CSE 322: Computer Network Sessional (NS3)

CUBIC-FIT: A High Performance Congestion Control Algorithm

Nahian Salsabil
1705091

Main Objective

- Simulates multiple (N) cubic flows in a single TCP connection

- Adjust the simulated CUBIC flow number

Modification

CUBIC

$$T_{cubic} = \frac{1}{RTT}\sqrt[4]{\frac{C(4-b)}{4b}\left(\frac{RTT}{PLR}\right)^3}.$$

$$T_{cubic}^{plain} = \frac{1}{RTT}\sqrt[4]{1.9\left(\frac{RTT}{PLR}\right)^3}.$$

CUBIC FIT

$$T_{cubic}^{fit} = N \cdot T_{cubic}^{plain}$$

$$T_{cubic}^{fit} = \frac{N}{RTT}\sqrt[4]{1.9\left(\frac{RTT}{PLR}\right)^3}.$$

Parameters

- **C, b** - tuned parameter (for CUBIC, C = 0.4)
- RTT - Round Trip Time
- PLR - Packet Loss Ratio

$$T_{cubic} = \frac{1}{RTT}\sqrt[4]{\frac{C(4-b)}{4b}\left(\frac{RTT}{PLR}\right)^3}.$$

$$T_{cubic}^{fit} = \frac{N}{RTT}\sqrt[4]{1.9\left(\frac{RTT}{PLR}\right)^3}.$$

Modification

CUBIC

$$w_{cubic} := C(t - I)^3 + w_{max}$$

$$I = \sqrt[3]{(w_{max}b)/C}.$$

CUBIC FIT

$$w_{cubic}^{fit} = 0.4(Nt - I)^3 + w_{max}$$

$$I = \sqrt[3]{\frac{10w_{max}}{19N + 1}}$$

Codebase Modification

New files:

- *src/internet/models/tcp-cubic-fit.cc*
- *src/internet/models/tcp-cubic-fit.h*

Add:

- *Tcp-cubic-fit.cc*
- *tcp-cubic-fit.h in*

In **src/internet/wscript**

To change the default congestion control algorithm(CUBIC), add this line at the beginning of the main function of **seventh.cc**

Config::SetDefault ("ns3::TcpL4Protocol::SocketType", StringValue ("ns3::TcpCubicFit"));

Codebase Modification

```
uint32_t
TcpCubic::Update (Ptr<TcpSocketState> tcb)
```

```
else
  {
    m_bicK = std::pow ((m_lastMaxCwnd - segCwnd) / m_c, 1 / 3.);
    m_bicOriginPoint = m_lastMaxCwnd;
    NS_LOG_DEBUG ("lastMaxCwnd > m_cWnd. K=" << m_bicK <<
                  " and origin=" << m_lastMaxCwnd);
  }
```

Codebase Modification

```
uint32_t
TcpCubic::Update (Ptr<TcpSocketState> tcb)
```

```
if (t.GetSeconds () < m_bicK)       /* t - K */
  {
    offs = m_bicK - t.GetSeconds ();
    NS_LOG_DEBUG ("t=" << t.GetSeconds () << " <k: offs=" << offs);
  }
else
  {
    offs = t.GetSeconds () - m_bicK;
    NS_LOG_DEBUG ("t=" << t.GetSeconds () << " >= k: offs=" << offs);
  }
```

Problem Faced

- TCP friendliness not found