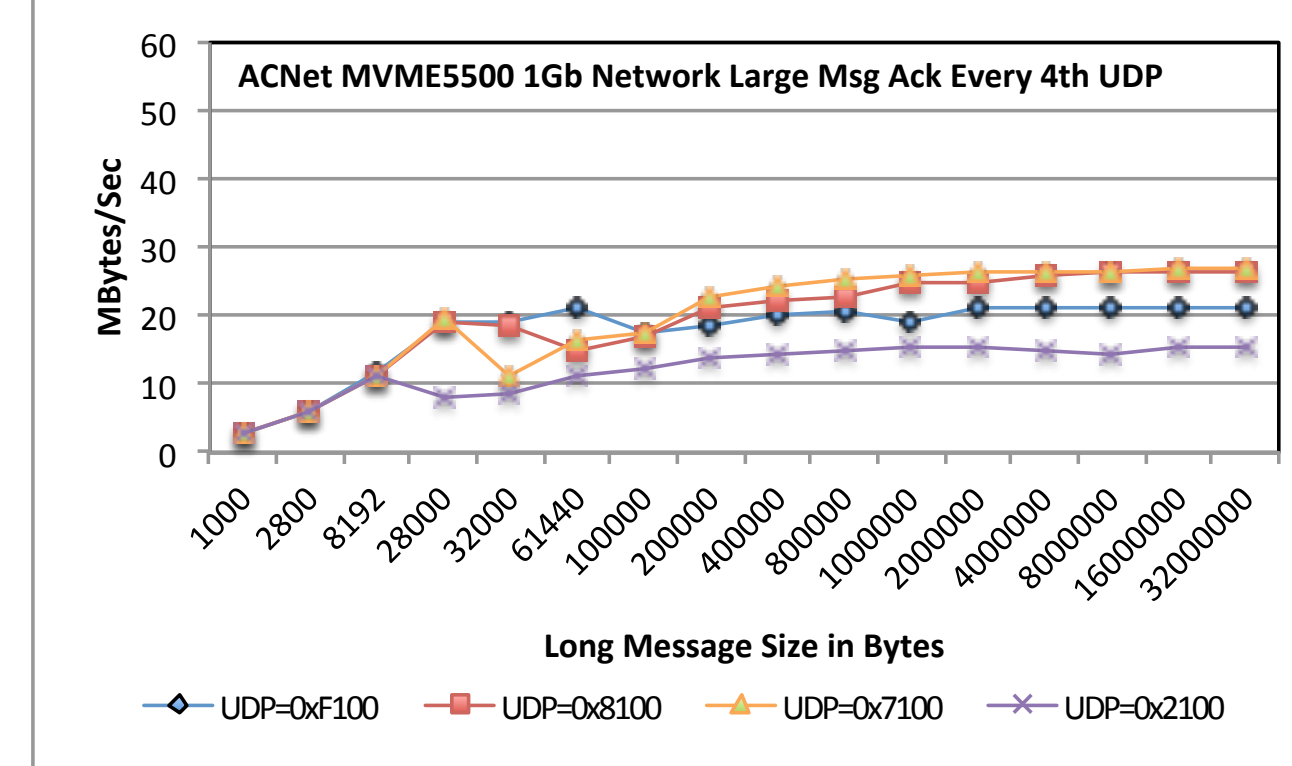
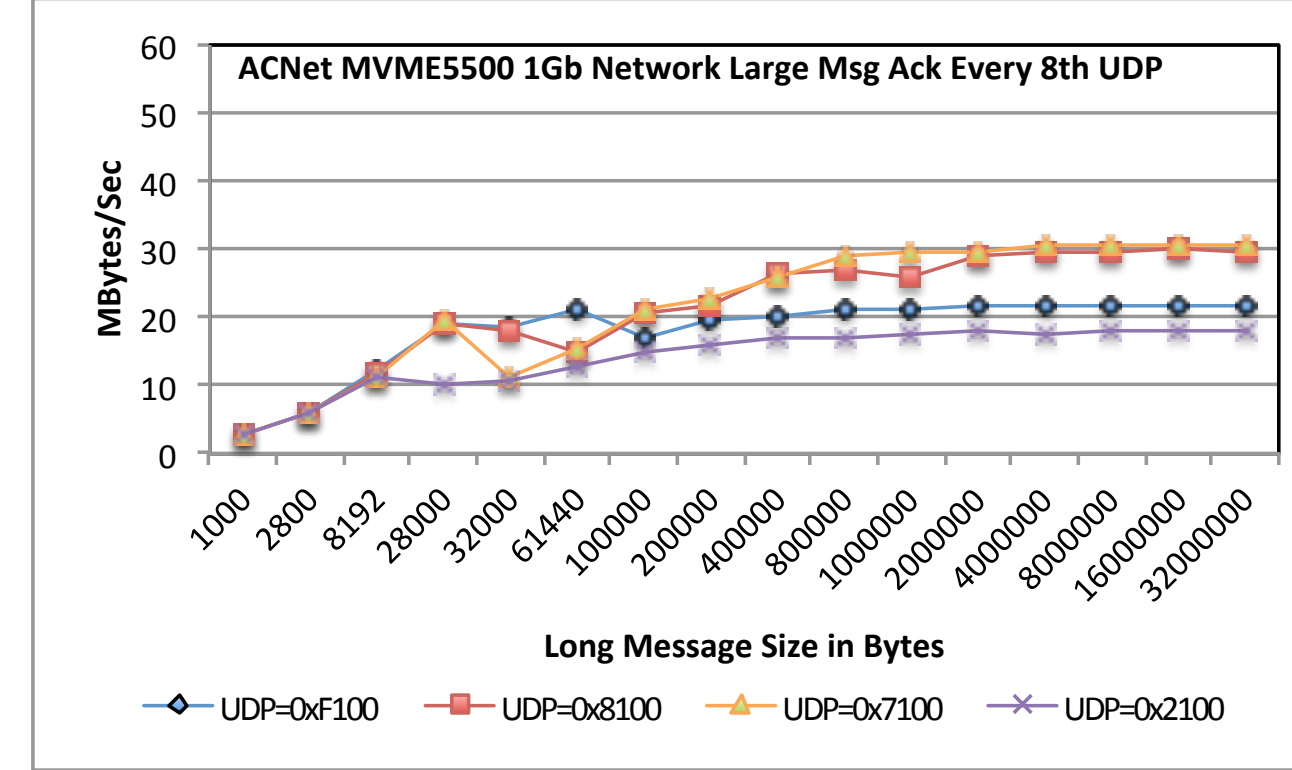
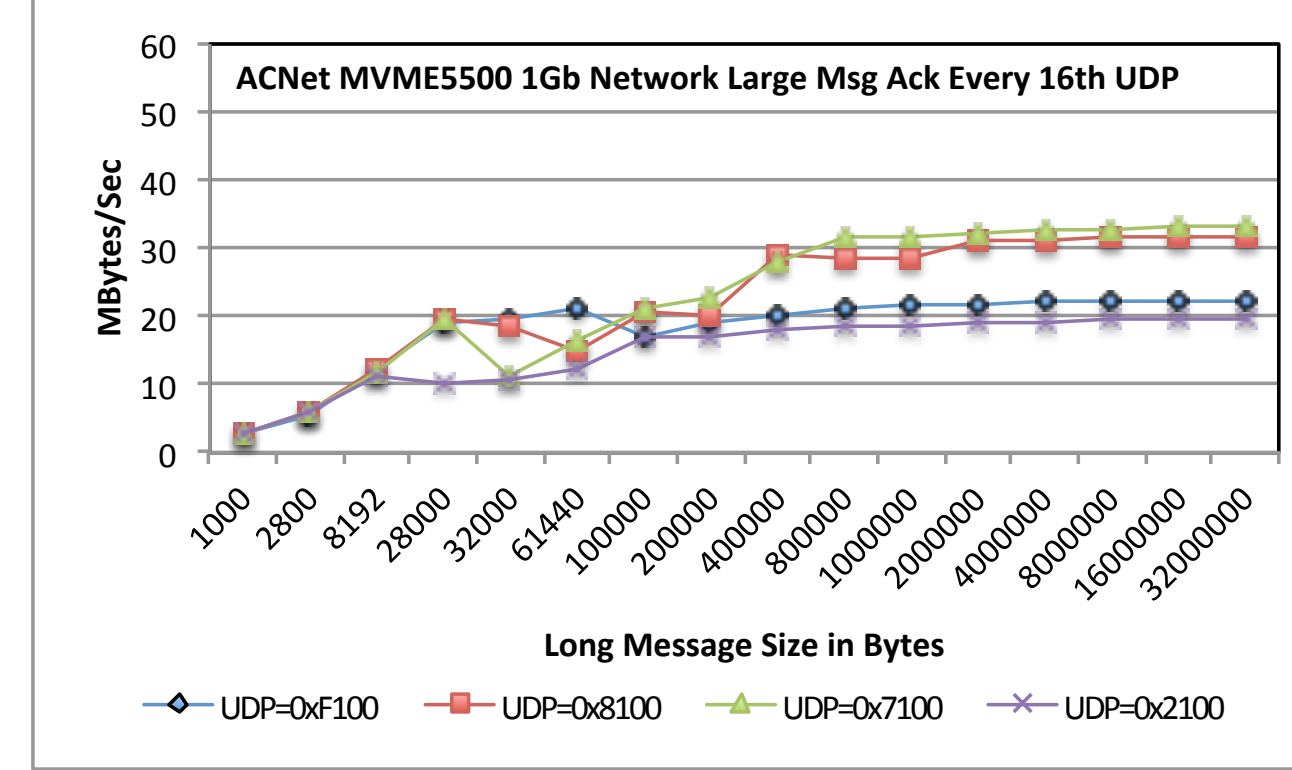
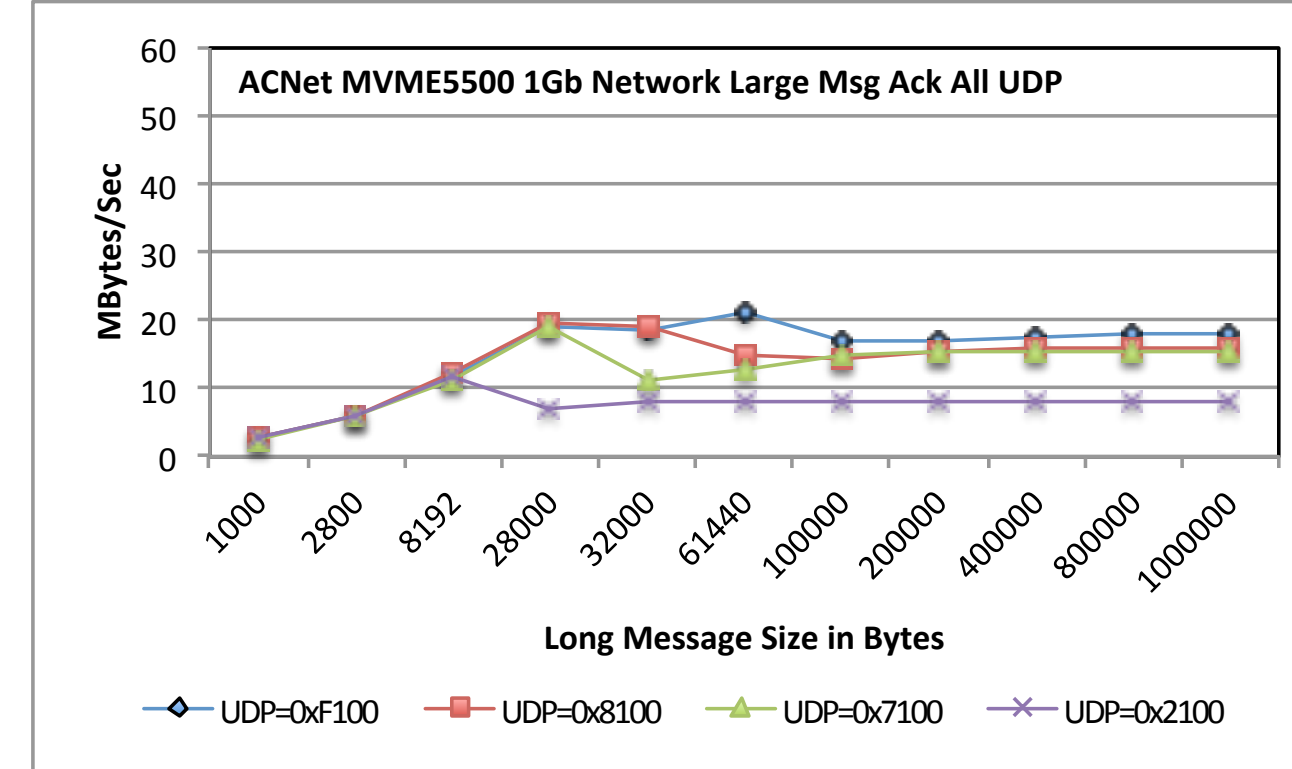
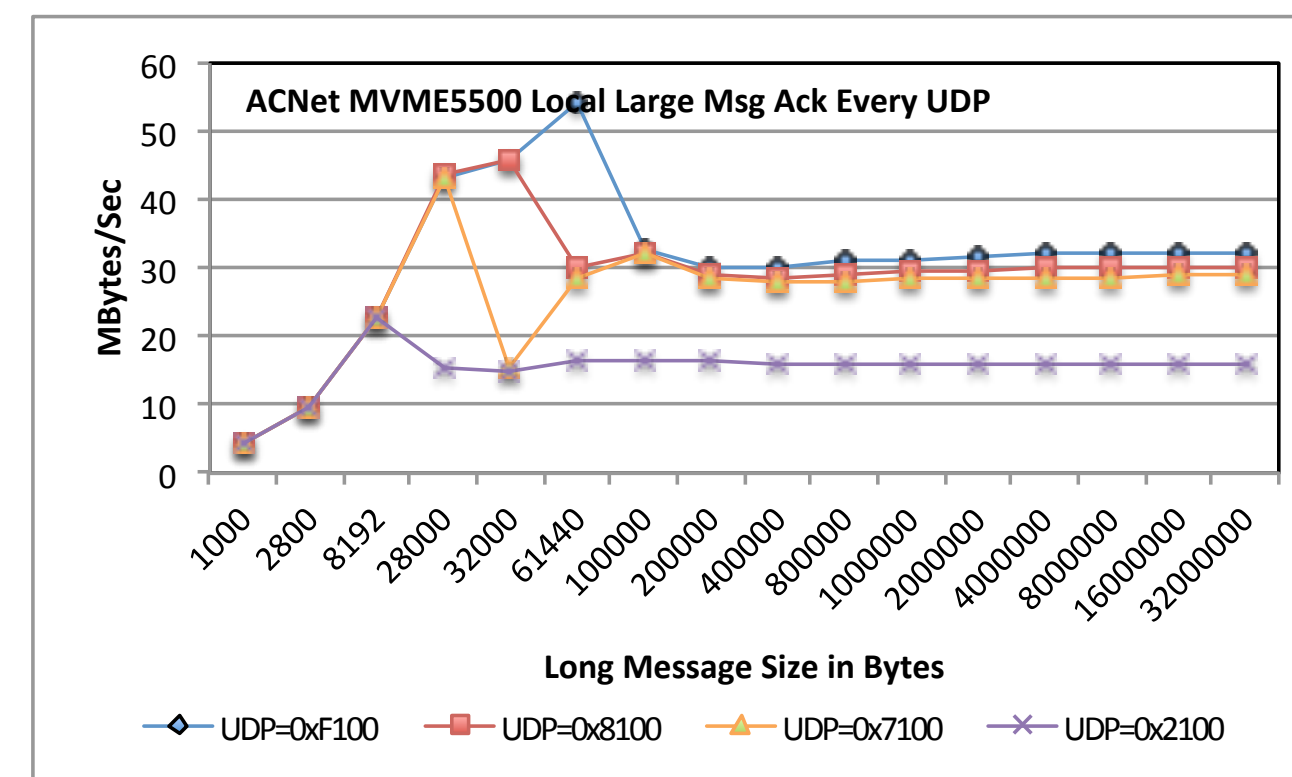


### Rules

- If the offset is zero, then a new reply is arriving. The receiver can use the size field to pre-allocate a buffer to hold the rest of the incoming data. After saving the data in the buffer, it sets the next expected offset to be equal to the size of data that was just received.
- If the offset is non-zero, it checks to see if the offset and transfer ID matches a reply that is in progress. If a match is found, the data is appended to the buffer and the next expected offset is updated.
- After appending the data, if the packet also asked for a response (typecode 1 in the long message header), the task will send a resume message (Figure 2) with the current expected offset.
- If the offset is non-zero and a reply to a transfer ID is in progress but the offset is too high (a packet was dropped), the task waits for a packet that also wants a reply. When it arrives, a resume message is sent to the sender with the offset of the missing data.
- When the transfer is complete, the last packet will also require a response. The receiver returns the expected offset (which at this point will be the size of the data) or a previous offset, if a packet was dropped.

### Recommendations

- The first segment **should** use typecode 1, asking the receiver for a resume message. By doing this, part of the payload gets sent in addition to checking whether the receiver supports large messages (a timeout indicates no support.)
- The last packet of the message **should** use typecode 1 to make sure the entire message was received.
- The sender **may** vary the interval between ACK requests to adapt to network conditions. For instance, the sender might begin the transfer with an interval of 4 packets before asking for an ACK. If there isn't an error, then 8 packets can be sent before the next ACK. If an error occurred, the sender reduces



## ACNet Results

