

**minavgmax:** The function *minavgmax* calculates and updates the minimum, average, and maximum round-trip times (RTTs) for packets.

The function takes a single argument, *ms\_delay*, which represents the round-trip time for a single packet in milliseconds.

Inside the function, a static integer *avg\_counter* is used to keep track of the number of times the function has been called. This is important for calculating the average RTT.

The function first checks if *rtt\_min* is zero or if *ms\_delay* is less than *rtt\_min*. If either condition is true, *rtt\_min* is updated to the value of *ms\_delay*. This ensures that *rtt\_min* always holds the smallest observed RTT. Next, the function checks if *rtt\_max* is zero or if *ms\_delay* is greater than *rtt\_max*. If either condition is true, *rtt\_max* is updated to the value of *ms\_delay*. This ensures that *rtt\_max* always holds the largest observed RTT.

The *avg\_counter* is then incremented by one, and the average RTT *rtt\_avg* is updated. The new average is calculated as a weighted average of the old average and the new RTT. The old average is weighted by  $(\text{avg\_counter} - 1) / \text{avg\_counter}$ , and the new RTT is weighted by  $1 / \text{avg\_counter}$ . This ensures that *rtt\_avg* always holds the average of all observed RTTs.

**Rtt function:** The function *rtt* calculates the round-trip time (RTT) for a packet. It takes three arguments: a pointer to an integer *seqp* [denoting the pointer to the sequence no. value], an integer *recvport*, and a pointer to a float *ms\_delay*.

The function starts by initializing two long integers, *sec\_delay* [total delay in secs] and *usec\_delay* [total delay in micro secs], to zero and an integer *tablepos* to -1. These variables will be used to calculate the delay and to keep track of the position in the delay table.

The function then checks if value pointed by *seqp* is not zero. If it's not, it searches the delay table for an entry with a sequence number matching value at *seqp* and sets *tablepos* to the index of that entry. If value at *seqp* is zero, it searches the delay table for an entry with a source port equal to *recvport* and sets *tablepos* to the index of that entry. If such an entry is found, *seqp* is updated with the sequence number of that entry.

If an entry in the delay table was found (*tablepos* is not -1), the function updates the status of that entry to *S\_RECV[1]*. It then calculates the delay in seconds and microseconds by subtracting the time the packet was sent from the current time. If the delay in microseconds is negative, it adds 1000000 to correct for the wraparound. The delay in milliseconds is then calculated and passed to the *minavgmax* function to update the minimum, average, and maximum RTT.

If no entry in the delay table was found, *ms\_delay* is set to zero and status is set to zero, indicating that the packet is not in the table and its *status* is unknown[could be possible in the case of a duplicate].

The function then performs a sanity check to ensure that *ms\_delay* is not negative. If it is, it prints a detailed report including the sequence number, status, current time, time the packet was sent, and calculated delays. This is for the debug output. Finally, the function returns the status of the packet.

**Delaytable\_add:**

Updates the *delaytable* with the given information.