SSR/SHR Simulator User Manual v3.0.3

Mark Lisee

January 15, 2007

Contents

1 Overview

Same set of executable images simulate both protocols

Located in examples/sense

Use sim_shr_ack, sim_shr_loc_ack

Channel: radio model vs. adjacency matrix

Component diagram

2 Installing the Simulator

2.1 Supported Operating Systems

SENSE v3.0.3has been tested on

- Linux (Ubuntu "Breezy Badger", kernel 2.6.12) with gcc version 4.0.2.
- Mac OS X v10.4 "Tiger" with gcc version 4.0.1
- Windows XP with cygwin (version unknown)

SENSE requires that lex (flex on Windows XP) be installed. This should be the case if the gcc (or Cygwin) build environment is installed.

2.2 Building the Simulator

Follow these steps to build and test SENSE.

- Download the desired file from the download download directory. Note that there is a zip file specifically for Windows XP.
- Decompress the file. This will create the directory tree `sense-v3.0.3`.
- `cd sense-v3.0.3`
- `make`
- To do a test run of all of the examples:
  - `cd examples/sense`
  - `make test`
- To build the utilities that will display the state of the nodes in the SSR or SHR simulation
  - `cd analyzer/src`
  - `make`

2.3 Compile Flags

-DSHR_PRINT

xxx

-DVISUAL_ROUTE

Defining this flag enables two print statements.

- The first shows the transmission time, sequence number and maximum hop count for each packet generated by a source node.
- The second shows source and desination nodes, delivery time, actual hop count and list of nodes visited for each successfully delivered packet. The size of the list is controlled by VR_SIZE.

-DVR_SIZE=*n*

xxx

2.4 The Organization of the Make files

The make system is a combination of two different methodologies, type A and type B. (I know this is bad, but the `cxx` utility will not automatically generate dependencies.) Both types get compile flags and optimization flags from `definitions.mk` and `optimizations.mk` respectively. These files are located in the top directory.

Type A: The directories `analyzer` and `libraries` use the type A make system. The type A make system use several include files in addition to those mentioned above.

- `basicDefinitions.mk`: This file is located in the directory `code/misc`. It defines global definitions, such as the compile command, which may need to be changed.
- `basicRules.mk`: This file is located in the directory `code/misc`. It defines the rules to build the targets. It is unlikely that you will need to modify this file.
- `projectDefinitions.mk`: This file is located in the directory `code/`*`library`*`/misc`. It identifies the libraries that will be built, or optionally, used. It is unlikely that this file will require modification.

The `Makefile` contained in each `src` directory defines the targets (executables) and object files to be built by that directory. These are listed at the top of the file by programming language (C, C++ or Objective C) and exclude the filename extension. If a new source file needs to be added to the build list, simply add its name to the appropriate variable assignment statement. The source dependencies are automatically created.

Type B: The remaining directories use the type B make system. Each `Makefile` is self contained, except for `definitions.mk` and `optimizations.mk`. Adding a source file to the build list is more work than for type A and the dependency list must be manually maintained. Good luck if you want to build a library.

2.5 Lex and Yacc

The SSR/SHR simulator uses lex and yacc in two ways. First, to parse the configuration file that defines the topology matrix. The configuration file syntax is described in section 3.2. Second, SSR/SHR may be instructed to write a stats file which is then parsed by the analysis utilities, which are described here. While the other protocol simulators do not use a configuration file, the make system links them against the configuration file library.

3 Running the Simulator

3.1 Command Line Arguments

Several of the arguments take a Boolean value. These arguments check only the first letter of the value and the comparison is case insensitive.

-a *float*
Default value: 1.0

The mean percent of time that each node is active. The value of this argument times the value of -A is the mean time that a node is active before going to sleep. A new active time is calculated for each cycle. Source and destination nodes are always active. The default value is 1.0, or 100%.

-A *float*
Default value: 100.0 seconds

The mean active cycle time in seconds. This is the sum of the node's active and sleeping times. See the description of -a.

-b *ssr* | *shr*
Default value: ssr

The backoff type to use when forwarding a packet. The permitted backoff types are *ssr* and *shr* and are case insensitive. To simulate the original SSR protocol, use the arguments "-b ssr -C true". To simulate the SHR protocol, use the arguments "-b shr -C false". See the publications for a full description of the protocols.

-c *filename*
Default value: *none*

The name of the configuration file. This argument applies only if the simulation has been built with the topology matrix channel (i.e. sim_shr_loc_ack). The contents and syntax of the configuration files are described in section 3.2.

-C *Boolean*
Default value: true

This argument determines if the back off delay is continuous (*true*) or slotted (*false*). The width of the slot is controlled by the argument -w. Proper implementation of the SSR protocol requires that the back off delay is continuous. Proper implementation of the SHR protocol requires that the back off delay is slotted. The simulator will allow the user to simulate an improper implementation.

-d *filename*
Default value: *none*

This argument is optional. It is the name of the statistics file that will be written at the end of the simulation. The analysis utilities use this file as input. See section 4.2 for a description of these utilities.

-e *float*
Default value: 1000.0 seconds

The simulation end time in seconds.

-i *float*
Default value: 20.0 seconds

The mean time in seconds between packet transmissions by the source nodes. Each source node will have its own packet transmission interval.

-l *float*
Default value: 0.100 seconds

The value of λ in seconds. This value specifies the upper limit on a node's back off delay. See the SSR and SHR protocol publications for a full description of λ.

-n *int*
Default value: 110

The number of nodes in the simulation.

-p *float*
Default value: 0.028 watts

The antenna transmission power in watts.

-r *int*
Default value: *current time*

The random number seed.

-R *int*
Default value: 0

This argument controls the behavior of SHR's route repair algorithm. The route repair algorithm adjusts this node's expected hop to the destination to be one more than the expected hop count of the responder. If this value is 0, then SHR will not implement route repair. Other positive values indicate the number of packets required before this node changes its expected hop count. Values less than 0 are not permitted. This argument has no effect if the back off type is set to *ssr*. See the SHR protocol publication for a full description of the route repair algorithm.

-s *int*
Default value: 11

The number of source nodes in the simulation. If the many-to-many traffic model is chosen (see the flag -S), then an equal number of destinations nodes are chosen. If the many-to-one traffic model is chosen, then one node is chosen as the destination. If the traffic is bidirectional (see the flag -u), then the destination node(s) will also transmit packets.

-S *Boolean*
Default value: false

This argument determines if the traffic is many-to-one (*true*) or many-to-many (*false*). Many-to-one communcations represents many nodes communicating with a single sink.

-t *float*
Default value: 0.0 seconds

The transition timer, which represents the amount of time it takes the hardware to transition from sensing the carrier to actually transmitting data. Typically, this value is set to 20 μsec.

-u *Boolean*
Default value: false

A Boolean flag that determines if the traffic is unidirectional (*true*) or bidirectional (*false*).

-v *filename*
Default value: *none*

This argument, if supplied, enables creation of the visualizer files. The value of the argument is the base name of the created files. See section 4.1 for a description of the visualizer utility.

-w *float*
Default value: 0.001 seconds

This argument is used in conjunction with "-C false". It specifies the slot width in seconds. This should be longer than the transmission time of the longest DATA packet. By default, DATA packets have [500,1500] data bytes plus headers, which fits in a 12 msec slot.

-x *float*
Default value: 2000.0 meters

The x and y dimensions in meters of the physical area covered by the simulation. The same value is used for both dimensions.

-z *float*
Default value: 0.0 seconds

The SSR/SHR simulator has the option to clear its statistics once during the simulation. This argument specifies that time in seconds. A value of 0.0 indicates that the statistics will not be reset.

3.2 Configuration File Syntax

xxx

3.3 Simulation Output

xxx

3.4 Examples

xxx

4 Utilities

xxx

4.1 Visualizer

xxx

4.2 Simulation Analysis

xxx

4.3 Scripts

xxx

5 Limitations

xxx

6 Publications

xxx