Ricean Propagation Module for OPNET

Author: Ratish J. Punnoose
Wireless and Broadband Networking Center
Dept of ECE
Carnegie Mellon Univ.
http://www.ece.cmu.edu/wireless/
http://www.ece.cmu.edu/~wbnc/

INTRODUCTION

This module for Opnet provides a realistic Ricean Fading power envelope at the receiver. The module is a pipeline stage that replaces the default power reception pipeline stage (dra_power). The implementation is based on the publication,

> Ratish J. Punnoose, Pavel V. Nikitin, and Daniel D. Stancil. *"Efficient Simulation of Ricean Fading within a Packet Simulator"*. IEEE Vehicular Technology Conference, Sept 2000.

The power envelope is time-correlated. The model takes as parameters, the Doppler velocity, and the Ricean K factor.

Antenna and Radio Communication Lab

INSTALLATION

File Listing

The following files are provided:

./arc_pwr_ricefading.ps.c	/* Pipeline Stage File */
./rice_table.gdf	/* A table with Ricean data values */
./README.TXT	Directions on installation
./RiceanOpnet.pdf	This file

Compilation

Copy the arc_pwr_ricefading.ps.c to the opnet models subdirectory (~/op_models) or to any other place. You can compile it in one of two ways.

a) Manual

Run ***'op_mko -type ps -m arc_pwr_ricefading'***.

This will create the file arc_pwr_ricefading.s1.ps.o in your ~/op_models directory.

b) Using the OPNET process editor.

Select **File->Open** from the main Opnet window. Instead of "**Project**", select "**Pipeline Stage (C Code)**". If the file arc_pwr_ricefading.ps.c is in the current directory or ~/op_models, it will appear as a possible selection in the file selection list. Select this and click "OK". This will open up an editor window with the source file loaded. Select **File->Compile** from the menu. This will create the compiled version arc_pwr_ricefading.s1.ps.o in your ~/op_models directory.

After compilation refresh Opnet's listing of available models by choosing "**File->Refresh Model Directories**" from the main Opnet menu.

DOCUMENTATION

The Opnet radio model has multiple pipeline stages. For more information on this see the Online Documentation. Look under "**General Models" -> "Chapter 5: Pipeline Stages/Radio Links.**" The radio model pipeline stages model the flow of data from a transmitter to a receiver. Accordingly, there is a Transmission Delay pipeline stage, a Transmitter Antenna Gain stage, a Receiver Antenna Gain stage, a Received Power stage, SNR stage, BER stage, etc. For a complete description, see the Opnet documentation.

The Ricean Fading Power module is a replacement for the "Received Power Model" pipeline stage. The default model "dra_power" uses a free space propagation model. The Ricean Fading generates a time-correlated power envelope obeying the Ricean distribution. It is ordinarily a computationally intensive calculation but the computationally intensive part of the calculation (that is parameter independent) has been pre-computed and stored in the data file (rice_data.gdf). The Ricean fading module takes five parameters.

Parameter Name	**Data Type**	**Possible Values**	**Explanation**
MaxVelocity	`double`	Any positive real value. (Typical 2).	Maximum velocity of the Tx, Rx or other objects in the environment in m/s. This is related to the Doppler velocity f_D. (f_D = MaxVelocity/λ)
Ricean Kfactor	`double`	Any positive real number.	The Ricean K factor (NOT in dB). A value of 0 yields Rayleigh fading
RiceDataFile	`typed file`	rice_table	The name of the file with the Ricean data table. Use the provided rice_table.gdf file
UseTwoRay	`int`	0 or 1	The Ricean model uses a large scale propagation model to compute the average power of the transmitted signal and then provides small scale fading about that average value. The large scale model can be a free-space propagation model or a two-ray model. Possible values are 0,1. If the TMM toolbox is used, the Ricean model uses the TMM toolbox to compute the large scale average power.
TableOffset	`int`	0 to 16,000	An offset into the Ricean data table. Ordinarily, time-index 0 will correspond to the first entry in the table. However, if multiple nodes in the simulation have the same setting, their fading will be correlated (they will undergo similar fading effects

			at the same time). Setting this to a different value for each node will keep the nodes from fading together. The provided data file has 16,384 elements. So if there are two nodes in the simulation fading separately, set one to have a TableOffset of 0, the other at 8192. If there are four such nodes, set the values to 0, 4096, 8192, 12288. etc. etc.

The Ricean Fading module is applied to a Mobile Receiver. To supply parameters to the Ricean Fading Module, use the "Extended Attributes definition" to create attributes (parameter names) for the required Mobile node. Make sure that the Attribute Names have the same case and spelling as shown.

EXAMPLE

Please be sure to read the following Opnet documentation before trying the example.

- Online Documentation -> General Models -> Chapter 5: Pipeline Stages/Radio Links
- Online Tutorial -> Modeler Tutorial -> Mobile Radio

We will start by creating a bare mininum transmitter and receiver model. You don't have to create a receiver model from scratch to use this module. But it is easiest to understand the working of the module with a simple receiver. Lets start with the transmitter. From the **File** menu, choose **New** and then **Node Model** from the list. In the Node Model Editor, create the following structure.

For clarification and more details, please read the Online Mobile Radio tutorial mentioned before. That has a more detailed explanation. The transmitter node has three components.

- A packet source (labelled tx_gen)
- A radio transmitter (radio_tx)
- A transmitting antenna (ant_tx)

Set the following attributes for them. For the packet source, **tx_gen,** change the process_model to simple_source. Set the following values.

Packet Format	NONE
Packet Interarrival Time	constant (0.01)
Packet Size	constant (100)
Start Time	0.1

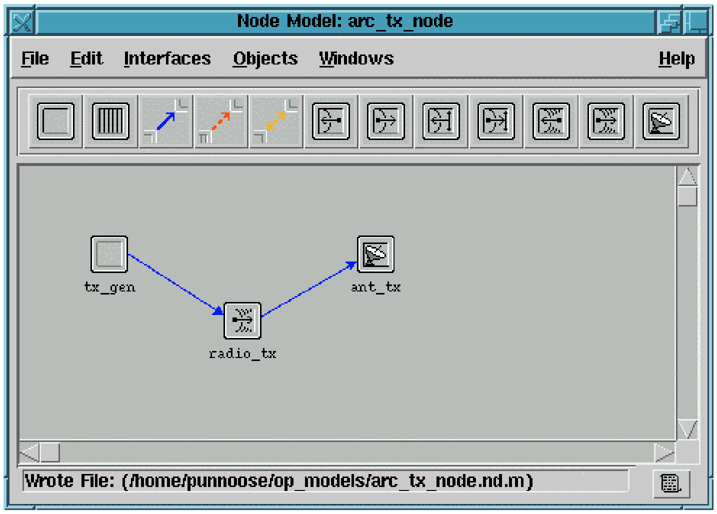

Figure 1: Transmitter Node Model

For the radio transmitter, **radio_tx**, use the default values, but change the channel settings to

data rate (bps)	**packet formats**	**bandwidth (kHz)**	**min frequency**	**spreading code**	**power**
1,000,000	<default value>	20,000	905	disabled	promoted

For the transmitting antenna, use the defaults.

Then, select **Interfaces** from the menu and then **Node Interfaces**. Change the attributes to **`hidden`**, except for **altitude** (**`promoted`**) and **radio_tx.channel[0].power** (**`promoted`**). For **Node Types** in the same dialog box, set **Supported** value for **Satellite** to **no** (See Fig 2). Then save the changes with the name ***arc_tx_node***.

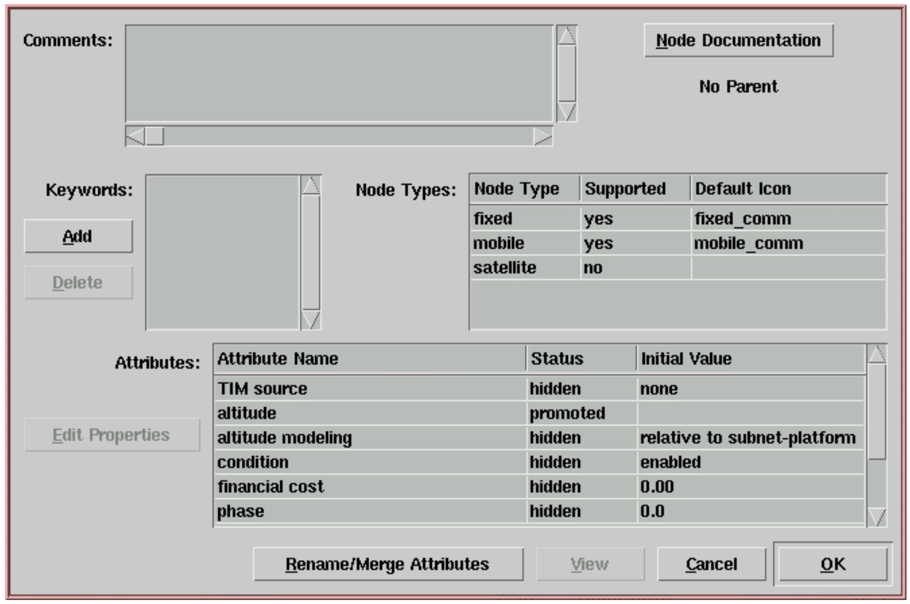

Figure 2: Node Interfaces Dialog Box

Now create a new Node model for the receiver as show in Fig 3. Use the default values for the receiver antenna and receive sink. Feel free to change their names to more meaningful ones.

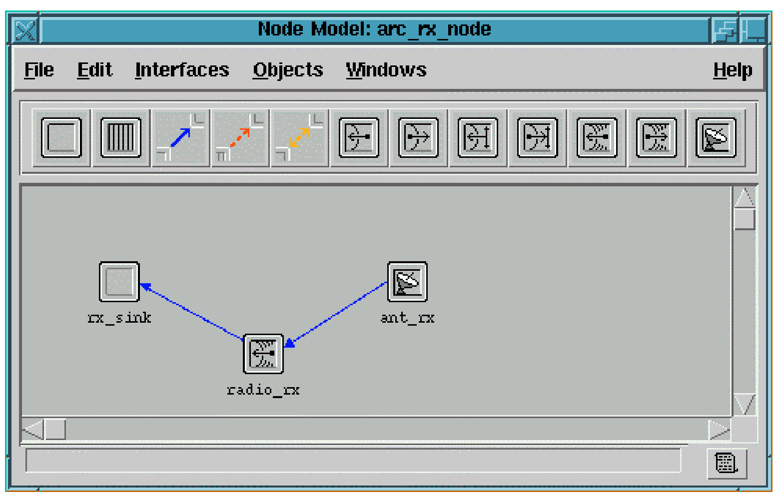

Figure 3: Receiver Node Model

For the receiver radio module (radio_rx), edit the attributes. Change the **channel** parameter to have the same settings as we defined earlier for the transmitter. There is a **processing gain** parameter. You can leave that at the default value. It is not invoked since the **spreading gain** parameter is set to **no**.

Change power model to **arc_pwr_ricefading**. This should be an available option if you have compiled the fading module properly and placed it in ~/op_models. The other parameters can stay at default values. However, we have to provide the parameter values to the fading module. We do that here using the **Extended Attributes** definition.

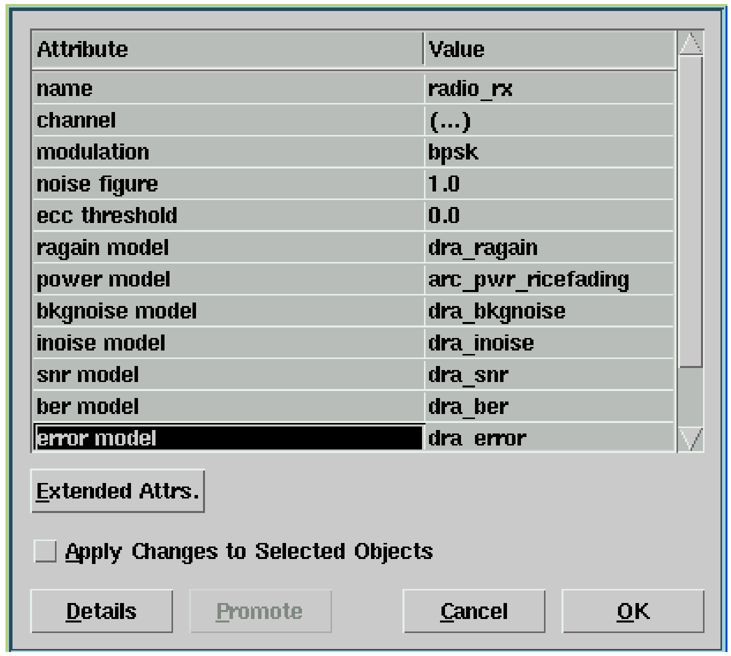

Figure 4: Radio_Rx Attributes

Click **Extended Attrs** from this dialog box and add the parameters needed by the Ricean module as shown in Fig 5.

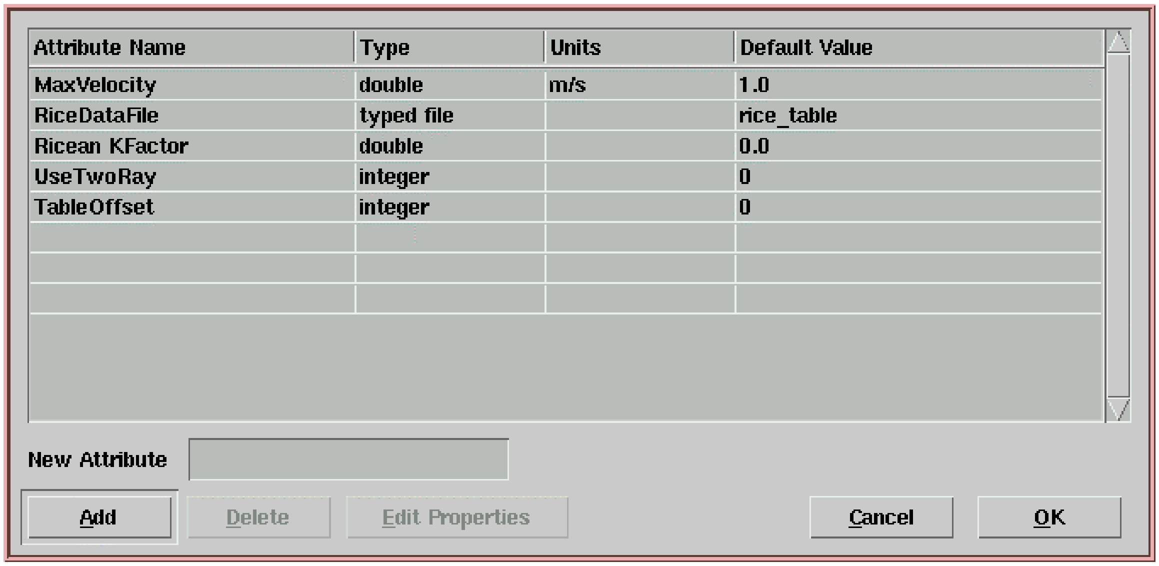

Figure 5: radio_rx Extended Attributes

Once these are added, they will show up as part of the radio node attributes.

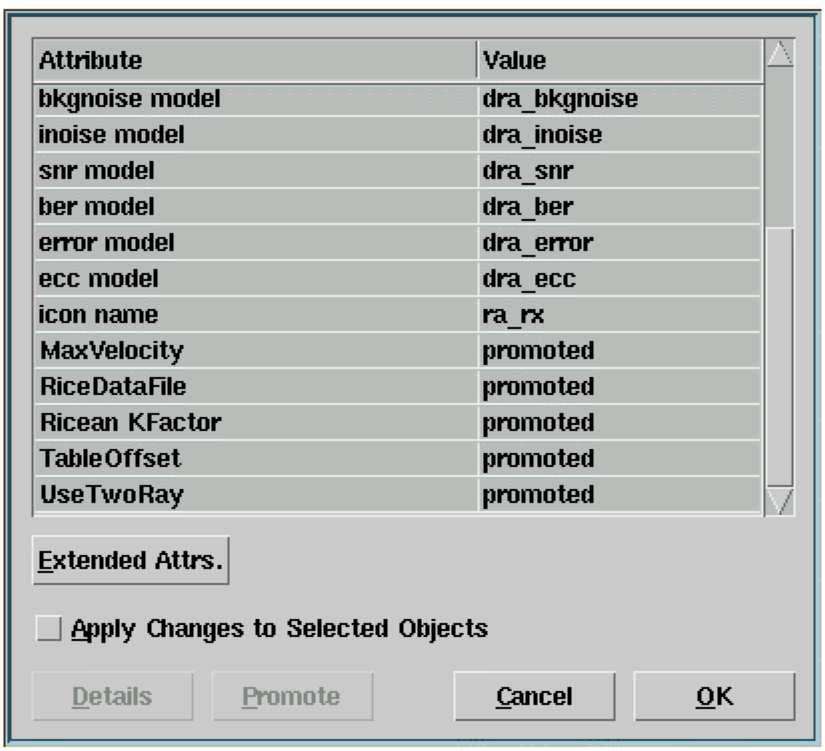

Figure 6: radio_rx attributes with Ricean module parameters

Set the values to promoted, so that they will appear as the simulation parameters. This allows us to run the simulation for different parameters easily. Also define the **Node Interfaces** in the same way that the transmitter model was defined (Fig 2). Save the node model as ***arc_rx_node***.

Now we are ready to use the transmitter and receiver nodes in a simulation.

From the main **File** menu, choose **New**, and then **Project**. Name the project **`arc_net`** and the scneario **`ricefading_test`**. Use the following settings.

Dialog Box Name	Value
Initial Topology	Create Empty Project
Choose Network Scale	Enterprise
Specify Size	5 km x 5 km
Select Grid Properties	1 km
Select Technologies	None
Review	Check values and click OK

In the object palette, click on **Configure Palette**. This brings up a dialog box, click **Clear**. This removes all the objects that we are not going to use in this simulation. Click **Node Models** and change both **arc_tx_node** and **arc_rx_node** to **`included`**. Then the palette looks as shown in Fig 7. Note that each node definiton appears both as a fixed node and a moving node.

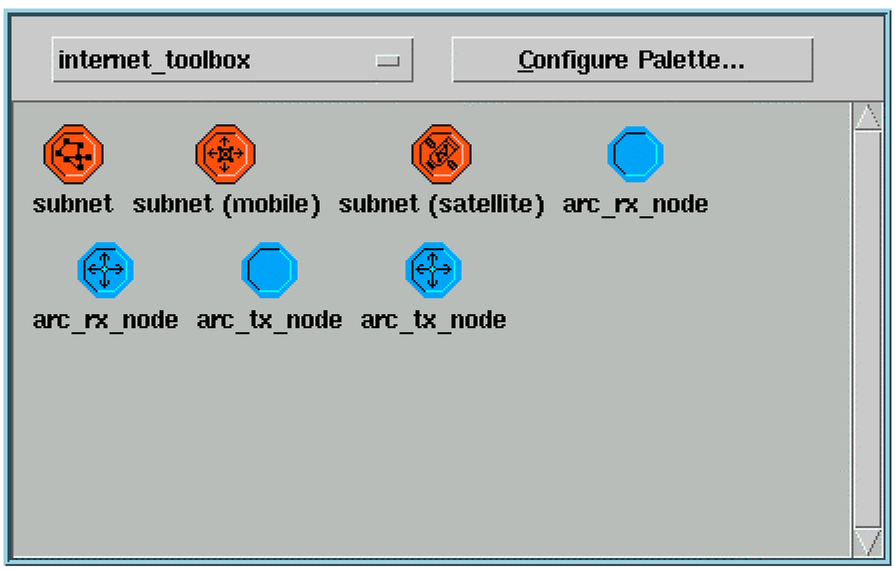

Figure 7: Palette Configuration

Using this palette, create a configuration as shown in Fig 8. Create node **TX** using the **arc_tx_node** and create the **RX1** and **RX2** nodes using the **arc_rx_node**.

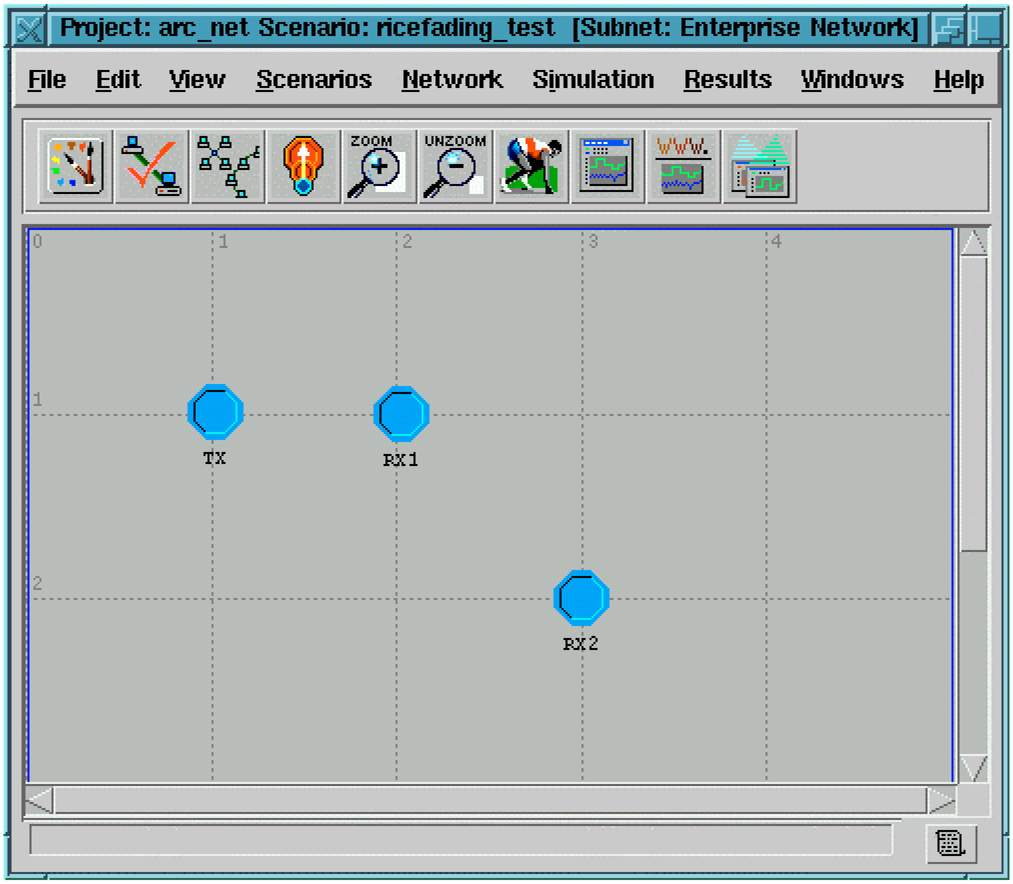

Figure 8: Simulation Scenario

For the TX model, set the attributes as follows.

Attribute	Value
name	TX
model	arc_tx_node
altitude	0.003
radio_tx.channel[0].power	1.0

Similarly set the RX model attributes as follows

Attribute	Value for RX1	Value for RX2
name	RX1	RX2
model	arc_rx_node	arc_rx_node
altitude	0.003	0.003
radio.rxMaxVelocity	1.0	0.5
radio_rx.RiceDataFile	rice_table	rice_table
radio_rx.Ricean KFactor	0.5	0.0
radio_rx.TableOffset	0	8192
radio_rx.UseTwoRay	1	1

The altitude setting is needed since we are using the two-ray propagation model for the average signal power.

Right-click on **RX1** and pick **Choose Individual Statistics** and select **Module Statistics -> radio_rx.channel[0]-> radio receiver -> received power** as show in Fig 9.

Do the same for **RX2**.

Now we are ready to run the simulation. Select **Simulation->Configure Simulation …** from the menu and set the simulation time to 5 seconds.

Then select **Simulation->Run**. The simulation runs and completes. Now we can look at the collected statistics.

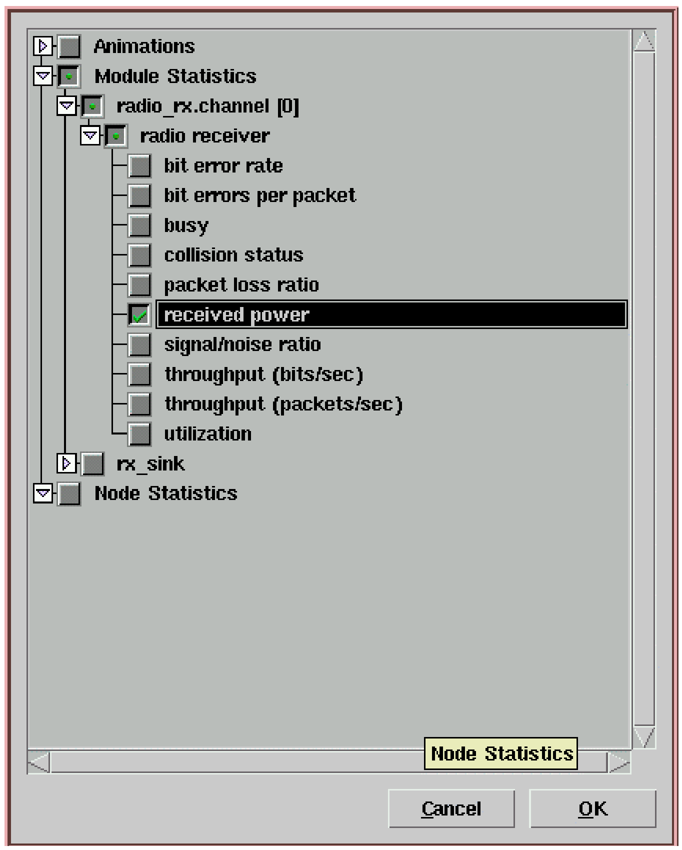

Figure 9: Individual Statistics

Select **Results->View Results** from the main menu, and select the received power for nodes **RX1** and **RX2** under **Object Statistics**.

Once this has been selected (Fig 10), click **Show**.

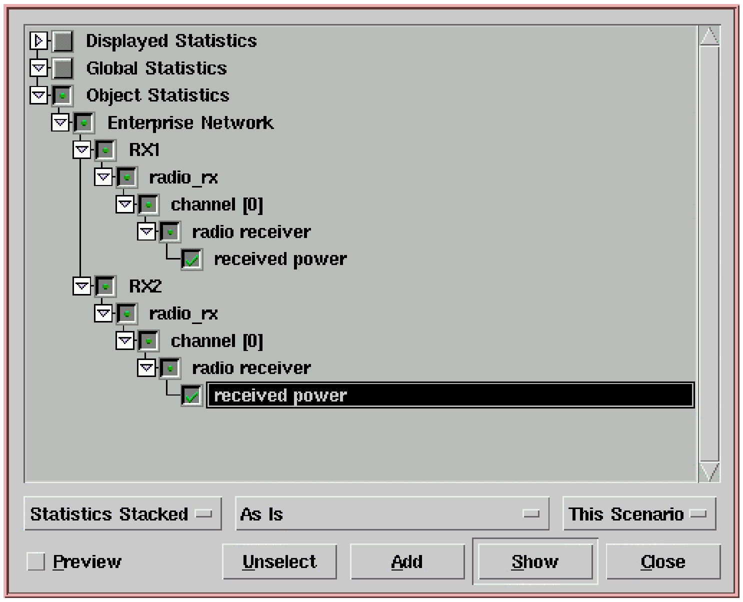

Figure 10: Result Selection

This brings up the received power envelope for the two nodes. This is shown in Fig 11.

The power levels of the received packets have a Ricean fading envelope with the appropriate parameters. Note that one of them fades more slowly than the other. This is because they have different **MaxVelocity** parameters.

These plot values can be exported to a text file and can then be graphed in log-format with Matlab or Excel.

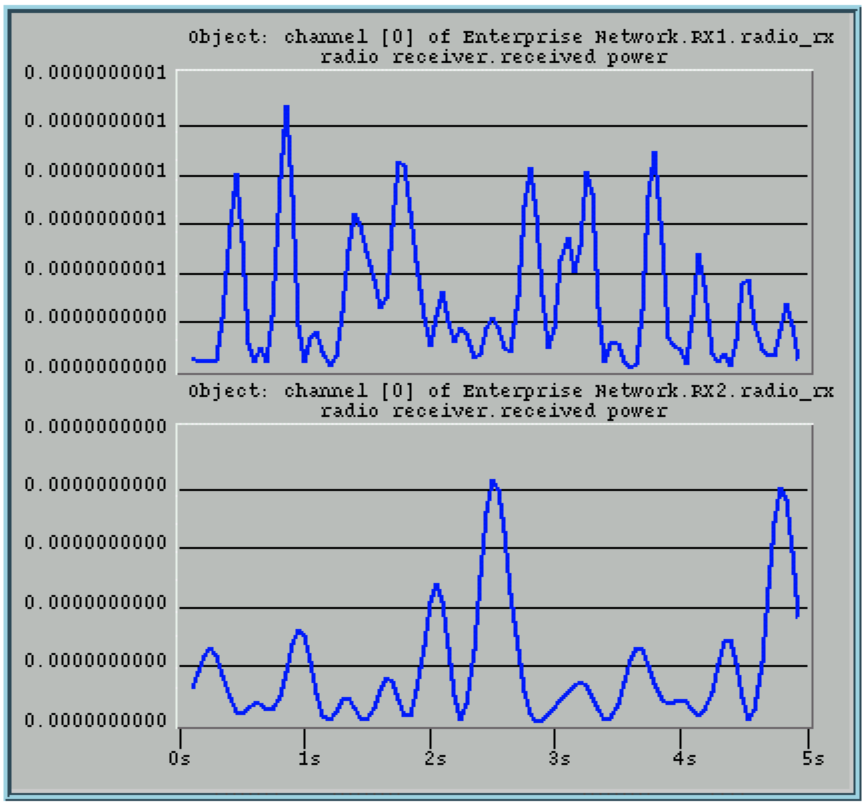

Figure 11: Ricean Fading Envelope