# Simulation of Video Transmission over Wireless IP Network in Fedora Environment

Lal Chand Bishnoi
Uttarakhand Technical University, Dehradun
India

Dharm Singh
College of Technology and Engineering, MPUAT
Udaipur, India

Shailendra Mishra
Kumaon Engineering College, Dwarahat
India

## ABSTRACT
Due to the very large and quick growth of the wireless network and increasing demand for multimedia information on the web, video transmission over the wireless network has received tremendous attention from academe and industry. Transmission of real-time video typically has bandwidth, delay, and loss requirements. Video transmission over wireless network poses many challenges. To address these challenges, extensive research has been conducted in the various areas of wireless video application. This paper is aimed at dissemination of the contributions a simulation environment for video transmission over the wireless network in Fedora environment because the Cygwin in window-xp environment has lots of limitation.

## General Terms
Multimedia Communication, Video over IP Network, Network Simulator, Wireless Network communication et. al.

## Keywords
NS2, VoD. VoIP, Video Transmission, Wireless Network et. al.

## 1. INTRODUCTION
The video transmission over wireless network is commonly today's requirement of each laptop, palmtop, mobile users. Without compression it is very difficult to transmit video over wired or wireless network because video content requires very large network bandwidth. For instance, 720p video at 60 frames/s using 10 b/colour requires about 1.4 Gb/s. To transmit the content over bandwidth- limited media like wireless IEEE 802.11, the content (even real-time content) needs to be compressed. The overview of video transmission over wireless network shown in figure 1.

In the research area of wireless communication and networking for transmitting video a simulation environment is required [1][2]. The Network Simulator 2 (NS2) is freeware simulation tool used for research. There are two commonly operating system to create simulation environments are: Cygwin in Microsoft Window–xp and other is Fedora.

The Cygwin has various limitations. It better work only in Microsoft Window-xp. The newly download Cygwin software [3] is not used for Network Simulation. Only Cygwin version 2.4 or 2.5 is used not higher version. It would not contain all facility which is provided by Linux. Even though this operating system itself also freeware.

So it is better to install NS2 for research in Fedora as Fedora support full Linux environment. In this environment normal NS2 research work will run but tool set of video transmission over wireless will create problem while running. They directly would not run. So for that we have requirement special installation process to do research on video transmission on wireless in Fedora Environment.

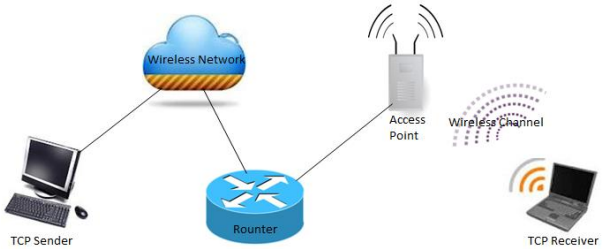


**Fig 1: Video transmission over wireless network**

This paper gives about two NS2 simulation environments for wireless network (WLAN) and multimedia technology, video transmission over wireless network on Cygwin on Microsoft Window-xp and Fedora. This paper explains detail setup steps of Network Simulation Environment on Fedora.

## 2. INSTALLATION AND PREPARING FEDORA FOR SIMULATION
A real world power full freeware operating system can be standalone installed easily on Desktop computer or Laptop or through VMware Virtual Environment. It can download from [4]. We selected Fedora12-i386 for setting multimedia environment for wireless network simulation.

### 2.1 Pre Installation Work for NS2
After installation of Fedora we must prepare it for NS2. For that we require to install some necessary packages those are mandatory to compile and run NS2 programs. All packages are installed in super user mode by yum installer.

1) The following list of packages should be accurate for Fedora 12 release; other releases or other Debian-based systems may slightly vary.
```
#yum install  gcc
#yum install  tcl-devel
#yum install  libX11-devel
#yum install  libXt-devel
#yum install  libXmu-devel
#yum install  gnuplot
```

2) Minimal requirements for Python (release): This is the minimal set of packages needed to work with Python bindings from a released tarball.
   #yum install gcc gcc-c++ python python-devel
3) Running regression tests requires mercurial to fetch the trace repositories. mercurial is also needed in general to work with ns development repositories. You may want to install mercurial from rpmforge repository (instructions here) or EPEL.
   #yum install mercurial
   # yum install gcc gcc-c++ python
4) Running python bindings from the ns development tree (ns-dev) requires bazaar. You may need EPEL repository for this.
   #yum install bzr
5) An optional but recommended package (for improving some wireless model fidelity) is GNU scientific library:
   #yum install gsl gsl-devel
6) A GTK-based configuration system.
   #yum install gtk2 gtk2-devel
7) Debugging packages:
   #yum install gdb valgrind
8) Doxygen and related inline documentation:
   #yum install doxygen graphviz ImageMagick
   #yum install texinfo texinfo-tex
9) The ns manual and tutorial are written in Texinfo (doc/tutorial or doc/manual):
   #yum install texinfo dia texinfo-tex texi2html
10) The Network Simulation Cradle (nsc) requires the flex lexical analyzer and bison parser generator:
    #yum install flex bison
11) To install gcc-3.4 for some Network Simulation Cradle (nsc) stacks:
    #yum install compat-gcc-34
12) To read pcap packet traces.
    #yum install tcpdump
13) Database support for statistics framework.
    #yum install sqlite sqlite-devel
14) Xml-based version of the config store (requires libxml2 >= version 2.7).
    #yum install libxml2 libxml2-devel
15) Binary Utility files for new VMwareTools 8.4.6 Installations.
    #yum install bilutils
16) Following packages are necessary for NS-2 common operation.
    #yum install gcc-c++ compat-gcc-34-c++ \
    automake autoconf libtool libX11-devel \
    libXext-devel libXau-devel libXmu-devel \
    xorg-x11-proto-devel
17) Packages required for X264/AVC Encoder.
    #yum install yasm
18) GLUT package for MP4BOX Toolset.
    #yum –y install freeglut
    #yum –y install freeglut-develd

## 2.2 Network Simulator Installation

Once the required packages installed in Fedora. We can install Network Simulator. You can install any Network Simulator (NS2 or NS3). In this paper we are taking the example of NS2.29.3 installation steps and assuming Fedora home directory is "/home/bishnoi/".

1) Download NS2.29.3 from [5] in home directory.
   $wget http://sourceforge.net/projects/nsnam/files \
   /allinone/ns-allinone-2.29/ \
   ns-allinone-2.29.3.tar.gz/download
2) Extract download file in home directory.
   $tar xzvf ns-allinone-2.29.3.tar.gz
3) Change directory to NS2 home before installation.
   $cd ns-allinone-2.29
4) For NS2.29.3, C++ compiler setting must require. So we use export command.
   $ export CC=gcc34 CXX=g++34
5) Similarly NS2.29.3 also require patch before start installation. So first download path from [6] and then install it as follows:
   $cp /home/bishnoi/tk-8.4-lastevent.patch \
   /home/bishnoi/ns-allinone-2.29/tk-8.4.11
   $cd /home/bishnoi/ns-allinone-2.29/tk-8.4.11
   $patch -p0 < tk-8.4-lastevent.patch
   $cd /home/bishnoi/
6) Before Installation please check the permission of all "configure" files (in all NS2 folders) permission is Executable or not? If not them give executable permission them.
   $chmod –R 775 */
7) Install NS2 in your computer. It will take at least ½ hours. Depends upon your computer speed.
   $cd /home/bishnoi/ns-allinone-2.29/
   $./install
8) After successful install it will display the important notice for setting the environment variables.

## 2.3 Setting NS2 Environment Variables

Once installation over we require to set environment variables such as LD_LIBRARY_PATH, TCL_LIBRARY, and PATH to run NS2 smoothly. For that we must modify the BASHRC file for current user.

1) Create folder in your home directory for installation of packages required by video transmission over wireless network and this folder in path variable for executing tools from anywhere.
   #mkdir /home/bishnoi/UtilityProgram/
2) Edit .bashrc file and add NS2 environment variables at the end of file.
   $gedit ~/.bashrc

```
……
……
export CC=gcc34 CXX=g++34
### LD_LIBRARY_PATH ###
OTCL_LIB=/home/bishnoi/ns-allinone-2.29/otcl-1.11
NS2_LIB=/home/bishnoi/ns-allinone-2.29/lib
TCL_LIB=/home/bishnoi/ns-allinone-2.29/tcl8.4.11/unix
TK_LIB=/home/bishnoi/ns-allinone-2.29/tk8.4.11/unix
X11_LIB=/usr/X11R6/lib
USR_LOCAL_LIB=/usr/local/lib
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH: \
        $OTCL_LIB:$NS2_LIB:$TCL_LIB:$TK_LIB:\
        $X11_LIB:$USR_LOCAL_LIB
### TCL_LIBRARY ###
TCL_LIB=/home/bishnoi/ns-allinone-2.29/tcl8.4.11/library
USR_LIB=/usr/lib
export TCL_LIBRARY=$TCL_LIB:$USR_LIB
```

```
### PATH ###
XGRAPH=/home/bishnoi/ns-allinone-2.29/bin:\
      /home/bishnoi/ns-allinone-2.29/tcl8.4.11/unix: \
      /home/bishnoi/ns-allinone-2.29/tk8.4.11/unix:\
     /home/bishnoi/ns-allinone-2.29/xgraph-12.1
NS=/home/bishnoi/ns-allinone-2.29/ns-2.29/
NAM=/home/bishnoi/ns-allinone-2.29/nam-1.11/
PATH=$PATH:$XGRAPH:$NS:$NAM:\
        /home/bishnoi/UtilityProgram/
```

3) For taking immediate effect we save environment variable into Kernal. Otherwise restart the system.
   $source ~/.bashrc

## 2.4 Verifying the Successful installation NS2 and Environment Variables

After setting environment variables we can check Network Simulator (ns) and Network Animator (nam) running successfully or not.

1) Type ns command from any directory, if it display % sign then it work fine.
   $ns
   %
   Output of ns command shown in figure 2.

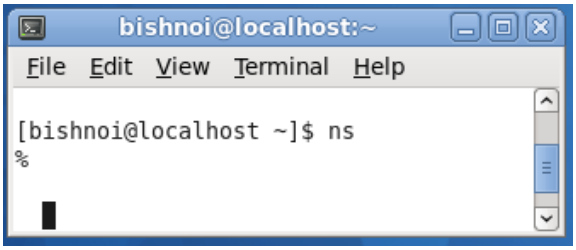


**Fig. 2: Output of ns command**

2) Type nam command from any directory, if it display nam window then it work fine.
   $nam
   Output of nam command shown in figure 3.

# 3. INSTALLING WINDOW ENVIRONMENT

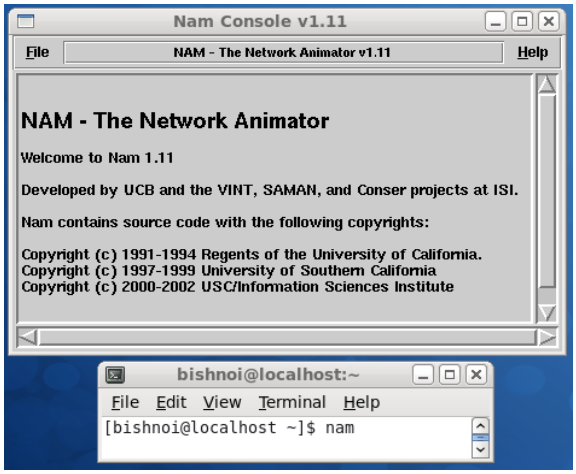


**Fig. 3: Output of NAM command**

Although the Fedora and NS2 are freeware and run almost all tool sets required by simulator for video transmission over wireless. All tools source or binary code are freeware and available over net such as EvalVid framework [7]. But some of them had written in visual C++ software for Microsoft Window environment such as mpeg4encoder, mpeg4decoder are not running directly in Fedora environment. So we require installing more freeware software to run window software on other operating system environment such as Fedora in known as wine [8]. It will run the window executable (*.exe) files in Fedora or other environment.

We install this software through yum. The steps for installation of wine software are as follows.

```
$su
Password: [Password]
#yum install apt
#yum install wine
```

# 4. INSTALLING TOOLS FOR MULTIMEDIA

Most of tools such as avgpsnr ,psnr, et , fixyuv, MP4, threshold etc. work normal way without installing extra software such as codec. But ffmpeg, etmp4, MP4Box tools require special codec software and library files installation. The details of these installations are as follows:

## 4.1 Installation of GIT version ffmpeg

FFmpeg has always been a very experimental and developer-driven project. It is a key component in many multimedia projects and has new features added constantly. Development branch snapshots work really well 99% of the time so people are not afraid to use them [9][10].

2) Download the source code of ffmpeg.
   $cd /home/bishnoi/UtilityProgram
   $wget https://github.com/FFmpeg/FFmpeg/archives\
   /master
2) Extract the downloaded source code in UtilityProgram folder.
   $tar xzvf FFmpeg-FFmpeg-n0.8-1774-gf5899f0.tar.gz
3) Change directory to ffmpeg folder.
   $cd FFmpeg-FFmpeg-f5899f0
4) Before installation start, login to super user.
   $su
   Passowrd: [Password]
5) Start configuration of ffmpeg with "--enable-shared" option. This option require for "etmp4" tool.
   #./configure --enable-shared
6) Make and install the code.
   #make; sudo make install
7) Output of ffmpeg command show in figure 4.

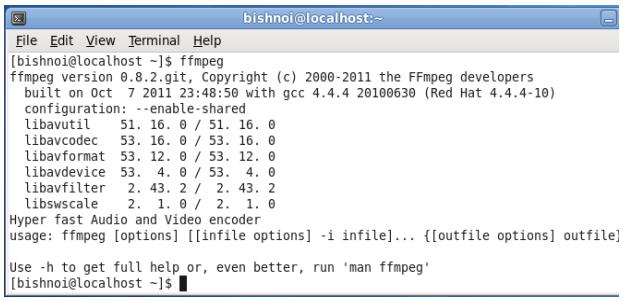


**Fig. 4: Output of ffmpeg command**

## 4.2 Installation of GPAC for MP4Box

GPAC is an Open Source multimedia framework for research and academic purposes. The project covers different aspects of multimedia, with a focus on presentation technologies (graphics, animation and interactivity) [11].

MP4Box is a MP4 multiplexer and used for conversion, splitting, hinting, dumping and to import MPEG-4 video, DivX, h264, XviD, 3ivx or FFmpeg and audio streams into the .MP4 container. The end result is a compliant MP4 stream.

MP4Box is part of the GPAC Project framework. It is a command line tool, but can be used with graphical user interfaces such as YAMB or my MP4box GUI.

2) Before GPAC installation install freeglut package to your system by suing yum. It is a completely Open Sourced alternative to the OpenGL Utility Toolkit (GLUT) library.
   #yum -y install freeglut
   #yum -y install freeglut-devel
3) Download the source code from [12].
   $cd /home/bishnoi/UtilityProgram
   $wget http://downloads.sourceforge.net/gpac \
   /gpac-0.4.5.tar.gz
   $wget http://downloads.sourceforge.net/gpac\
   /gpac_extra_libs-0.4.5.tar.gz
4) Extract Downloaded source code in UtilityProgram folder.
   $tar -zxvf gpac-0.4.5.tar.gz
   $tar -zxvf gpac_extra_libs-0.4.5.tar.gz
5) Change to folder gpac_extra_libs and copy all files to /gpac/extra_lib folder.
   $cd gpac_extra_libs
   $cp -r * /home/bishnoi/UtilityProgram /gpac/extra_lib
   $cd ..; cd gpac
6) Login to super user mode.
   $su
   Password: [Password]
7) Change the access permission of configure file.
   #chmod 755 configure
8) Start configuration with PIC mode for "etmp4" tool.
   #./configure --enable-pic
9) Make and install the library and application.
   #make lib
   #make apps
   #make install lib
   #sudo make install
10) Copy gpac library file to /usr/lib/ folder.
    #cp bin/gcc/libgpac.so /usr/lib
    #ldconfig
11) Copy the header file folder from /gpac/include/ to /usr/include/ . Because these header files require by "etmp4" tool.
    #cp –r /home/bishnoi/UtilityProgram/gpac/include/* \
    /usr/include/
12) Still MP4Box gives error while running shared libraries. To remove this error we apply following command and change SELINUX to disabled.
    #vi /etc/sysconfig/selinux
    #SELINUX=enforcing
    SELINUX=disabled
    #chcon -t texrel_shlib_t /usr/local/lib/libgpac.so

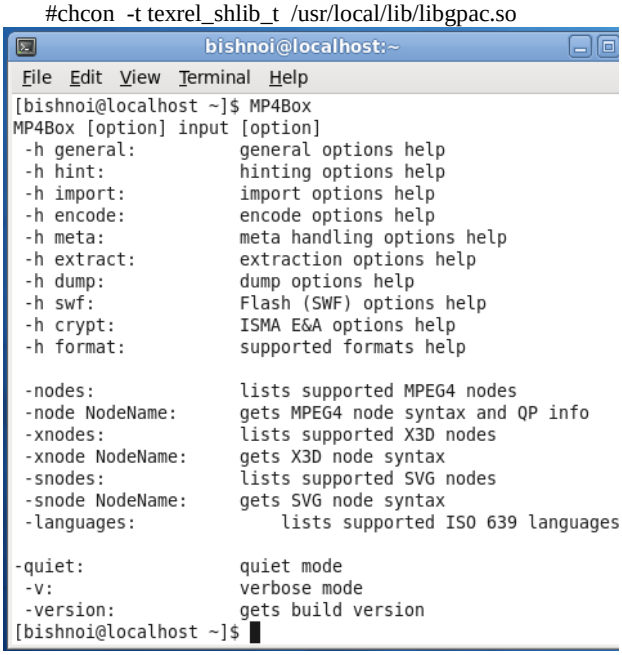


**Fig. 5: Output of MP4Box command**

13) Install libgpac.so.0 library file for GPAC environment.
    #wget ftp://rpmfind.net/linux/rpmfusion/free/fedora/ \
    \releases/12/Everything/i386/os/ \
    gpac-libs-0.4.6-0.2.cvs20090919.fc12.i686.rpm
    #gpac-libs-0.4.6-0.2.cvs20090919.fc12.i686.rpm
14) Output of MP4Box command shown in figure 5.

## 4.3 Installation of etmp4

It uses as reconstruction of the transmitted video as it is seen by the receiver. For this, the video and trace files are processed by etmp4 (Evaluate Traces of MP4-file transmission) [14][15]. This generates a video file, where all frames that got lost or were corrupted are deleted from the original video track. Installation steps are as follows:

1) Configure ffmpeg with enabled shared library and GPAC with pic support as mention in above step A and B respectively. Also copy gpac header file from /gpac/include/ subfolder to /usr/include/ as mentioned in step B.
2) Download EvalVid source code [16].
3) Extract the evalvid archive. Uncomment u64 typedef in types.h
4) Edit Makefile. Change -lgpac_static to -lgpac.
5) $make; sudo make install

6) $export MALLOC_CHECK_=0
7) Run etmp4. The output of command as shown in figure 6.

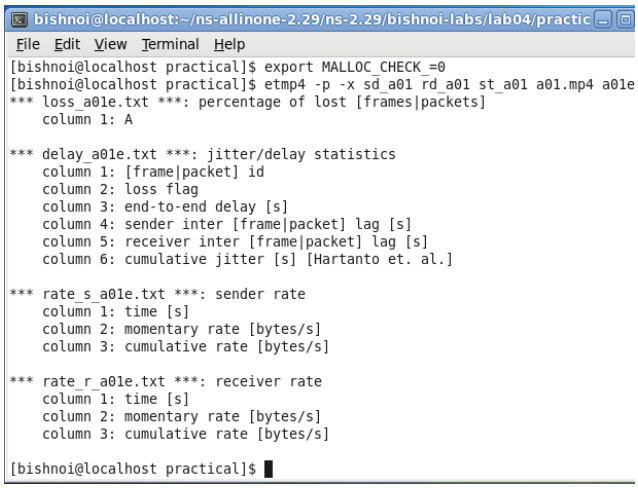


**Fig. 6: Output of etmp4 command**

## 4.4 Installation of MPlayer

For running the multimedia files we require MPlayer tool [17]. The steps of installation as follows:

#su

Password: [Passwrd]

#svn checkout svn://svn.mplayerhq.hu/mplayer/trunk mplayer

#cd mplayer

#./configure

#make; make install

# 5. SIMULATION RESULT

We simulate multimedia over wireless network on Fedora simulation environment. In this simulation we uses network simulator 2.29.3. Simulation topology shown in figure 7.

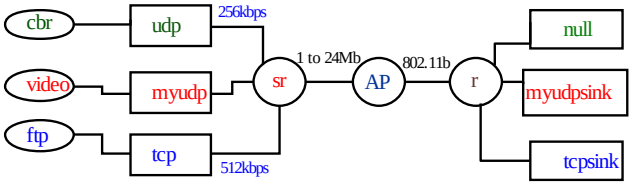


**Fig. 7: Simulation topology**

In the Fedora simulation environment, we uses three traffics are: cbr, video, and ftp. The cbr traffic is transmitted over udp protocol at the data rate is 256kpbs and the packet size is 1500 bytes. The video server transmits video stream over the Internet and wireless links to reach the video receiver. The ftp traffic is transmitted over tcp protocol at the data rate is 512kbps and the packet size is 1500 bytes. The video delivered is foreman_qcif for simulation. The maximum transmitting packet size is 1000 bytes. The link between the base station and the video receivers is IEEE 802.11b 2Mbps. We simulate above topology for link between the video server and the base station has a 1to24Mbps bandwidth and 10 ms latency.

The over objective is to evaluate the video transmission of an MPEG stream and considering average PSNR for the interaction of bandwidth 1 to 24 Mb and queue size 10 to 100. The simulation result graph as shown in figure 8 .

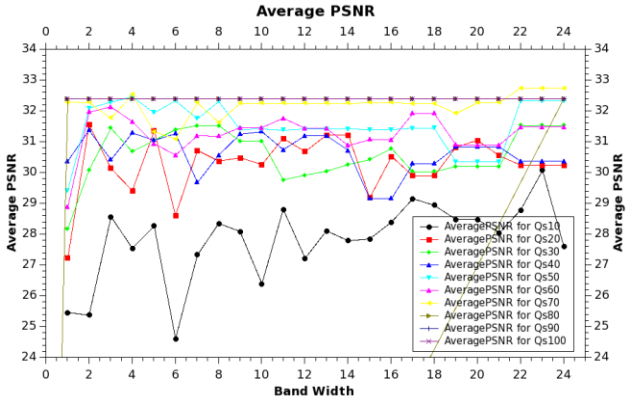


**Fig. 8: Simulation results**

# 6. CONCLUSION

Unlike Cygwin window simulation environment Fedora environment is better for video transmission over wireless network. It provides all Linux facility and we can simulate large file also. In this environment all software's and tools are freeware including Fedora operating system itself. It also capable to run window executable files those code written in Visual C++ under wine environment.

Maximum average PSNR is 32.739773 on queue size 70 and band width 22. There is very slit changes in average PSNR on queue size 80 on 2 to 24 bandwidth. Most of the time is remains 32.394766. Similarly in queue size 70 average PSNR also give better result. It remains approximately constant after bandwidth 8 to 24. The average PSNR is very low on queue size 10. It also observed that after band width 20 the average PSNR remains constant. Finally we reach in this conclusion that the queue size 80 gives excellent results at all type of bandwidth and maximum average PSNR of our topology is 32.739773.

# 7. REFERENCES

[1] Jianli Pan, A survey of Network Simulations Tool: Current Status and future Development. Email: jp10@cse.wustl.edu

[2] Stuart Kurkowski Tracy Camp Neil Mushell Michael Colagrosso, "A Visualization and Analysis Tool for NS-2 Wireless Simulations: iNSpect", IEEE Proceedings MASCOTS'05, 2005

[3] Cygwin, available at http://www.cygwin.com

[4] Source forge is available at http://sourceforge.net/projects/nsnam/files/ allinone/ns-allinone-2.29/ns-allinone-2.29.3.tar.gz/download

[5] http://www.linuxquestions.org/questions/attachment.php?attachmentid=3326&d=1271149430

[6] "EvalVid - A Video Quality Evaluation Tool-set", http://www.tkn.tu-berlin.de/research/evalvid/

[7] Wine HQ, http://www.winehq.org/

[8] https://github.com/FFmpeg/FFmpeg

[9] GPAC,http://gpac.wp.institut-telecom.fr/

[10] Source forge is available at http://sourceforge.net/projects/gpac/files/

[11] Telecommunications network group, available at http://www.tkn.tu-berlin.de/research/evalvi/

[12] Chih-Heng Ke, Chia-Yu Yu, Bilal Munior, "Overview of Evaluation Framework", http://hpds.ee.ncku.edu.tw/~yufrank/YCY/simulator_overview.htm

[13] Chih-Heng Ke, "myEvalvid-NT (myEvalvid Network Trace)" http://140.116.72.80/~smallko/ns2/myEvalvidNT.htm

[14] Telecommunications network group, available at http://www.tkn.tu-berlin.de/research/evalvid/

[15] mplay is available at http://www.mplayerhq.hu/design7/news.html