EEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HEW SG Simulation Scenarios | | | | |
| Date: November 4, 2013 | | | | |
| Authors and Contributors | | | | |
| Name | Company | Address | Phone | Email |
| Guoqing Li | Intel | 2111 NE 25th Ave, Hillsboro, OR 97124 | 503/712-2089 | Guoqing.c.li@intel.com |
| Yiting Liao | Intel |  |  |  |
| Dmitry Akhmetov | Intel |  |  |  |
| Robert Stacey | Intel |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Abstract

This document describes the video traffic model for wireless display, video streaming and video conferencing for HEW SG

# Revisions

|  |  |  |
| --- | --- | --- |
| **Revision** | **Comments** | **Date** |
| *R0* | Initial draft template | Oct 30th |
| *R1* |  |  |
| *R2* |  |  |
| *R3* |  |  |
|  |  |  |

# Introduction

We introduce the traffic models for the following three video applications based on information described in doc#xx

* Wireless display
* Video steaming
* Video conferencing

# Video Traffic Model

1. **Wireless Display (lightly compressed video) Traffic Model**

Wireless display is a single-hop unidirectional (e.g., laptop to monitor) video application. The video slices (assuming a slice is a row of macroblocks) is generated at fixed slice interval. For example, for 1080p, the slice interval is 1/4080 seconds.

The video slices are typically packizeded into MPEG-TS packets in wireless display application. But for HEW simulation, we will ignore the MPEG-TS packetization process and assume video slices are delivered to MAC layer for transmission directly.

The traffic model for wireless display is modified from [TGad] with modifications below due to the fact that some parameters have dependency on video formats.

1. Parameters
   1. Set **IAT**, **MaxSliceSize** according to video format as Table xx.
   2. Normal distribution parameters
      1. µ = 15.798 Kbytes
      2. σ = 1.350 Kbytes
      3. b = 300 Mbps
2. Algorithm for generating each video slice/packet

* Input: target bit rate in Mbps (**p**)
* Output: slice size in Kbytes (L): At each IAT, generate a slice size L with the following distribution: Normal(µ\*(p/b), σ\*(p/b))
  + - If L > MaxSliceSize, set L= MaxSliceSize

|  |  |  |  |
| --- | --- | --- | --- |
| **Video format** | **Inter-arrival time (IAT)** | **MaxSliceSize** | **p** |
| 1080p60 | 1/4080 seconds | 92.160 Kbytes | 300 |
| 4K UHD (3840x2160) 60fps | 1/8100 seconds | 184.320 Kbytes | 600 |
| 8K UHD (7680x4320) 60fps | 1/16200 seconds | 368.640 Kbytes | 1200 |
| 1080p60 3D | 1/4080 seconds | 92.160 Kbytes | 450 |

# Note: the data rate increase from 1080p to higher resolution is not linearly scaling as the uncompressed data rate due to higher redundancy in the images at higher resolution. Similar argument applies to 3D video. A 100% incease is assumed for 4K video as compared to 1080p, and 50% bit rate increase for 3D from 2D video.

**Evaluation metric**

* MAC throughput, latency

1. **Buffered Video Steaming (e.g., Youtube, netflix) Traffic Model**

Unlike wireless display, video streaming is generated from a video server, and tranverses multiple hops in the internet before arriving at AP for transmission to STA. It is a unidirectional traffic from the video server to the station.

Typically, Video streaming application runs over TCP/IP protocol, and video frames will be fragmented at TCP layer before leaving the video server. Since these TCP/IP packets experiences different processing and queueing delay at routers, the inter-arrival time between these TCP/IP packets are not a constant despite the fact that video frames are generated at constant interval at the video application layer.

**STA Layering Model**

STA layering model is shown in Figure xx. Both AP and STA generate video frames at application layer. The video traffic goes through TCP/IP layer and then to MAC layer. The TCP protocol used for video streaming simulation is the same as other traffic model described in section x.x. of this document.



Figure xx Traffic layering model

**Video traffic generation**

The video traffic from AP to STA is generated as follows.

**Step 1**: At application layer, generate video frame size (bytes) according to Weibull distribution with the following PDF.


f(x;\lambda,k) =
\begin{cases}
\frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1}e^{-(x/\lambda)^{k}} & x\geq0 ,\\
0 & x<0,
\end{cases}

Depending on the video bit rate, the parameters to use are specified in Table x.

|  |  |  |
| --- | --- | --- |
| **Video bit rate** | **lamda** | **k** |
| 10Mbps | 34750 | 0.8099 |
| 8Mbps | 27800 | 0.8099 |
| 6Mbps | 20850 | 0.8099 |
| 4Mbps | 13900 | 0.8099 |
| 2Mbps | 695 | 368.640 Kbytes |

**Step 2**: AT TCP layer, set TCP segment as 1500 bytes and fragment video pacekt into TCP segments.

**Step 3**: Add network latency to TCP/IP packets when thse segments arrive at AP for transmission. The network latency is generated according to Gamma distribution whose PDF is shown below

f(x;k,\theta) =  \frac{x^{k-1}e^{-\frac{x}{\theta}}}{\theta^k\Gamma(k)} \quad \text{ for } x > 0 \text{ and } k, \theta > 0.

Where

* + k=0.2463
  + theta=55.928

The mean of the latency with the above parameters is 14.834ms. To simulate longer or shorter network latency, scale theta linearly since mean of Gamma distribution is K\*theta

If network latency value is such that the packet arrives at MAC layer after the end of the simulation time, then re-generate another network latency value until the packet arrives at MAC within the simulation window.

**Evaluation metrics**

* MAC throughput, latency
* TCP throughput, latency

1. **Video Conferencing (e.g., Lync) Traffic Model**

Unlike video conferencing where video traffic is unidirectional, video conferencing is two-way video traffic. The video traffic is generated at each station, send to AP, tranverse the internet and reach another AP and then send to the destination.

**Station layer model**



Because the traffic from AP to station has experienced network jiter, it can be modelled the same way as the traffic model of video streaming.

For the traffic sent from Sation to AP, since the traffic has not experienced network jitter, it is a periodic traffic generation as the first two steps described in video streaming.

**Video traffic generation**

Traffic model from AP to station: use the same model as video streaming.

Traffic model from station to AP: use the first two steps in video streaming traffic model

**Evaluation metrics**

* MAC throughput, latency

# References

1. **11-13/486, “HEW video traffic modeling” Guoqing Li et al, (Intel) [1] 11-13-1162-01-hew-vide-categories-and-characteristics**
2. **[2] 11-13-1059-01-hew-video-performance-requirements-and-simulation-parameters**
3. **[3]11-09-0296-16-00ad-evaluation-methodology.doc**
4. **[4] Rongduo Liu et al., “An Emperical Traffic Model of M2M Mobile Streaming Services ”, International conference C on Multimedia information networking and security, 2012**
5. **[5] JO. Rose, “ Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems ”, Tech report, Institute of CS in University of Wurzburg**
6. **[6] Savery Tanwir., “A survey of VBR traffic models”, IEEE communication surveys and tutorials, Jan 2013**
7. **[7] Aggelos Lazaris et al., “A new model for video traffic originating from multiplexed MPEG-4 videoconferencing streams”, International journal on performance evaluation, 2007**
8. **[8] A. Golaup et al., “Modeling of MPEG4 traffic at GOP level using autoregressive process”, IEEE VTC, 2002**
9. **[9] K. Park et al., “Self-Similar network traffic and performance evaluation”, John Wiley&Son, 2000**
10. **[10] M Dai et al., “A unified traffic model for MPEG-4 and H.264 video traces”, IEEE Trans. on multimedia, issue 5 2009.**
11. **[11] L Rezo-Domninggues et al., “Jitter in IP network: A cauchy approach”, IEEE Comm. Letter, Feb 2010**
12. **[12] Hongli Zhang et al., “Modeling Internet link delay based on measurement”, International conference on electronic computer technology, 2009.**