

# NSG: simulation file generation software

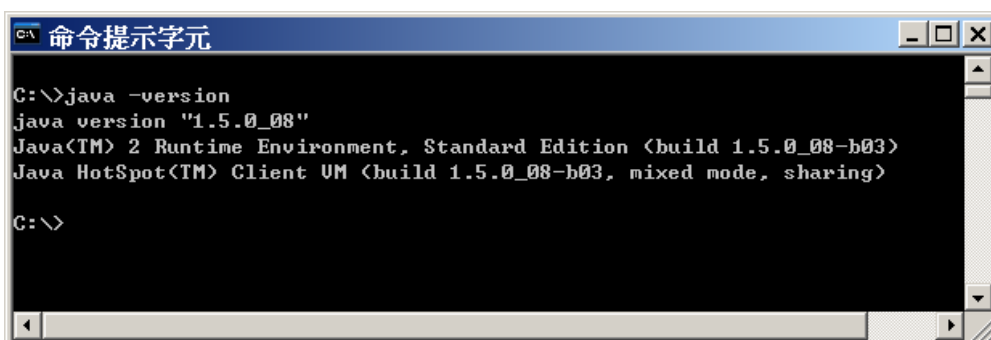
## NSG Introduction and Installation

The NSG is designed for ns2 script generator design, the latest version of the second edition of (NSG2) and the previous version (NSG1) comparison, the biggest difference is that the NSG2 support in addition to the originally supported Wi-Fi Internet wired network script, object types are listed below NSG2 main features and support ns2 :

- 1.Generate script of a wired network and wireless network (wireless and wired mixed script is not supported)
- 2.The Support Node types : WirelessNode, and WiredNode
- 3.Support mobile WirelessNode settings
- 4.The Support Link Type : Duplex-link and Simplex-link
- 5.Agent Support (TCP, TCP / Tahoe, TCP / Reno, TCP / Newreno, TCP / Vegas, TCPSink, UDP and NULL)
- 6.Application types supported: CBR and FTP
- 7.The routing protocol types supported : DSDV, DSR, AODV and TORA
- 8.Support Queue Type : DropTail, RED, FQ, DRR, SFQ and CBQ
- 9.Wireless network-related settings
- 10.802.11 parameter setting the (NSG provides a set of default parameters 802.11b)

NSG is written in Java, so you can run on any supported Java platform (such as Windows, Linux, Mac OS ...) In addition NSG2 based on the java jar file to publish in the form of so NSG2 not need to install, just download NSG2 and save it to your computer be able to Java to start. Download URL is as follows -NSG2 [Download URL](https://sites.google.com/site/pengjungwu/) : <https://sites.google.com/site/pengjungwu/>

In addition, before the implementation of the NSG2 must make sure the computer is equipped with J A V A Runtime Environment (JRE) or Java SE Development Kit (JDK), Window operating system, for example, to determine whether the computer has been installed Java in the DOS command column, enter " java-version " Instructions, if the Java installed on your computer screen will show the version of Java installed (see Figure 1).



```
C:\>java -version
java version "1.5.0_08"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.5.0_08-b03)
Java HotSpot(TM) Client VM (build 1.5.0_08-b03, mixed mode, sharing)

C:\>
```

Figure 1. Confirm the version of Java in your computer

If the computer is not already installed JRE or JDK on Sun's Java Web site to download the latest version of the JRE or JDK installation steps described on the website.

Sun's Java official website : <http://java.sun.com/javase/downloads/index.jsp>

To start NSG2 in the DOS command line switch to the storage directory of the NSG2 program (in this case in NSG2 is stored in the C: \ NSG2 \ directory) Input the java-jar nsg2.jar is to (refer to Figure 2).



Figure 2. Execution of NSG2

## The basic concepts of NSG

The NSG to generate a script file is very simple in design NSG is to ns2 script file that allows users to generate the most natural and intuitive way, so as to "what you see is what you get" approach to design, means that most of the users were able to directly see on the screen Node position, as well as other relevant information (coordinates, connections, and so on ...), users can script generated for their own needs to go for appropriate After the changes, archiving script file (TCL) are then referred the ns2 to perform can use very simple, if you can grab a few key points, you can easily operation NSG2 of.

Following simple description will for several NSG2 of basic concepts in the NSG in the most convenient place is generated the Node coordinates in the wireless network, the author's experience, writing the ns2 wireless network script file must be (contains the x, y, z) coordinates of each Node Write a script file will become quite cumbersome, so as to generate a large and regular Node, which is the original author wants to write a tool to assist in the use of The reason of the generated script file, to generate the coordinates of the Node NSG 2 is a fairly easy thing to do.

The NSG, is divided into five modes, these modes are mostly designed in accordance with the main ns2 several object to the relationship between each object can refer to Figure 3. To switch mode, simply press the switch button to the top of the screen the NSG in 2 contains the following five modes:

- 1.Hand Mode
- 2.Node Mode
- 3.Link mode (only applies to the wired network screenplay)
- 4.Agent mode

## Application mode

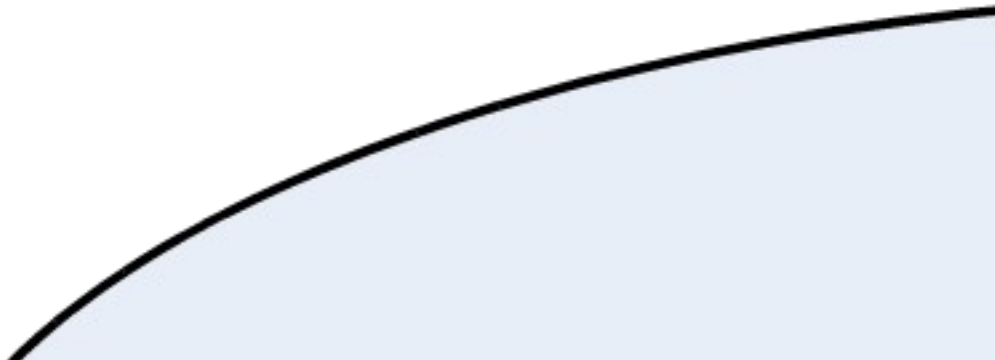


Figure 3 NSG2 main object relations

### Hand Mode

In the Hand mode mode is the most simple, is designed to allow the user to adjust the range that you want to watch. Build a large script, may contain a large number of Node screen may not fit all objects, you can use Hand Mode to adjust the range that you want to watch, to use as long as you hold down the left mouse button, and then drag can.

### Node Mode

In Node mode mode, the user can configure the Node position, if it is wired network script, then the coordinates of the node is not very important, nothing is set relative to the location, however, the NSG will still set Node location to configure the position in NAM Node.

In the design of wireless network script, Node coordinates becomes very important, because each Node coordinates will affect the transmission range, signal strength, and routing protocol, the NSG will build a wireless network script information at the bottom of the screen shows the coordinates as a reference. And NSG assumes that the transmission distance of the wireless Internet Node 250m (actual simulation of the effective transmission distance of wireless network according to the set parameters to determine), so When two Node transmission distance within 250m, NSG will drawing a line between the two, the user can click Node and drag to adjust the position of the Node to be placed. In addition, the Node mode under multiple Node manner at once, which contains: Chain vertical and horizontal Grid ( $n \times n$ ) and Random, users only need to set the distance between the Node and Node , you can easily produce a large number of regular network topology.

### Link Mode

Link mode only applies to the script of a wired network, a wireless network does not require the setting used to set the link between the Node and Node Link MODE in, for example the Simplex-link or Duplex-link, and set the parameters related to (transmission speed, the type of queue). When you first set the property, want to link two Node can then use the mouse to select, set up, you can also use the right mouse button to modify the parameters of the link or delete the link.

### Agent mode

In the Agent Mode in, the user can set the protocol to be attached on the Node (called Agent ns2), the current NSG Support TCP (contains a variety of deformation) and UDP protocols, the set up is

very simple, as long as Node want to attach, click the left mouse button, and then tap placed Gent A position to the Agent's focus on add-on which Node rather than its location, location set just for the screen display Bale, so the user can position placed in clear easily identifiable location can be.

In addition to setting the Node Agent belongs in Agent Mode can also set the sending and receiving end, just select the two paired Agent can use, Generally in ns2 TCP A Gent must correspond to the TCP -Sink A gent, UDP A Gent corresponds to the Null A Gent, TCP A Gent and UDP A Gent is the sender, and TCP-Sink A gent and Null A gent TCP A Gent and UDP A Gent the receiving end. Pay attention to matching problems in the setting of sender and receiver, for example, two TCP Agent pair, or a TCP A Gent and a UDP A Gent paired up are meaningless.

Set Agent, you can also use the right mouse button to modify the parameters of the Agent to modify or delete an agent, some detailed parameters can not be set directly in the generated Agent, you must first create A Gent, re-use the right mouse button pop-up menu settings, such as TCP congestion window and packet size set.

## **Application mode**

Under the Application mode mode can be used to set the top-level applications, such as CBR and FTP settings as long as you want to attach Agent, click the left mouse button, and then tap to be placed in the position of Application to the same Application of focused on additional Agent rather than in its position, as long as the position setting in the clear position can be easily identified.

And setting the Agent, set the Application, you can use the right mouse button to modify the parameters of the Application or delete Application parameters of some detail in the Application which can not be set directly, you must first create Application, and then use the right mouse button pop-up menu to set the related settings such as CBR Transmission rate and packet size.

## **Parameters**

In addition to the above-mentioned five modes, in addition to the function of parameter set, the script of the wired network, parameter setting contains the simulation time and the trace file name (including the NAM Trace file), set up a wireless network script , the other wireless network (Wireless) and channel characteristics (Channel) settings. Comparison is worth mentioning that, the channel characteristics set NSG 2 provides a set of reference values of the analog 802.11b, however, and is not enabled by default, if the user wants to enable these settings, select the box to the left of the screen can, of course, can also be set according to the demand of the simulation value.

When all setting is completed, press the button TCL TCL script file can be generated, the user can in the script file to make some adjustments after the adjustment is completed, save the file and use ns2 to perform.

**Finally, there are a few precautions:**

1. When setting parameters, if necessary parameters, the user must be provided or the NSG warning will appear; non-essential parameters, you can enter -1, If the parameter is set to -1 Then the TCL script file is generated NSG will ignore this parameter. , NSG not to check the correctness of the parameters, such as the size of the input queue -50 NSG will be "faithful" the value which wrote the script file.
2. When an object is deleted, the object belongs to the object will be one and deleted, such as deleting a Node, all connected to the Node, Link and all attached to the Node Agent will be deleted, and when the Agent is deleted, all attached to the Agent Application are also deleted.
3. Set Agent is not set, the sender (or receiver), TCL script file is generated when the Agent will be ignored, because there will be no effect even if the setting of the Agent is added to the script file

which analog.

## NSG wired network script

Then I will be a simple example to describe how the NSG in a wired network script file, shown in Figure 4 shows the simulated environment.



Figure 4. Wired network topology

Six were placed in a simulated environment Node, the following conditions:

1. n 0 and n 4 is established between a CBR connection to UD P protocol (labeled CBr 0).
2. CBR connection is established between a UDP protocol (labeled as cbr 1) n 1 and n 5.
3. N0 - n2, n1 - n2, n3 - n4, n3 - n5 bandwidth is 2Mbps, the delay time is set to 1 ms.
4. n2 - n bandwidth is set to 1Mbps, the delay time is set to 20ms.
5. All links are Duplex-link, are based DropTail queue management mechanism, and the queue length (Queue limit) are set to 10 packet length.

In Application Settings aspects:

1. the CBr the packet size MTU of 1500 bytes, and to send the data transmission speed of 1Mbps, the simulation time is 1 second to send information to the end of the transfer in the first 4 seconds.
2. CBR 1 the packet size is set to MTU of 1500 bytes, and to send the data transmission speed of 1Mbps, and began sending data in the simulation time is 2 seconds, 3 seconds ends the transmission.
3. The entire simulation time is set to 5 seconds.

Note: Because the ns2 UDP protocol the default packet size of 1000 bytes, that is, when the packet size exceeds 1000bytes when UDP packets will be cut into smaller fragments to send (Segmentation), so to be specified at the time of UDP packet size MTU of 1500 bytes to avoid the upper application packet UDP cutting.

In this experiment, the the two CBR connection will be through n2 - n3 this link and this link bandwidth of 1Mbps, however CBR transmission speed of 1Mbps, which will make the n2 - n3 bandwidth cbr packets share full, therefore when cbr transmission, will find n2 - n3 traffic began to become crowded, over time, will begin when the When n3 queue stuffed, some packets discarded n3, this phenomenon continued until the CBR 1 end of the transmission.

## The actual operation

**Step1:** Add the script of a wired network, users can Scenario → New wired scenario to create (or press the hotkey Ctrl + W), after the establishment of Figure 5 shows, in the case of default, the NSG will switch to Node Mode (Node button will be grayed out), which means that users can establish Node.

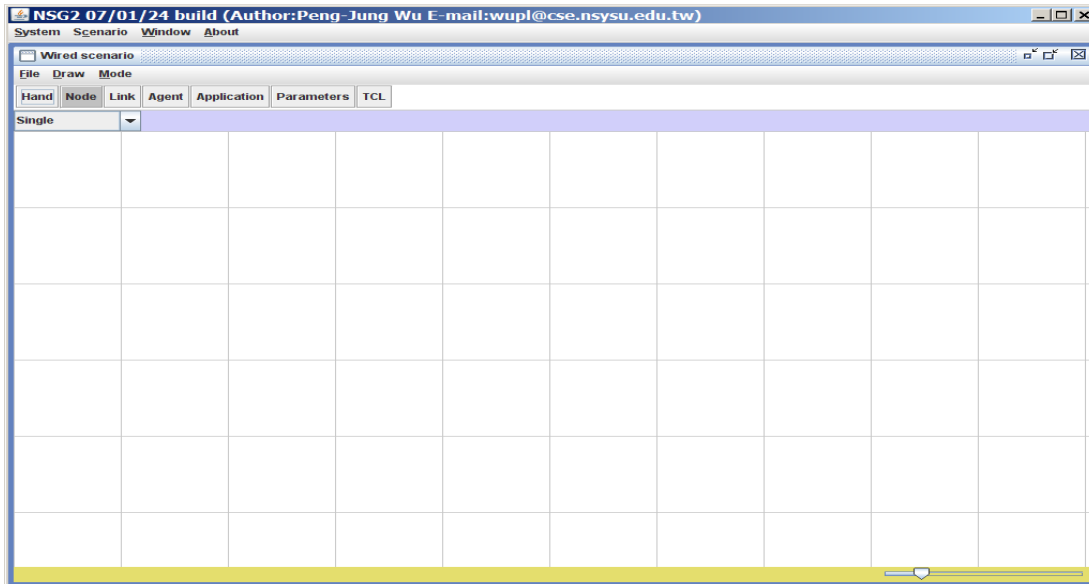


Figure 5. Added a wired network script

**Step2:** Start create a script Node, use the mouse to click on the location you want to display Node to establish Node In this example, the establishment of six Node its relative position in Figure 6, by adjusting the screen bottom right of the crossbar different proportions.

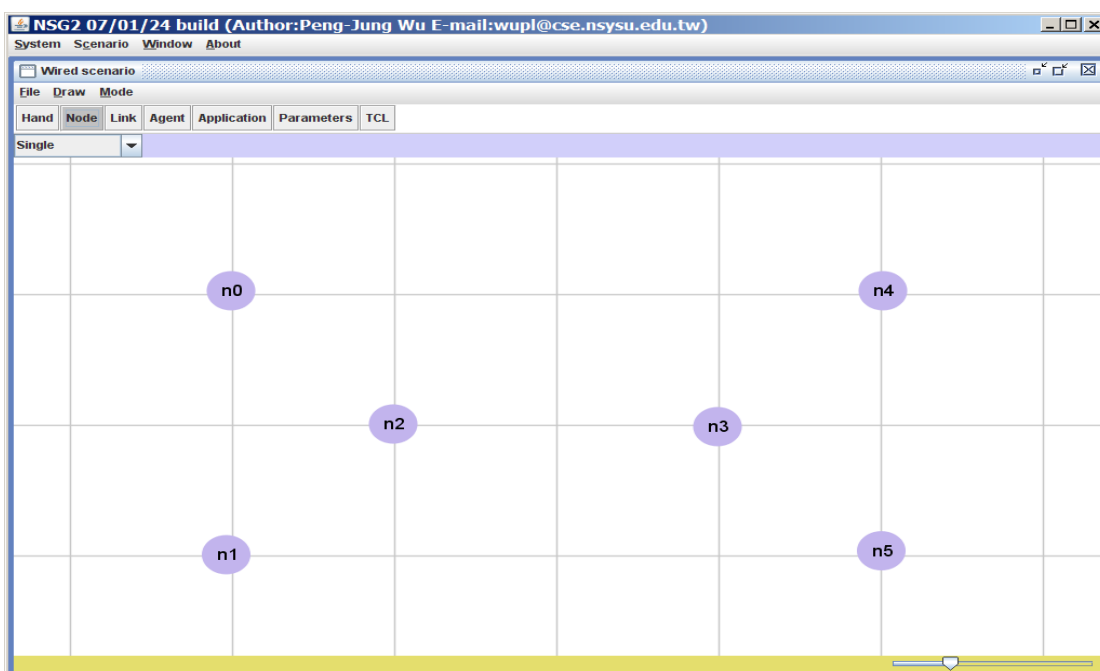


Figure 6: set a good six node position

**Step3:** Set the Node link between the first switch to the Link mode (press the Link button or hotkey Ctrl +3), then "set the following parameters.

- 1.Link-type: duplex-link
- 2.Queue type: DropTail
- 3.Capacity: 2Mbps
- 4.Propagation delay: 10ms
- 5.Queue Size: 10

**Step4:** click n0 - n2, n1 - n2, n3 - n4 and n3 - n5 link between the established, then Capacity set to 1Mbps Propagation delay is set to 20ms, and then create n2 - the link between n3, after the establishment of Figure 7 shows, the process parameter is not set, you can also right-click on the link to be modified.

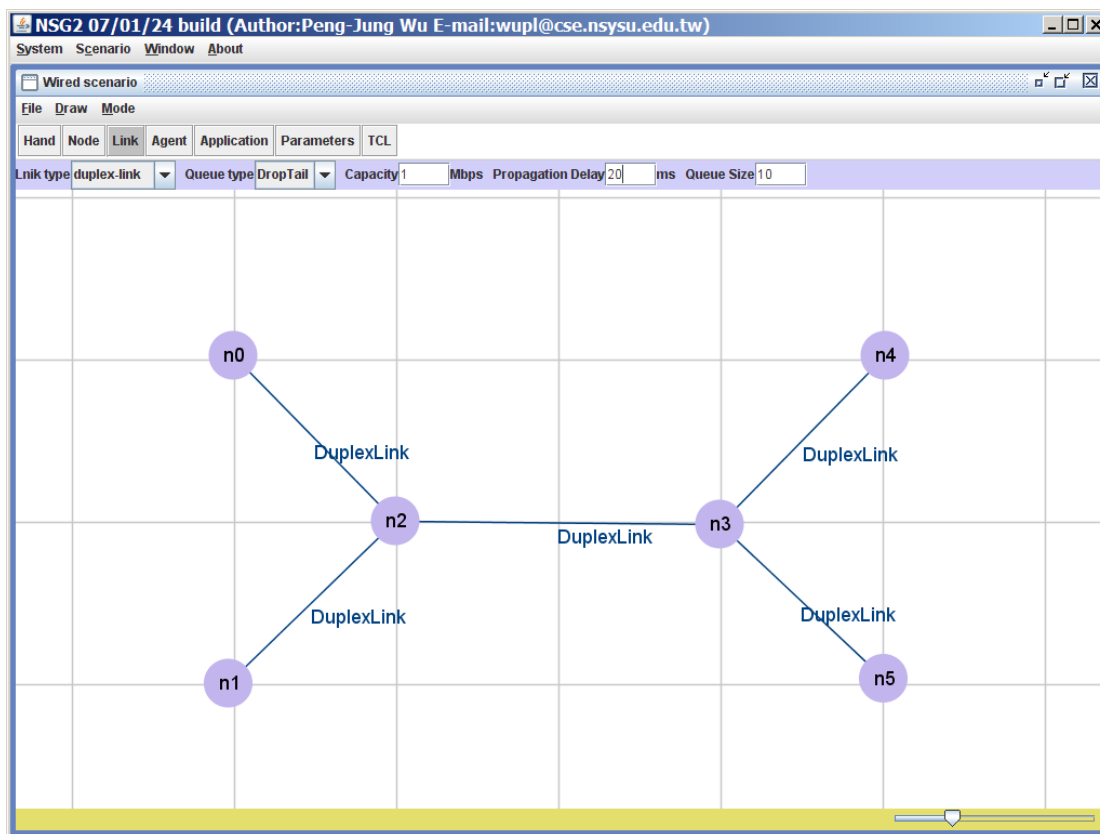


Figure 7: setting a good link between the six node

**Step5:** then switch to the Agent Mode (Press Agent button or hotkey Ctrl +4) to the Protocol setting. First, select the Agent type for UD P and set the packet size (Packet size) to 1500 bytes, and establish a Agent on each of the n0 and n1, then select Agent type to NULL, and to establish a NULL Agent on n4, n5 each. Creating the Agent after which they continue to set up the connection between the Agent, its set way is to use the mouse to select the two agents want to establish a connection, respectively (udp0, null2) and (UDP1-n ull 3) establish a connection between a good shown in Figure 8.

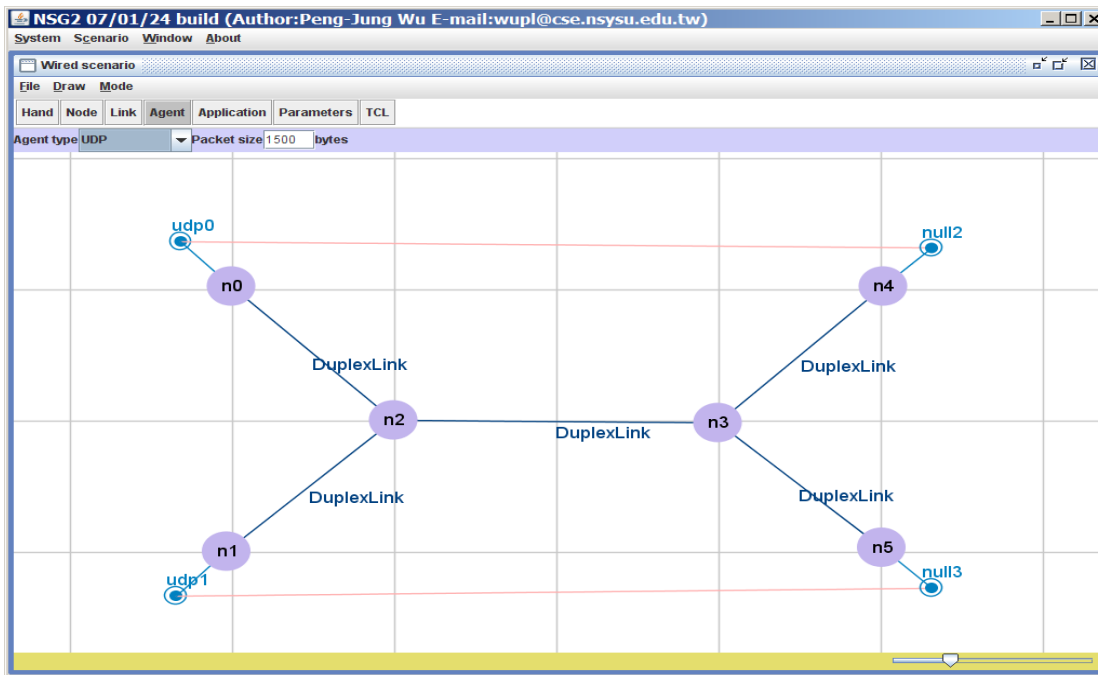


Figure 8: Develop Agents

**Step6:** switch to Application Mode (press the Application button or hotkey Ctrl +5) to the top of the application settings, Application type set for the CBR and set the start time (Start time) of 1 second, the end of time (Stop time) for 4 seconds, the packet size is 1500bytes, transmission speed to 1Mbps, and then create a CBR Application n0, followed by the start time (Start time) is 2 seconds, the end time (stop time) is 3 seconds, and n 1 on then create a CBR Application, as shown in Figure 9.

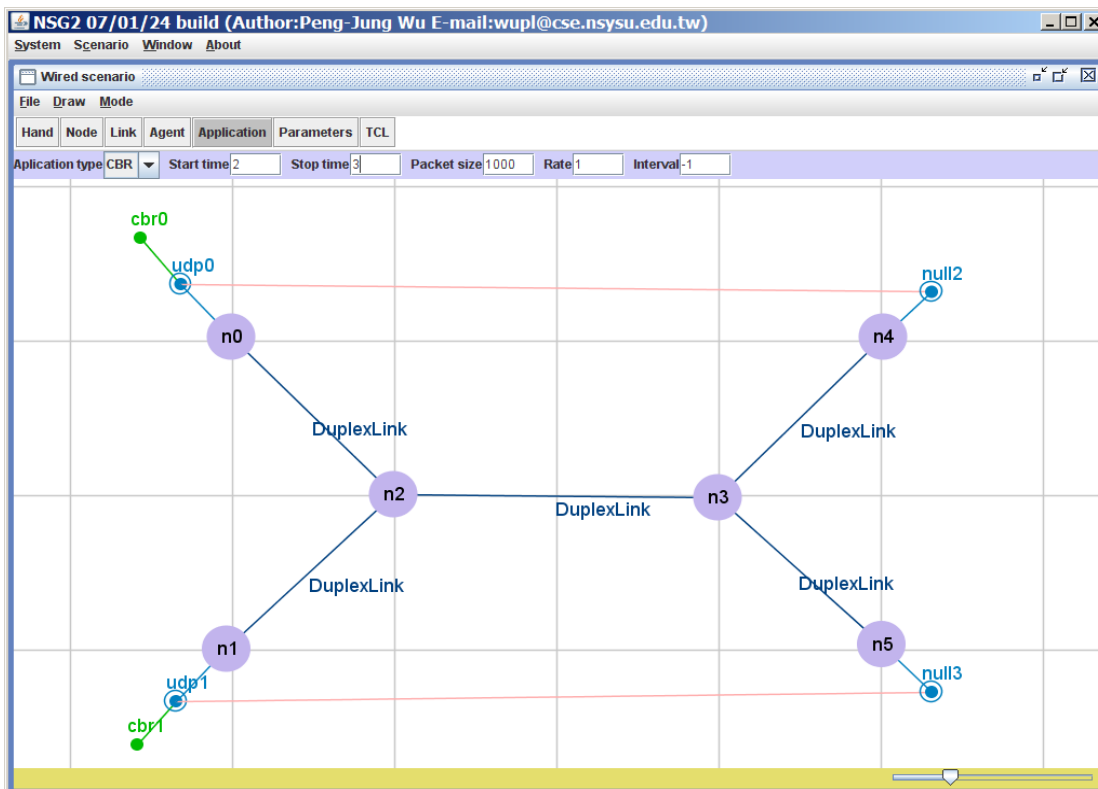


Figure 9: Create Application



**Step7:** Finally, set up the simulation parameters (Parameters "button is pressed or the hotkey Ctrl +6), in this case, because it is a wired network script, so part of the parameters only need to simulation time (Simulation time ) is set to 5.0, and specify a trace file name can, as shown in Figure 10.

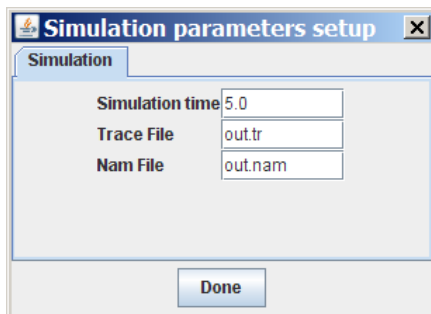


Figure 10: Setting the simulation parameters

**Step8:** When everything is set after TCL button is pressed (shortcuts Ctrl +7) to generate a script file, every script file will be created in a separate window, user can also according to their own correction needs to do, as shown in Figure 11 above.

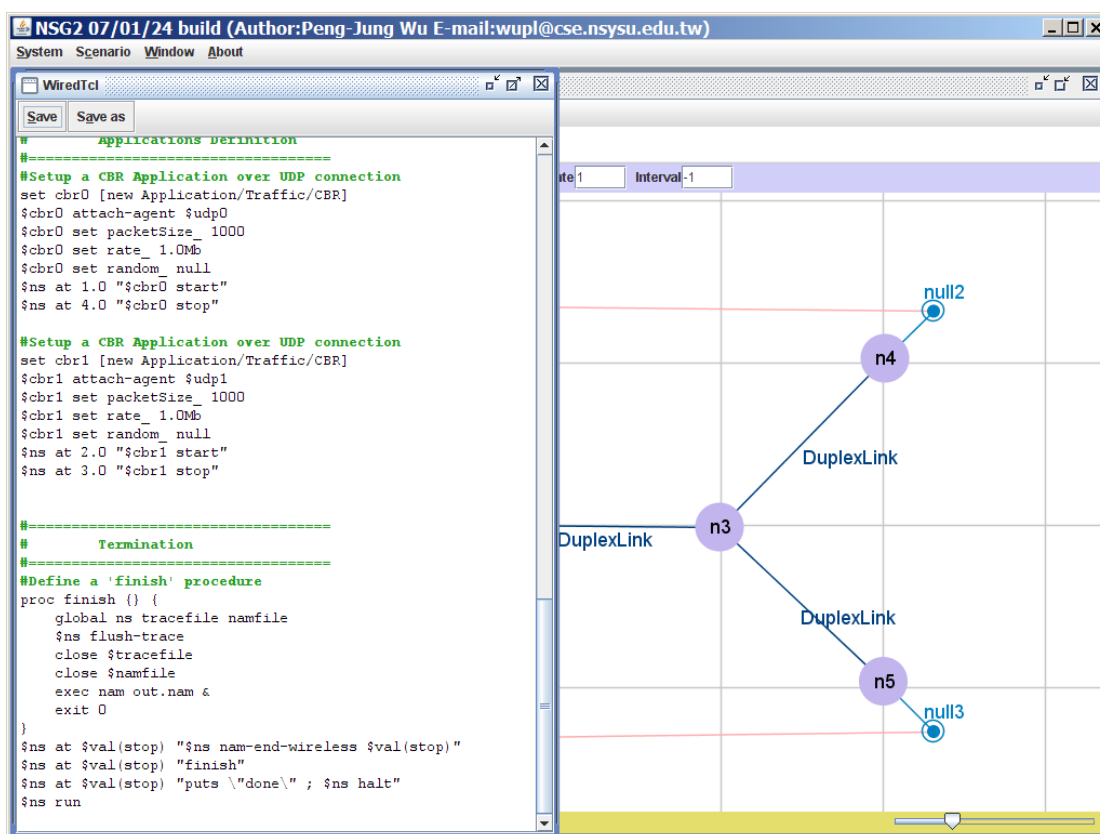


Figure 11. Generate a script file

The TCL file **Step9:** Press the Save button (shortcut key Alt + S) generated stored can be handed over to the ns2 to start the simulation, arising out of the script file are as follows .

```
# This script is created by NSG2 beta1
# <http://wushoupong.googlepages.com/nsg>
```

```
# =====  
# Simulation parameters setup  
# =====  
set val (stop) 5.0; # time of simulation end  
  
# =====  
# Initialization  
# =====  
# Create a ns simulator  
set ns [new Simulator]  
  
# Open the NS trace file  
set tracefile [open out.tr w]  
$ Ns trace-all $ tracefile  
  
# Open the NAM trace file  
set namfile [open out.nam w]  
$ Ns namtrace-all $ namfile  
  
# =====  
# Nodes Definition  
# =====  
# Create 6 nodes  
set n0 [$ ns node]  
set n1 [$ ns node]  
set n2 [$ ns node]  
set n3 [$ ns node]  
set n4 [$ ns node]  
set n5 [$ ns node]  
  
# =====  
# Links Definition  
# =====  
# Createlinks between nodes  
$ Ns duplex-link $ n0 $ n2 2.0Mb 10ms DropTail
```

```

$ Ns queue-limit $ n0 $ n2 10
$ Ns duplex-link $ n2 $ n1 2.0Mb 10ms DropTail
$ Ns queue-limit $ n2 $ n1 10
$ Ns duplex-link $ n4 $ n3 2.0Mb 10ms DropTail
$ Ns queue-limit $ n4 $ n3 10
$ Ns duplex-link $ n3 $ n5 2.0Mb 10ms DropTail
$ Ns queue-limit $ n3 $ n5 10
$ Ns duplex-link $ n2 $ n3 1.0Mb 20ms DropTail
$ Ns queue-limit $ n2 $ n3 10

# Give node position (for NAM)
$ Ns duplex-link-op $ n0 $ n2 orient right-down
$ Ns duplex-link-op $ n2 $ n1 orient left-down
$ Ns duplex-link-op $ n4 $ n3 orient left-down
$ Ns duplex-link-op $ n3 $ n5 orient right-down
$ Ns duplex-link-op $ n2 $ n3 orient right

# =====
# Agents Definition
# =====

# Setup a UDP connection
set udp0 [new Agent / UDP]
$ Ns attach-agent $ n0 $ udp0
set NULL2 [new Agent / Null]
$ Ns attach-agent $ n4 $ null2
$ Ns connect $ udp0 $ null2
$ Udp0 set packetSize_ 1500

# Setup a UDP connection
set udp1 [new Agent / UDP]
$ Ns attach-agent $ n1 $ udp1
set null3 [new Agent / Null]
$ Ns attach-agent $ n5 $ null3
$ Ns connect $ udp1 $ null3
$ Udp1 set packetSize_ 1500
# =====

```

```

# Applications Definition
# =====
# Setup a CBR Application over UDP connection
set cbr0 [new Application / Traffic / CBR]
$ Cbr0 attach-agent $ udp0
$ Cbr0 set packetSize_ 1500
$ Cbr0 set rate_ 1.0Mb
$ Cbr0 set random_ null
$ Ns at 1.0 "$ cbr0 start"
$ Ns at 4.0 "$ cbr0 stop"

# Setup a CBR Application over UDP connection
set cbr1 [new Application / Traffic / CBR]
$ Cbr1 attach-agent $ udp1
$ Cbr1 set packetSize_ 1500
$ Cbr1 set rate_ 1.0Mb
$ Cbr1 set random_ null
$ Ns at 2.0 "$ cbr1 start"
$ Ns at 3.0 "$ cbr1 stop"
# =====
# Termination
# =====
# Define a 'finish' procedure
proc finish {} {
    global ns tracefile namfile
    $ Ns flush-trace
    close $ tracefile
    close $ namfile
    exec nam out.nam &
    exit 0
}
$ Ns at $ val (stop) "$ ns nam-end-wireless $ val (stop)"
$ Ns at $ val (stop) "finish"
$ Ns at $ val (stop) "puts \" done \"; $ ns halt"
$ Ns run

```

NSG generated script file will automatically start the NAM to render the simulation process, if you do not want the NAM start parameter setting in the NSG, the NAM File option is set to -1, the NSG will not start the NAM instruction write into the script file which. Figure 12 is a the NAM start screen can be seen in the simulation time is 2.1 seconds, n2 phenomenon occurs the packet is discarded while the influx of too packet.

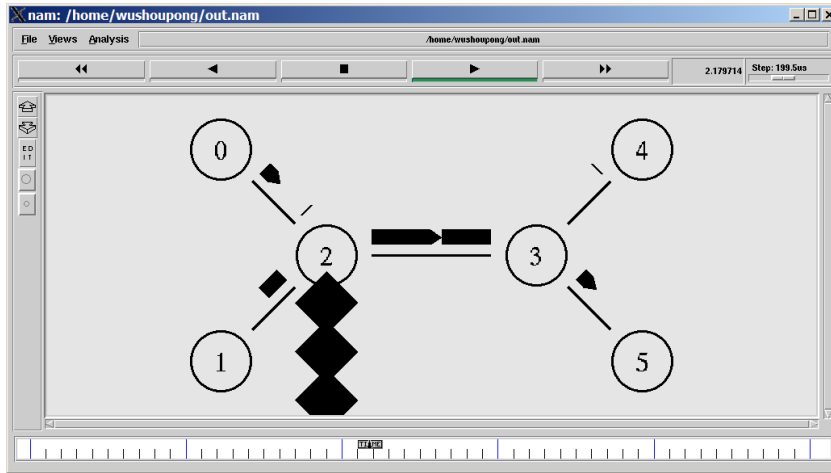


Figure 12

Following the interception of a part of the contents of the Trace file from the the Trace file contents can also be observed phenomenon (d Representative drop) the packet is discarded.

Before slightly

```
r 2.12 3 5 cbr 1500 ----- 0 1.0 5.0 2 88
r 2.12 0 2 cbr 1500 ----- 0 0.0 4.0 92 101
+ 2.12 2 3 cbr 1500 ----- 0 0.0 4.0 92 101
d 2.12 2 3 cbr 1500 ----- 0 0.0 4.0 92 101
- 2.12 2 3 cbr 1500 ----- 0 1.0 5.0 4 92
r 2.124 1 2 cbr 1500 ----- 0 1.0 5.0 9 102
+ 2.124 2 3 cbr 1500 ----- 0 1.0 5.0 9 102
r 2.128 2 3 cbr 1500 ----- 0 1.0 5.0 3 90
+ 2.128 3 5 cbr 1500 ----- 0 1.0 5.0 3 90
- 2.128 3 5 cbr 1500 ----- 0 1.0 5.0 3 90
+ 2.128 0 2 cbr 1500 ----- 0 0.0 4.0 94 105
- 2.128 0 2 cbr 1500 ----- 0 0.0 4.0 94 105
+ 2.132 1 2 cbr 1500 ----- 0 1.0 5.0 11 106
- 2.132 1 2 cbr 1500 ----- 0 1.0 5.0 11 106
r 2.132 3 4 cbr 1500 ----- 0 0.0 4.0 86 89
r 2.132 0 2 cbr 1500 ----- 0 0.0 4.0 93 103
+ 2.132 2 3 cbr 1500 ----- 0 0.0 4.0 93 103
d 2.132 2 3 cbr 1500 ----- 0 0.0 4.0 93 103
- 2.132 2 3 cbr 1500 ----- 0 0.0 4.0 88 93
```

r 2.136 1 2 cbr 1500 ----- 0 1.0 5.0 10 104

+ 2.136 2 3 cbr 1500 ----- 0 1.0 5.0 10 104