vanet-routing-compare.cc

User's Guide

1. Motivation and Background

The *ns-3* simulator provides a rich set of modules which support various networking simulation scenarios. Effective simulation of a vehicular ad hoc network (VANET) requires models that capture the highly dynamic wireless, propagation and mobility characteristics.

We describe here an *ns-3* script, *vanet-routing-compare.cc*, that uniquely combines several *ns-3* modules and examples to assist the VANET simulation developer. The work is described in [1] and the script source code and supporting modifications to *ns-3.19* are provided at the following:

```
https://github.com/scarpenncsu/ns-3.19a.git
```

Most notably, we draw from the following *ns-3* examples:

[R.1] *./examples/routing/manet-routing.compare.cc* :
Runs DSDV, AODV, OLSR, or DSR routing protocols in a typical random waypoint model using 802.11b.

[R.2] */src/propagation/model/itu-r-1411-los-propagation-loss-model.cc* :
Models Recommendation ITU P.1441 propagation loss model, suitable for addressing propagation methods for short-range outdoor systems in the 300 MHz to 100 GHz, making it a suitable contender for 802.111p VANET simulations.

[R.3] *./src/mobility/examples/ns2-mobility-trace.cc* :
Loads ns2 mobility trace files (which may, for example, be generated from SUMO)

[R.4] *./src/wave/examples/wave-simple-80211p.cc* :
Models 802.11p/WAVE message transmission and receipt between two nodes.

Individually, each example contributes some capability, but none provides everything desired in a VANET simulation. For example, [R.1] simulates routing protocols, but does so in a MANET environment using a Random Waypoint (RWP) model and 802.11b Wi-Fi settings. Furthermore, the scenario is absent assumed Basic Safety Messages (BSMs) and utilizes the Friis propagation loss model.

We contend that more-realistic VANET routing protocol simulation can be achieved with the following: 1) realistic vehicular mobility traces; 2) a propagation loss model suitable to VANET conditions (e.g.. ITU R-1411); 3) background data (i.e. safety messages) contending for the channel along with routing data; and 4) routing protocols better-suited for VANET (e.g. GPSR). Thus, the *ns-3* script, *vanet-routing-compare.cc*, combines the modules and examples above, allowing parameter customization in order to model various VANET scenarios.

2. Parameters

The following parameters may be used when running a VANET routing simulation:

2.1 General Simulation Parameters

Simulation				
Parameter	**Meaning**	**Values**	**Default**	**Example**
totaltime	Simulation end time	<double>	300.01	./waf --run "scratch/vanet-routing-compare --totaltime=100"
verbose	Output	0=quiet; 1=verbose	0	./waf --run "scratch/vanet-routing-compare --verbose=1"
flowmon	Output flowmon statistics	0=off; 1=on	0	./waf --run "scratch/vanet-routing-compare --flowmon=1"
routing_tables	Dump routing tables	0=no; 1=yes	0	./waf --run "scratch/vanet-routing-compare --routing_tables=1"
ascii_trace	Dump ASCII trace file	0=no; 1=yes	0	./waf --run "scratch/vanet-routing-compare --ascii_trace=1"
pcap	Create PCAP files for all nodes	0=no; 1=yes	0	./waf --run "scratch/vanet-routing-compare --pcap=1"
CSVfileName	The name of the CSV output file name	<string>	vanet-routing.output.csv	./waf --run "scratch/vanet-routing-compare --CSVfileName=AODV.csv"
scenario	Predefined scenarios	1=Realistic,Zurich; 2=51 nodes, RWP, 100s; 3=101 nodes, RWP, 10s	1	./waf --run "scratch/vanet-routing-compare --scenario=2"

2.2 Mobility Parameters

Mobility				
Parameter	**Meaning**	**Values**	**Default**	**Example**
mobility	Mobility model	1=trace; 2=RWP		./waf --run "scratch/vanet-routing-compare --mobility=2"
nodes	Number of nodes (i.e. vehicles)	<int>	156	./waf --run "scratch/vanet-routing-compare --nodes=51"
traceMobility	Enable mobility tracing	<boolean>	FALSE	./waf --run "scratch/vanet-routing-compare --traceMobility=1"
traceFile	Ns2 movement trace file (Trace playback only, i.e. mobility=1)	ns-2 mobility trace file	./scratch/ct-unterstrass-1day.filt.5.adj.mov	./waf --run "scratch/vanet-routing-compare --traceFile=trace.out"
logFile	Mobility log file	<string>	ct-unterstrass-1day.filt.5.adj.log	./waf --run "scratch/vanet-routing-compare --logFile=mobility.log"
speed	Node speed (m/s) (RWP only, i.e. mobility=2)	<int>	20	./waf --run "scratch/vanet-routing-compare --speed=15"
pause	Node pause (s) (RWP only, i.e. mobility=2)	<int>	0	./waf --run "scratch/vanet-routing-compare --pause=3"

2.3 MAC/PHY Characteristics

MAC/PHY Characteristics				
Parameter	**Meaning**	**Values**	**Default**	**Example**
txp	Transmit power (dB)	<double>	7.5	./waf --run "scratch/vanet-routing-compare --txp=20"
txdist	Expected BSM transmission range, m	<double>	145	./waf --run "scratch/vanet-routing-compare --txdist=210"
lossModel	Propagation loss model	1=Friis; 2=ItuR1411Los; 3=TwoRayGround; 4=LogDistance	2 (ITU R-1411)	./waf --run "scratch/vanet-routing-compare --lossModel=3"
80211Mode	802.11 mode (i.e. b/p)	1=802.11p; 2=802.11b		./waf --run "scratch/vanet-routing-compare --80211Mode=2"
phyMode	Wifi Phy mode	802.11p PHY mode settings	OfdmRate6MbpsBW10MHz	
phyMode_b	Phy mode 802.11b	<see default>	DsssRate11Mbps	
gpsaccuracy	GPS time sync accuracy in ns	<int>	10000	./waf --run "scratch/vanet-routing-compare --gpsaccuracy=100"

2.4 Application Traffic Parameters

Application Traffic				
Parameter	**Meaning**	**Values**	**Default**	**Example**
bsm	(WAVE) BSM size (bytes)	<int>	200	./waf --run "scratch/vanet-routing-compare --bsm=175"
interval	(WAVE) BSM interval (s)	<double>	0.1	./waf --run "scratch/vanet-routing-compare --interval=1.0"
protocol	Routing protocol	1=OLSR; 2=AODV; 3=DSDV; 4=DSR	2 (AODV)	./waf --run "scratch/vanet-routing-compare --protocol=4"
rate	Data rate for routing data	<see default>	2048bps	./waf --run "scratch/vanet-routing-compare --rate=4048bps"
sinks	Number of routing sinks	<int>	10	./waf --run "scratch/vanet-routing-compare --sinks=25"

3. Scenarios

The following predefined scenarios may be used to set up certain parameters prior to a simulation run.

[Scenario 1] Realistic, Zurich
--scenario=1
1=Realistic,Zurich

Scenario 1 plays back a realistic vehicular trace file of 156 vehicles over 300 seconds in a 4.6 km x 3.0 km suburban section of Zurich, Switzerland. One vehicle (#14) serves as the WAVE "sink" node (receiving WAVE messages) while the other 155 vehicles transmit 200-byte WAVE BSMs 10 times per second (i.e. total WAVE transmit rate = 155 x 10 = 1550 pps).

Default parameter modifications:

```
m_traceFile = "./scratch/ct-unterstrass-1day.filt.5.adj.mov";
m_logFile = "ct-unterstrass-1day.filt.5.adj.log";
m_mobility = 1;
m_nNodes = 156;
m_TotalTime = 300.01;
m_nodeSpeed = 0;
m_nodePause = 0;
```

[Scenario 2] 51 nodes using Random Waypoint for 100s
--scenario=2
2=51 nodes, RWP, 100s

Scenario 2 models 51 nodes using a Random Waypoint model in a 300 m x 1500 m synthetic 4-lane highway. One node serves as the WAVE "sink" node (receiving WAVE messages) while the other 50 nodes transmit 200-byte WAVE BSMs 10 times per second (i.e. total WAVE transmit rate = 50 x 10 = 500 WAVE pps).

Default parameter modifications:

```
m_traceFile = "";
m_logFile = "";
m_mobility = 2;
m_nNodes = 51;
m_TotalTime = 100.0;
m_nodeSpeed = 20;
m_nodePause = 0;
```

[Scenario 3] 101 nodes using Random Waypoint for 10s
--scenario=3
3=101 nodes, RWP, 10s

Scenario 2 models 101 nodes using a Random Waypoint model in a 300 m x 1500 m synthetic 4-lane highway. One node serves as the WAVE "sink" node (receiving WAVE messages) while the other 100 nodes transmit 200-byte WAVE BSMs 10 times per second (i.e. total WAVE transmit rate = 100 x 10 = 1000 pps).

Default parameter modifications:

```
m_traceFile = "";
m_logFile = "";
m_mobility = 2;
m_nNodes = 101;
m_TotalTime = 10.0;
m_nodeSpeed = 1;
m_nodePause = 0;
```

References

[1] SE Carpenter, ML Sichitiu, DA Underwood, M. Patwardhan, S Starr. Evaluating VANET Performance in ns-3. In Workshop on *ns-3* (WNS3), 2014.