The GPS Simulator Pseudocode

March 25, 2017

1 GPS Simulator

Algorithm 1 Handling the Interrupt Triggered by a Packet Arrival

1: procedure GPSSim::HandleNewPacketArrival(PACKET *pPKT)
2:   double nowRTime, nowVTime;
3:   FLOW *pFlow;
4:   BOOL b;
5:   PACKET *pCurPacket;
6:   nowRTime = current real time;
7:   if (mIdling) then
8:     nowVTime = 0;
9:     mIdling = FALSE;
10: else
11:     nowVTime = mThenVTime + (nowRTime - mThenRTime) / mSumWeight;
12:     pFlow = pPKT→mpFlow;
13:     b = flow→IsBackloggedUnderGPS();
14:     if (!b) then
15:       mSumWeight = mSumWeight + pFlow→mWeight;
16:       pPKT→GPS_VFTime = nowVTime + pPKT→mLength/pFlow→mWeight;
17:       pFlow→AppendPacket(pPKT);
18:     if (!b) then
19:       mpPQ_HOL→Enqueue(pPKT);
20:       pCurPacket = mpPQ_HOL→Peek_Min();
21:       if (pCurPacket != mpCurPacket) then
22:         mpCurPacket = pCurPacket;
23:         ResetTimer(nowVTime, mpCurPacket→mGPS_VFTime);
24:       mThenVTime = nowVTime;
25:       mThenRTime = nowRTime;
26:     end procedure
Algorithm 2 Reset the Timer
1: procedure GPSSim::ResetTimer(double nowVTime, double newWakeupVTime)
2:  double newInterval;
3:  newInterval = (newWakeupVTime - nowVTime) * mSumWeight;
4:  StopTimer();
5:  StartTimer(newInterval, WakeupProcessing());
6: end procedure

Algorithm 3 When a Packet Finishes Service Under GPS
1: procedure GPS::WakeupProcessing()
2:  double nowVTime;
3:  FLOW *pFlow;
4:  PACKET *pPKT;
5:  nowVTime = mpCurPacket→mGPS_VFTime;
6:  mpPQ_HOL→PopMin();
7:  pFlow = mpCurPacket→mpFlow;
8:  pFlow→PopHOL();
9:  if (pFlow→IsBackloggedUnderGPS()) then
10:     pPKT = pFlow→PeekHOL();
11:     mpPQ_HOL.Enqueue(pPKT);
12:  else
13:     mSumWeight = mSumWeight - pFlow→mWeight;
14:     mpCurPacket = mpPQ_HOL→PeekMin();
15:  if (mpCurPacket == NULL) then
16:     CleanUpAfterBusyPeriod();
17:  else
18:     ResetTimer(nowVTime, mpCurPacket→GPS_VFTime);
19: end procedure