Comparing a Real-Life WSN Platform Small Network and its OPNET Modeler model using Hypothesis Testing

Gilbert E. PEREZ
Government Communications Systems Division, Harris Corporation
Melbourne, FL, USA gperez07@harris.com

and

Ivica KOSTANIC
Electrical and Computer Engineering Department, Florida Institute of Technology
Melbourne, FL, USA kostanic@fit.edu

ABSTRACT

To avoid the high cost and arduous effort usually associated with field analysis of Wireless Sensor Network (WSN), Modeling and Simulation (M&S) is used to predict the behavior and performance of the network. However, the simulation models utilized to imitate real life networks are often used for general purpose. Therefore, they are less likely to provide accurate predictions for different real life networks. In this paper, a comparison methodology based on hypothesis testing is proposed to evaluate and compare simulation output versus real-life network measurements. Performance related parameters such as traffic generation rates and goodput rates for a small WSN are considered. To execute the comparison methodology, a "Comparison Tool", composed of MATLAB scripts is developed and used. The comparison tool demonstrates the need for model verification and the analysis of good agreements between the simulation and empirical measurements.

Keywords: Wireless Sensor Network; Simulation; Modeling

1. INTRODUCTION

Modeling and Simulation (M&S) tools provide a comprehensive environment supporting the modeling and simulation of communication networks and distributed systems. Such tools include Riverbed's OPNET Modeler [1], QualNet [2], NS-2 [3] and emerging NS-3 [4]. These tools offer the capability to study the behavior and performance of the networks through model design, simulation, data collection and analysis. Hence, simulation models are needed to adequately represent the network or system being modeled. Although any one of the tools could have been used, the commercially available OPNET Modeler was chosen to have some benefits as described in [5]-[7] and available to universities via Riverbed's OPNET University Programs [8]. The development of the model is an iterative process involving subject matter experts (SME). The goal of the process is to develop a model that represents the particular network components and their interactions. It behooves the modeler and the model user to frequently ask the imperative question: "How representative is my network model to the actual real-life network being modeled?"

Assuming the model is a good enough representation of the actual network, model simulations are performed as trade-off studies to evaluate performance evaluations. This assumption needs to be validated to put any trust in the outputs of the model and the performance evaluations. Hence, verification and validation (V&V) of the simulation model is necessitated, that is verification that the model was indeed coded correctly and validation that it produces valid results. As stated in [9], one of the key V&V principles stated in the Verification, Validation, and Accreditation Recommended Practices Guide (DMSO, 1996) is that "V&V is both an art and a science, requiring creativity and insight." Current simulation packages, offer a process for analyzing and designing communication networks, devices, protocols, and applications. System behaviors can be compared in regards to different technologies but not to real-life implementations of the network. In [10], a method is shown to establish statistical validity of discrete event simulations through running multi-seeded simulation runs, however, this is model statistical validity and a comparison to a real-life platform is still needed.

In order to avoid the high cost to run test and "what-if" scenarios on real-life network systems, model simulations of the real-life networks are executed. Subject Matter Experts (SME) are the primary decision makers in determining whether the model output is adequately representative of the expected output. With these simulation results being used to make system design and implementation decisions, it is imperative to ensure the simulation model itself is an adequate representation of the real-life network. This paper illustrates an automated methodology for comparison of the output results of a Small real-life WSN and its complimentary OPNET WSN model.

Organization of the paper

Section 2 contains an overview of the two environments used for evaluation of the output of simulation against real systems. In Section 3, the experiment scenario performed on the real system environment is explained and its results shown. Section 4 contains an overview of the simulation environment setup to model the scenario executed on the real system. In Section 5, the comparison methodology is described and Section 6 encompasses the results of the comparison methodology as produced by the "Comparison Tool". Section 7 contains the conclusions and the proposed follow-on work.

2. EXPERIMENTAL SETUP OVERVIEW

The experimental setup consists of three distinct yet connected realms: a Real-Life WSN Wireless Sensor Network Platform, a WSN OPNET Modeler Model and a comparison tool. The three areas are described and their relationships established as follows.

Real-Life WSN Wireless Sensor Network Platform
The Crossbow Starter Kit [11]-[17], provided by the Wireless Center of Excellence at FIT [18], contains all the components needed to deploy a basic wireless sensor network and is used to generate the real world data. The kit includes three sensor devices known as motes or nodes, and a server gateway to connect sensors to a management system. A Windows® Based User Interface (Development Environment) for remote analysis, monitoring and mote programming, is also included. The battery powered mote, the USB interfaced programming board (which can act as a sniffer or a base station) and the laptop running the development environment are shown in Fig. 1.

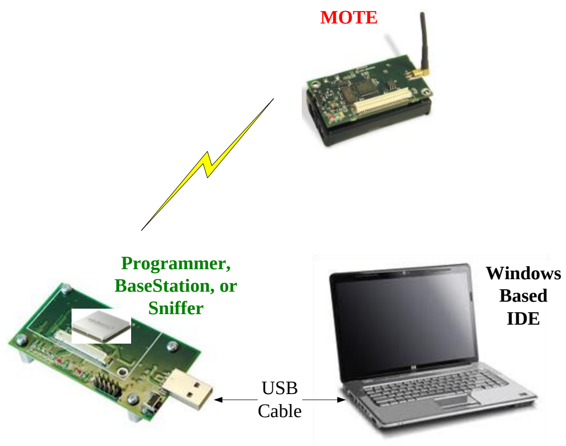


Fig. 1WSN Kit Components

Modeling and Simulation Tool
OPNET Modeler (Release 15.0.A PL3) provided by Riverbed under the OPNET University Programs [8], is the discrete event simulator used to simulate the WSN real-life platform. Although OPNET Modeler is capable of simulating all 7 layers of the Open Systems Interconnection (OSI) model, of particular interest are the application, network, MAC and physical layers. In general, the architecture of an OPNET model is structured in a hierarchical fashion from top to bottom into three domains: network, node and process. The network domain encompasses definition of the network model nodes and their interconnecting communication links. Geographical location of nodes and the communication mediums are also defined in the network domain. The node domain entails definition of how the developer implements the behavior of the specific node with respect to the 7 layers of the OSI and the data flow between the layers. Further definition of each OSI layer processing is captured in the process domain. Model and network statistics generation and implementation are part of the process domain.

WSN Model
The OPNET model of the motes, obtained from open-ZB web site [19] as an open source, is the starting point for simulating the real-life WSN platform.

Application Layer: The application layer of the OPNET Model provides the scenario builder with an option to control traffic generation patterns and attributes of the node and simulation.

Network Layer: The network layer of the OPNET model provides the scenario builder with an option to control the network layer attributes, hence defining the Zigbee network and its sizing for the node during the scenario. Packets from the application layer are encapsulated and propagated to the MAC layer. Packets from the MAC layer are de-encapsulated and propagated to the application layer.

3. THE EXPERIMENT

The experiment performed is described in this section. The rationale is to characterize the Small WSN Real-World Platform in order to model and form the basis of comparison.

Small Real-World WSN Platform

Fig. 2 is the small WSN network, consisting of a PAN Coordinator node and two Router nodes used to generate the "experimental data".

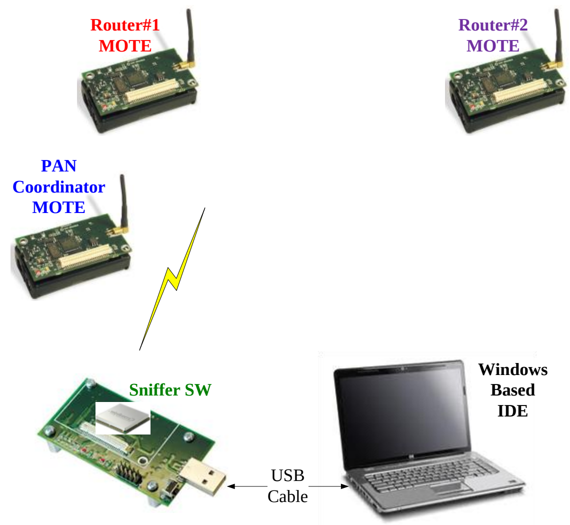


Fig. 2 Small Real-Life WSN Network Setup

The network scenario timeline is shown in Fig. 3. The Pan Coordinator runs for 2 mins (blue in the Network Startup Scenario Timeline) trying to setup the network. Router#1 starts to attempt to join the network at 2 mins into the scenario (red in the Network Startup Scenario Timeline). Router#2 joins the network at minute 7 of the scenario and runs for 10 mins (purple in the Network Scenario Timeline) Simultaneously the sniffer is running and capturing all traffic from the routers and coordinator for 17 mins (green in the Network Startup Scenario Timeline). This experiment is executed; saving the sniffer captured experimental data.

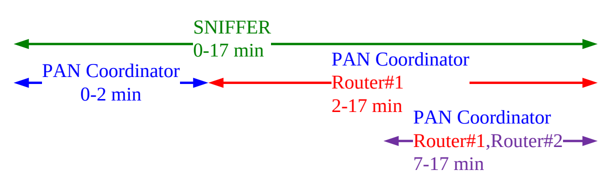


Fig. 3 Network Scenario Timeline

The Experiment Results
The Sniffer data is processed by the MATLAB scripts (part of the "Comparison Tool") yielding the traffic and its rates as shown in Table 1. The traffic rate statistics (transmission rate average and standard deviation) are used to generate the model traffic and used for comparison for model validation. Analysis of the Sniffer data yielded the WSN PAN Coordinator mote transmitting routing (Rte) messages approximately every 34

seconds. The WSN Router#1 mote joins the network 2 minutes into the scenario and transmits its routing (Rte) messages approximately every 37 seconds. Router#1 also transmits is sensor information (DatUp) messages approximately every 4.3 seconds. In addition, the Router#1 transmits its health information (Hlth) messages approximately every 2 minutes. In the scenario at minute 7 Router#2 joins the network via Router#1 and Router#1 relays the sensor information (DatUp) messages from Router#2 to the PAN Coordinator approximately every 4.6 seconds.

TABLE 1 SMALL REAL-WORLD WSN PLATFORM TRAFFIC

PAN Coordinator **(NodeID 0)** Traffic 0-17 min			
Traffic Data Type	Message Size (Bytes)	Ave Duration between Transmissions Ts (ms)	Std Dev (Ts) Ts (ms)
Routing (Rte)	12/15	34223.750	436.026
Router#1 **(NodeID 86)** Traffic 2-17 min Router#2 **(NodeID 87)** Traffic 2-17 min			
Traffic Data Type	Message Size (Bytes)	Ave Duration between Transmissions Ts (ms)	Std Dev (Ts) Ts (ms)
Router#1 Sensor (DatUp)	55	4305.580	285.125
Mote Health (Hlth)	29	119989.500	306.404
Routing (Rte)	12	37333.417	41.371
Router#2 Sensor (DatUp)	55	4558.279	1193.936

4. THE SIMULATION

The simulation performed is described in this section. The rationale is to model and simulate the Small WSN Real-World Platform and execute the "network scenario" in order to model and form a basis of comparison.

OPNET Model: Small WSN
The OPNET Model acquired from [19] was modified to generate the PAN Coordinator and Router traffic patterns as analyzed and captured in Table 1. The model statistic collection mode was set to collect all values mode, as opposed to sample mode, bucket mode or glitch mode. The small WSN model is executed long enough in duration to obtain at least 30 samples of the periodic traffic patterns. The simulation scenario was setup to mimic the Network Scenario Timeline in Fig. 3. The scenario runtime duration was selected to ensure that at least n > 30 samples are collected to deal with the large sample case in the test of means as described in [20]. A snapshot of the scenario data traffic settings for the model scenario is shown in Fig. 4.

PAN Coordinator Routing (Rte)	
Best Effort (CAP)	(...)
Start Time (sec)	0.1
Stop Time (sec)	end of simulation
Packet Interarrival Time (sec)	normal (34.22375, 0.190119083)
Packet Size (bit)	constant (12)
Acknowledgment	disabled
Router #1Sensor (DatUp)	
Be1st Effort (CAP)	(...)
Start Time (sec)	120
Stop Time (sec)	end of simulation
Packet Interarrival Time (sec)	normal (4.305580153, 0.081296138)
Packet Size (bit)	constant (55)
Acknowledgment	disabled
Router #1Mote Health (Rte)	
Be2st Effort (CAP)	(...)
Start Time (sec)	120
Stop Time (sec)	end of simulation
Packet Interarrival Time (sec)	normal (37.33341667, 0.001711538)
Packet Size (bit)	constant (12)
Acknowledgment	disabled
Router#1 Sensor (Hlth)	
Best Effort (CAP)	(...)
Start Time (sec)	120
Stop Time (sec)	end of simulation
Packet Interarrival Time (sec)	normal (119.9895, 0.0938835)
Packet Size (bit)	constant (29)
Acknowledgment	enabled
Router #2Sensor (DatUp)	
Be1st Effort (CAP)	(...)
Start Time (sec)	420
Stop Time (sec)	end of simulation
Packet Interarrival Time (sec)	normal (4.278947826, 0.004863997)
Packet Size (bit)	constant (55)
Acknowledgment	disabled

Fig. 4 Scenario data traffic settings

The Simulation Results
The OPNET Modeler data is processed by the MATLAB scripts as part of the "Comparison Tool" using the comparison methodology as described in the following.

5. THE COMPARISON METHODOLOGY

The comparison methodology is composed of the steps outlined in Table 2. In this paper, the process is illustrated for the traffic generation of the small WSN. The data generated by the Real-Life WSN Platform and the OPNET Model are the input to the "Comparison Tool" as shown in Fig. 5. The comparison tool is developed to implement the methodology allowing for repeatability and ease of use as discussed in [21]. The comparison methodology process is implemented in MATLAB [22] scripts.

TABLE 2 COMPARISON METHODOLOGY PROCESS

Comparison Methodology Process	
Step#	Description
1.	Instrument the Real-Life WSN Platform to generate the “experimental” data.
2.	Model the WSN to generate the “simulated” data.
3.	Run the Hypothesis Test Procedure of section 8.1 of [20] on the following: a) Two-Sample Test of Means (section 8.2.2) b) Two-Sample Test of Variance (section 8.3.2)
4.	Calculate and compare the Goodput *(ratio of the delivered amount of information in bits to the total delivery time)* for the Links Pan Coordinator ➔ Router#1 Router#1 ➔ Pan Coordinator Router#2 ➔ Router#1

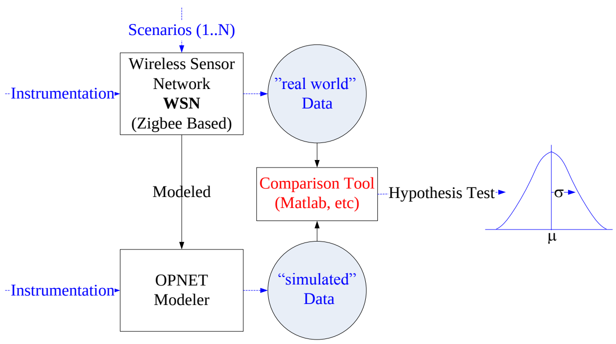


Fig. 5 Comparison Methodology Diagram

The “Comparator Tool” MATLAB scripts read the files containing the ’experimental’ and ‘simulated’ data and executes the (Two-Sample Test of Means Algorithm) TABLE 3and (Two-Sample Test of Variance Algorithm) TABLE 4 Hypothesis tests. In addition, the “Comparator Tool” MATLAB scripts calculate the application level throughput (i.e. goodput) for the Pan Coordinator Mote and the Router Motes. The goodput is the ratio of the delivered amount of information in bits to the total delivery time in seconds as shown in Equation (1).

$$\text{Goodput} = \frac{\text{number of useful information bits}}{\text{total delivery time in seconds}} \quad \text{(bps)} \tag{1}$$

Furthermore, the goodput for the experimental and the simulation data (for the [Pan Coordinator ➔ Router#1], [Router#1 ➔ Pan Coordinator] & [Router#2➔Router#2] Links) are compared, yielding a percentage delta. The percent delta is calculated as shown in (2) and anticipated to be extremely low, if the simulation data does indeed compare well with the experimental data.

$$\text{Delta (\%)} = \frac{|\text{Exp\_Goodput(bps)} - \text{Sim\_Goodput(bps)}|}{\text{Exp\_Goodput(bps)}} \times 100\% \tag{2}$$

The comparison results are obtained graphically and shown in the following section.

TABLE 3 TWO-SAMPLE TEST OF MEANS ALGORITHM

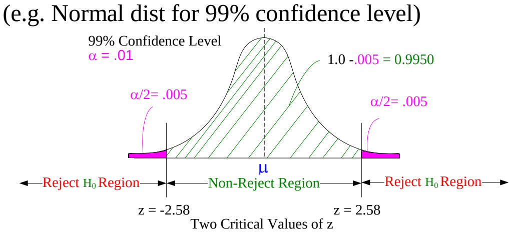
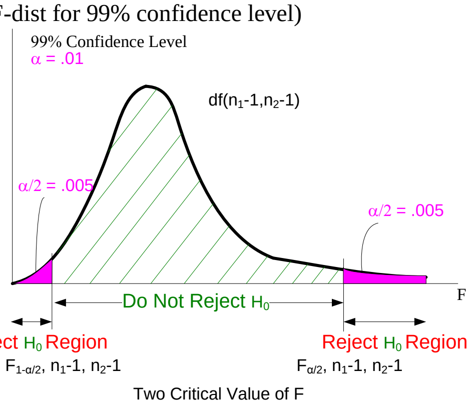
Two-Sample Test of Means	
Step #	Description
1.	State the Null H_0 & Alternative H_1 Hypothesis $H_0: (\mu_1 - \mu_2) = 0$ (The mean of experiment and simulated data are NOT different) $H_1: (\mu_1 - \mu_2) \neq 0$ (The mean of experiment and simulated data are different)
2.	Select the distribution to use: Since the sample sizes are ($n_1 \geq 30$), ($n_2 \geq 30$), both the samples are large. So, the sampling distribution of $\bar{x}_1 - \bar{x}_2$ is (approx or exactly) normal & we use the normal distrib to make the hypothesis test.
3.	Determine the rejection and non-rejection regions (from the Confidence Level, yielding the significance level α) The ≠ sign in the alternative hypothesis H_1, indicates that the test is Two –Tailed. The area in each tail of the normal distribution curve is α/2 Calculate the critical values of z for α/2 (area in each tail). (e.g. Normal dist for 99% confidence level) 
4.	Compute the value of the test statistic z for $\bar{x}_1 - \bar{x}_2$: $z = \frac{(\bar{x}_1 - \bar{x}_2) - \overbrace{(\mu_1 - \mu_2)}^{0 \text{ from } H_o}}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$
5.	Plot the test statistic and make a decision to reject or not reject the null hypothesis H_0

TABLE 4 TWO-SAMPLE TEST OF VARIANCE ALGORITHM

<table>
<tr><th colspan="2">Two-Sample Test of Variance</th></tr>
<tr><th>Step#</th><th>Description</th></tr>
<tr><td>1.</td><td>State the Null H_0 & Alternative H_1 Hypothesis
$H_0: (\sigma_1^2 - \sigma_2^2) = 0$
(The variance of experiment and simulated data are NOT different)

$H_1: (\sigma_1^2 - \sigma_2^2) \neq 0$
(The variance of experiment and simulated data are different)</td></tr>
<tr><td>2.</td><td>Select the distribution to use:
Since the sample sizes are ($n_1 \geq 30$), ($n_2 \geq 30$), both the samples are large. The sampling distribution of $(\sigma_1^2 - \sigma_2^2)$ is (approx or exactly) normal & we use the F distribution with $\nu_1 = n_1$-1 and $\nu_2 = n_2$-1 degrees of freedom to make the hypothesis test.</td></tr>
<tr><td>3.</td><td>Determine the rejection and non-rejection regions (from the Confidence Level, yielding the significance level α)
The $\neq$ sign in the alternative hypothesis H_1, indicates that the test is Two –Tailed.
The area in each tail of the F distribution with $\nu_1=n_1$-1 and $\nu_2=n_2$-1 degrees of freedom curve is $\alpha/2$
Calculate the critical value F for $\alpha/2$ (area in each tail).
(e.g. F-dist for 99% confidence level)

</td></tr>
<tr><td>4.</td><td>Compute the value of the test statistic F
(ratio of the sample variances):
$$F = \frac{S_1^2}{S_2^2}$$</td></tr>
<tr><td>5.</td><td>Plot the test statistic and make a decision to reject or not reject the null hypothesis H_0</td></tr>
</table>

6. THE COMPARISON METHODOLOGY RESULTS

What follows are the results from the "Comparison Tool" as described in the previous section.

Model Link Goodput Rates

Fig. 6 is the "Comparison Tool" graphical output result of comparing the Goodput Links:

- Pan Coordinator ➔ Router#1
- Router#1 ➔ Pan Coordinator
- Router#2 ➔ Router#1

As can be seen, the Goodput rates for the Pan Coordinator only differs by 0.33% leading to an inference that the simulation data (Model links) goodput match the experimental data (Real-Life Platform Link). It can be seen that this is not the case for the Router goodputs and the simulation model needs further modification to match the Real-Life Platform.

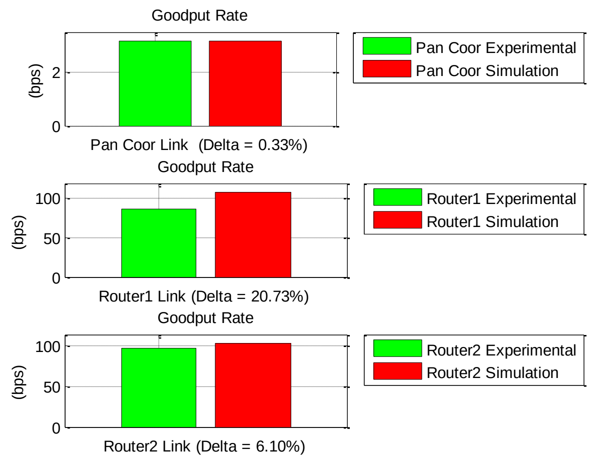
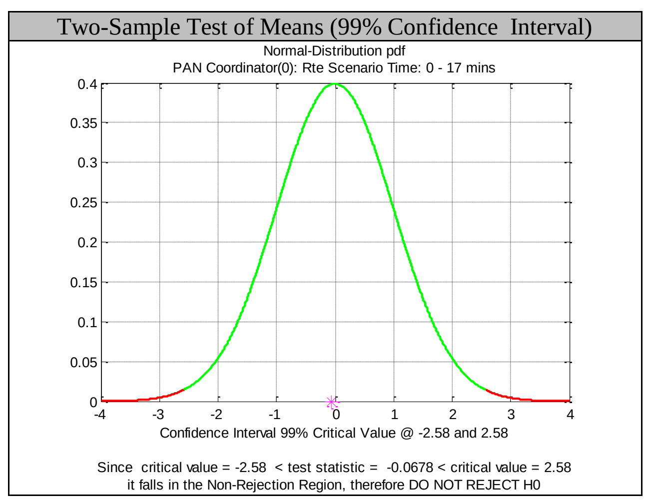


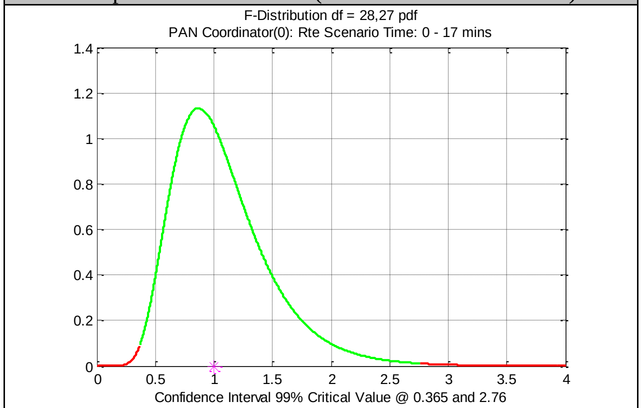
Fig. 6 Link Rate Goodput Comparison Result

Model Traffic Rates

Fig. 7 is the "Comparison Tool" graphical output result of comparing the Small Real-World WSN Platform Traffic in Table 1, using the aforementioned Hypothesis test. The "reject" region is annotated in red, the "Fail to reject" region is annotated in green and the test statistic is annotated with a magenta asterisk. As can be seen from the plots (Hypothesis Test results), the experimental data and the simulated data appear to agree with a 99% Confidence Level got all the traffic rates. The "Comparison Tool" generates the graphs in a matter of seconds, providing the user the ability to see results quickly and confirm the reproducibility of real-life results in simulation as analyzed in [23],[24].

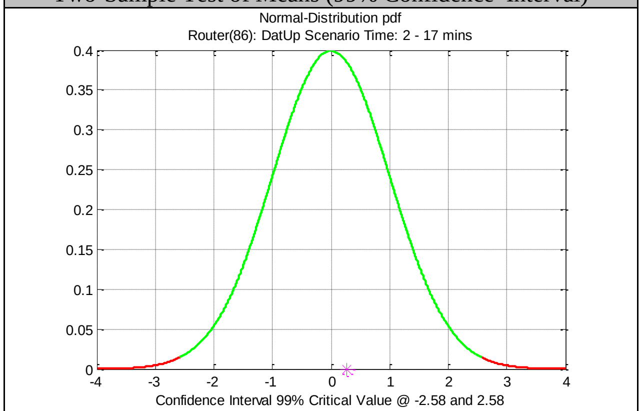


Two-Sample Test of Variance (99% Confidence Interval)



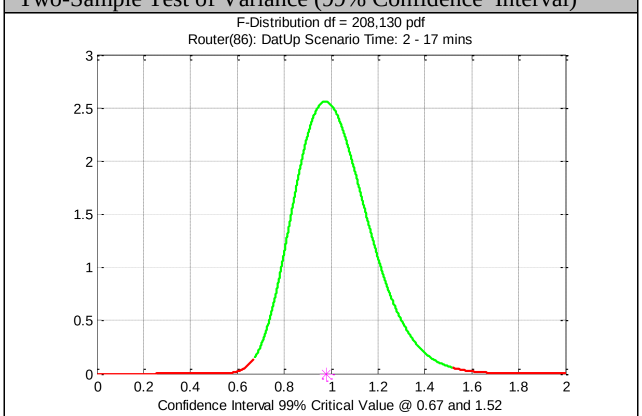
Since critical value = 0.365 < test statistic = 1 < critical value = 2.76
it falls in the Non-Rejection Region, therefore DO NOT REJECT H0

Two-Sample Test of Means (99% Confidence Interval)



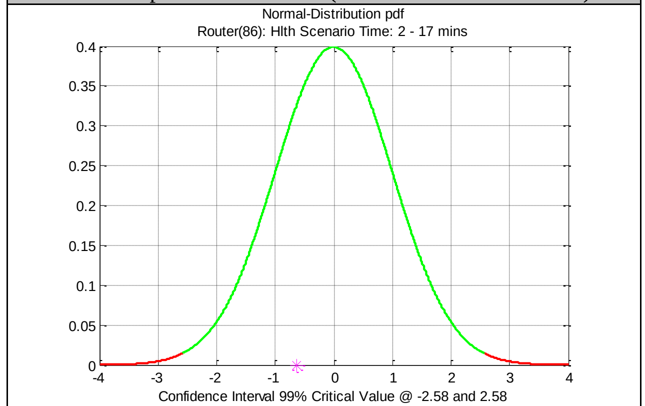
Since critical value = -2.58 < test statistic = 0.264 < critical value = 2.58
it falls in the Non-Rejection Region, therefore DO NOT REJECT H0

Two-Sample Test of Variance (99% Confidence Interval)



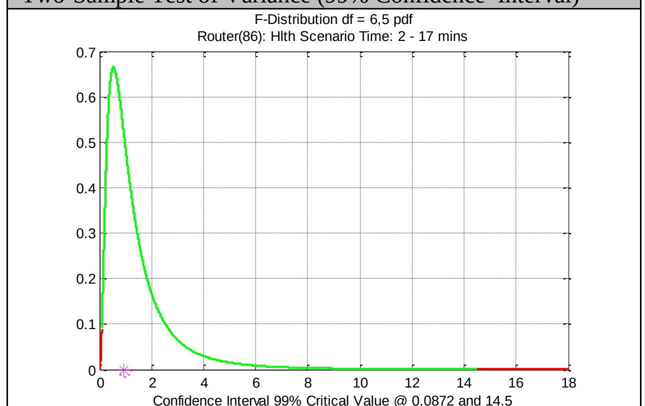
Since critical value = 0.67 < test statistic = 0.977 < critical value = 1.52
it falls in the Non-Rejection Region, therefore DO NOT REJECT H0

Two-Sample Test of Means (99% Confidence Interval)



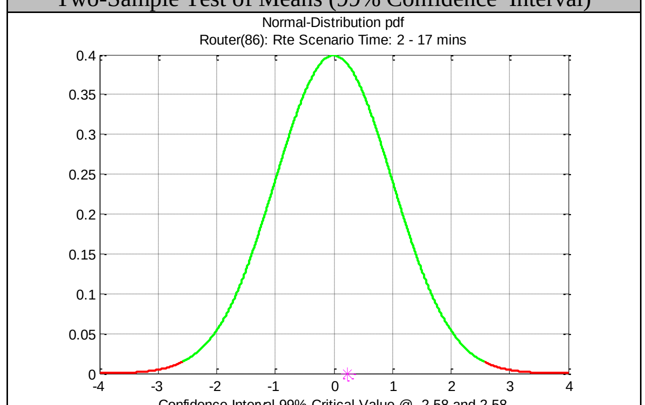
Since critical value = -2.58 < test statistic = -0.629 < critical value = 2.58
it falls in the Non-Rejection Region, therefore DO NOT REJECT H0

Two-Sample Test of Variance (99% Confidence Interval)



Since critical value = 0.0872 < test statistic = 0.917 < critical value = 14.5
it falls in the Non-Rejection Region, therefore DO NOT REJECT H0

Two-Sample Test of Means (99% Confidence Interval)



Since critical value = -2.58 < test statistic = 0.232 < critical value = 2.58
it falls in the Non-Rejection Region, therefore DO NOT REJECT H0

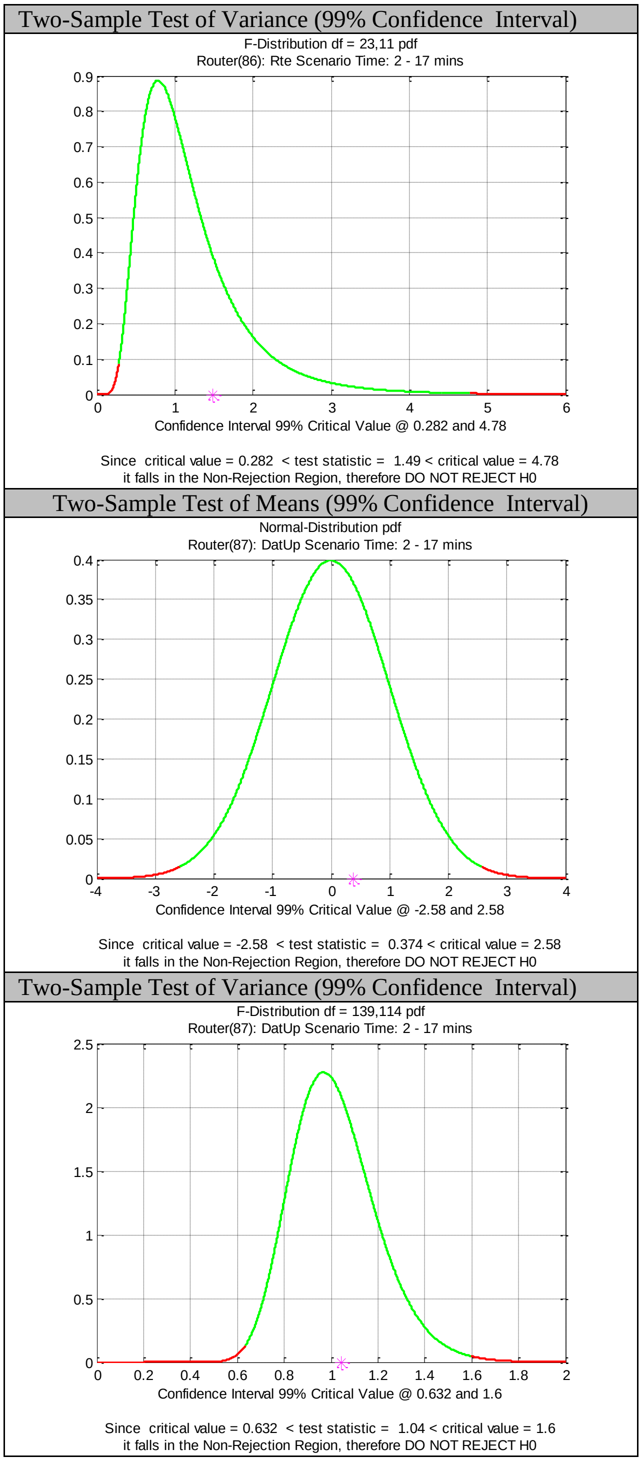


Fig. 7 Comparison Tool Output

7. CONCLUSIONS AND FOLLOW-ON WORK

In the paper, hypothesis testing was used to evaluate the mean and variance of the traffic generation rate of a small wireless sensor network consisting of a PAN Coordinator node and two Router nodes. The comparison was done between the 'real world' experimental data generated from an instrumented real-life WSN platform and an OPNET Modeler model that generated the 'simulated' data. The comparison methodology consisted of running hypothesis testing, implemented via MATLAB scripts. The scripts, integrated into the "Comparison Tool" allow for repeatable tests to be performed.

The results show that with a 99% confidence level or level of significance of 0.01, the test statistic fails in the "fail to reject" or "Non-rejection region" and the null hypothesis that the means and variance between the real-life node and the model are indeed equal. The desired inference that follows is that the model of the small wireless sensor network is representative of the real-life platform as far as the traffic generation rates and the links goodput are concerned. This is in part a continuation of the work presented in [25].

In follow-on work, the "Comparison Tool" will be enhanced to compare the pathloss reported by the simulation model and the real-life WSN node. Further work will be in the direction of networks having increased complexity of architecture and node interaction. Performance parameter(s) such as End to End (ETE) delay and Packet Loss will also be considered.

8. REFERENCES

[1] Riverbed OPNET Modeler URL: http://www.riverbed.com/products-solutions/products/network-planning-simulation/Network-Simulation.html

[2] Qualnet URL: http://web.scalable-networks.com/content/qualnet

[3] NS-2 URL: http://nsnam.isi.edu/nsnam/index.php/User_Information

[4] NS-3 URL: http://www.nsnam.org/

[5] Korkalainen, Marko, et al. "Survey of wireless sensor networks simulation tools for demanding applications." Networking and Services, 2009. ICNS`09. Fifth International Conference on. IEEE, 2009

[6] Timm-Giel, Andreas, et al. "Comparative simulations of WSN." ICT-MobileSummit (2008)

[7] Marghescu, C., et al. "Simulation of a wireless sensor network using OPNET."Design and Technology in Electronic Packaging (SIITME), 2011 IEEE 17th International Symposium for. IEEE, 2011

[8] OPNET University Program: http://www.opnet.com/university_program/research_with_opnet/

[9] Love, G. and Back, G. (2000) "Model Verification and Validation for Rapidly Developed Simulation Models: Balancing Cost and Theory", Proceedings of the 18th International Conference of the System Dynamics Society, Bergen, Norway, August 6-10.

[10] "Verifying Statistical Validity of Discrete Event Simulations", OPNET Technologies, Inc. URL: http://www.opnet.com , White_Paper_from_OPNET\Verifying_Statistical_Validity_DES.pdf

[11] Crossbow® http://bullseye.xbow.com:81/industry_solutions/BuildingAutomation.aspx

[12] Crossbow Wireless Sensor Network Start Kit: WSN_START_Datasheet.pdf

[13] Crossbow Technology, May 2009, http://www.xbow.com

[14] TinyOS, May 2009, http://www.tinyos.net

[15] MICA Wireless Measurement System Data Sheet: MICA_data_sheet.pdf

[16] MIB520- USB Gateway Data Sheet: URL: www.dinesgroup.org/projects/images/pdf_files/mib520_datasheet.pdf

[17] MoteWorks™ Software Platform: URL: www.willow.co.uk/MoteWorks_OEM_Edition.pdf

[18] Wireless Center of Excellence (FIT) URL: http://research.fit.edu/wice/

[19] IEEE 802.15.4 OPNET Simulation Model, http://www.open-zb.net/

[20] Allen, Arnold. Probability, Statistics, and Queueing Theory with Computer Science Applications. 2nd ed. Academic Press Inc., 1990

[21] G. E. Perez, W.A. Windham, "Automating Modeling & Simulation Pre/Post Processing," in Proceedings of OPNETWORK, 2011

[22] MATLAB and Statistics Toolbox Release 2012b, The MathWorks, Inc., Natick, Massachusetts, United States.

[23] B. Genge, C. Siaterlis, I. Nai Fovino, M. Masera: A Cyber-Physical Experimentation Environment for the Security Analysis of Networked Industrial Control Systems. Computers and Electrical Engineering, Elsevier, Volume 38, Issue 5, pp. 1146-1161, 2012

[24] B. Genge, C. Siaterlis: Analysis of the Effects of Distributed Denial-of-Service Attacks on MPLS Networks. International Journal of Critical Infrastructure Protection, Elsevier, Volume 6, Issue 2, pp. 87-95, 2013

[25] G. E. Perez, I. Kostanic, "OPNET Modeler model and Real-Life WSN Platform comparison: Validating the network model using Hypothesis Testing," in Proceedings of OPNETWORK, 2013