

CSE322 COMPUTER NETWORKS SESSIONAL

Asif Ajrof 1705092

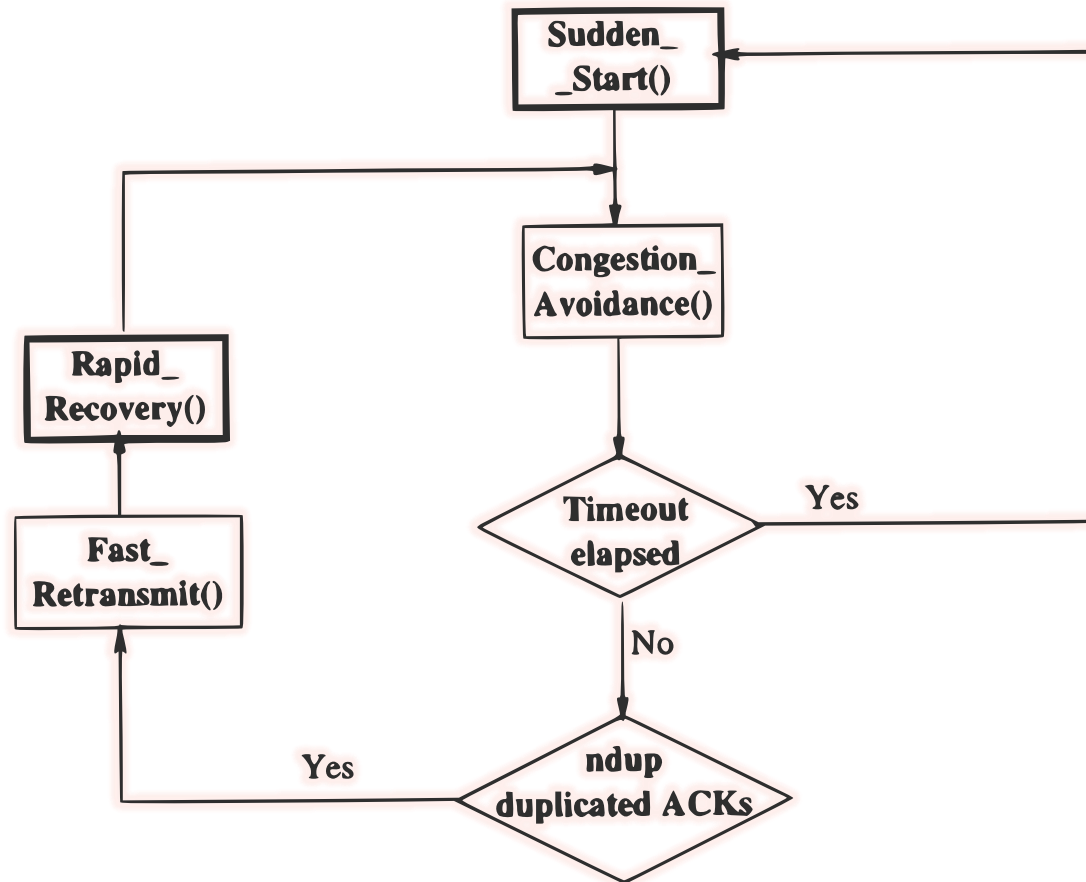
TCP-Peach

A New Congestion Control Scheme for Satellite IP Networks

Update 1

IDEA Overview

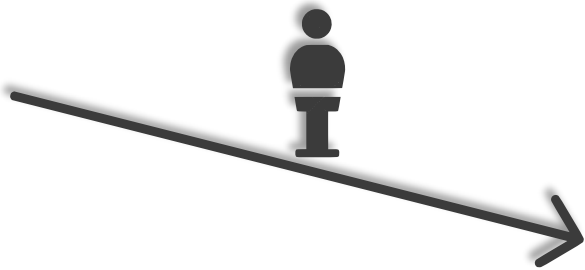
TCP-Peach Scheme



IDEA Overview

Modifications

- Dummy Segment



IDEA Overview

Modifications

- Dummy Segment
- Sudden_Start()

```
Sudden_Start( )  
  cwnd=1;  
   $\tau = RTT/rwnd$ ;  
  send(Data_Segment);  
  for (i=1 to  $rwnd - 1$ ),  
    wait( $\tau$ );  
    send(Dummy_Segment);  
  end;  
end.
```

IDEA Overview

Modifications

■ Dummy Segment

■ Sudden_Start()

■ Rapid_Recovery()

```
Rapid_Recovery()  
  cwnd=cwnd/2;  
  adsn=2*cwnd;  
  wdsn=cwnd;  
  infl_seg=0;  
  tRetr=t;  
  END=0;  
  while (END=0)  
    if (ACK_ARRIVAL)  
      if (DATA_ACK_ARRIVAL)  
        cwnd=cwnd+1;  
        infl_seg=infl_seg+1;  
      else if (DUMMY_ACK_ARRIVAL)  
        if (wdsn=0)  
          cwnd=cwnd+1;  
          infl_seg=infl_seg+1;  
        else  
          wdsn=wdsn-1;  
        end;  
      end;  
    end;  
  end;
```

```
  if (cwnd>nackseg)  
    while (cwnd>nackseg)  
      send(Data_Segment);  
      nackseg=nackseg+1;  
    end;  
  else if (adsn>0)  
    send(Dummy_Segment);  
    send(Dummy_Segment);  
    adsn=adsn-2;  
  end;  
  if (LOST_SEGMENT_ACKED)  
    END=1;  
    cwnd=cwnd-infl_seg;  
  end;  
end;  
if (t>tRetr+RTO)  
  Slow_Start();  
end;  
end;  
end.
```

Dummy Segment

Modification in packet

```
255 void
256 Packet::AddHeader (const Header &header)
257 {
258     uint32_t size = header.GetSerializedSize ();
259     NS_LOG_FUNCTION (this << header.GetInstanceTypeId ().GetName () << size);
260     m_buffer.AddAtStart (size);
261     m_byteTagList.Adjust (size);
262     m_byteTagList.AddAtStart (size);
263     header.Serialize (m_buffer.Begin ());
264     m_metadata.AddHeader (header, size);
265 }
```

```
296 uint32_t
297 Packet::PeekHeader (Header &header, uint32_t size) const
298 {
299     Buffer::Iterator end;
300     end = m_buffer.Begin ();
301     end.Next (size);
302     uint32_t deserialized = header.Deserialize (m_buffer.Begin (), end);
303     NS_LOG_FUNCTION (this << header.GetInstanceTypeId ().GetName () << deserialized);
304     return deserialized;
305 }
```

Files

packet.h
packet.cc

Sudden_Start()

Modification in Slow_Start()

```
165  uint32_t
166  TcpNewReno::SlowStart (Ptr<TcpSocketState> tcb, uint32_t segmentsAacked)
167  {
168      NS_LOG_FUNCTION (this << tcb << segmentsAacked);
169
170      if (segmentsAacked >= 1)
171      {
172          tcb->m_cWnd += tcb->m_segmentSize;
173          NS_LOG_INFO ("In SlowStart, updated to cwnd " << tcb->m_cWnd << " ssthresh " << tcb->m_ssThre
174          return segmentsAacked - 1;
175      }
176
177      return 0;
178 }
```

Files

tcp-congestion-ops.h
tcp-congestion-ops.cc

Rapid_Recover()

Modification in EnterRecovery()

```
1600 void
1601 TcpSocketBase::EnterRecovery (uint32_t currentDelivered)
1602 {
1603     NS_LOG_FUNCTION (this);
1604     NS_ASSERT (m_tcb->m_congState != TcpSocketState::CA_RECOVERY);
1605
1606     NS_LOG_DEBUG (TcpSocketState::TcpCongStateName[m_tcb->m_congState] <<
1607                  " -> CA_RECOVERY");
1608
1609     if (!m_sackEnabled)
1610     {
1611         // One segment has left the network, PLUS the head is lost
1612         m_txBuffer->AddRenoSack ();
1613         m_txBuffer->MarkHeadAsLost ();
1614     }
1615     else
1616     {
1617         if (!m_txBuffer->IsLost (m_txBuffer->HeadSequence ()))
1618         {
```

src/internet/model/

Files

tcp-socket-base.cc.h
tcp-socket-base.cc

