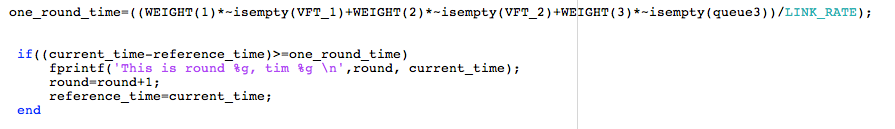
In round robin service, we do not necessarily obtain a fair allocation of transmission bandwidth. For example, if the packets of one flow are twice the size of packets in another flow, then in the long run the first flow will obtain twice the bandwidth of the second flow. A better approach is to transmit packets from the user buffers so that the packet completion times approximate those of a fluid-flow fair-queueing system. Each time a packet arrives at a user buffer, the completion time of the packet is derived from a fluid-flow fair-queueing system.

Part(D)

WFQ server sends out packets in the same order as the perfectly-fair fluid-flow GPS system. In particular, each packet is assigned a Virtual-Finishing-Time (VFT) when it arrives, based on the current round and the state of its queue. In order to assign each packet a VFT, we need to main two clocks, namely "round" and "real time." At the same time we increment the real time, we also update the virtual time according to the state of the queues and their weights. For example, suppose we have queue 1 active for 0 to 2 and queue 2 active from 2 to 5 ms queue3 active from 3 to 9 ms. Each flows has a weight 1 and the capacity is 1kbs.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Real time/ms | (0,1) | (1,2) | (2,3) | (3,4) | (4,5) | (5,6) | (6,7) | (7,8) | (8,9) |
| Duration of one round/ms | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 1 | 1 |
| Number of active queues. | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 1 | 1 |
| Virtual time/round | 0 | 1 | 2 | 3 | 3 | 4 | 4 | 5 | 6 |

Each time I update the round, I set a reference time at that moment so that I would be able to know when I should update the next round based on time difference of current time and previous reference time.



In the matlab code, the GPS/WFQ methodology is realized as follow:

1. Increment the real time
2. Compute round for every real time value
3. Calculate the VFT for each packet that have arrived using equation 
4. Pick up the packet with smallest VFT in the storage and then transmit to the buffer.

The detailed matlab code implementation is shown next part.

Part(E)

For the purpose of clarity and convenience of analysis, I just choose 40 rounds for Round\_Robin Service and 2 seconds for Weighted Fair Queue service.

From the first graph, we can clearly see that each flow feed the transmission buffer one packet a time based on the arrival time no matter what size it is. This might cause unfairness as demonstrated before. Therefore, when we go to second graph, this situation is improved. Each flow has a relative same chances to show up and that means packets that arrived earlier but with a large size would be delayed whereaus small packets with relative late arrival time would be transmitted ahead. By doing this, each flow will gain a more fair system to transmit their packets.

