

Performance Analysis of the Augmented Wireless Sensor Network Testbed

Kenneth G. LeSueur
U.S. Army Redstone Technical Test Center
Subsystem Test and Analysis Branch
Redstone Arsenal, AL 35898
Ken.lesueur@us.army.mil

Dr. Emil Jovanov
Electrical and Computer Engineering Department
The University of Alabama in Huntsville
Huntsville, Alabama 35899
jovanov@ece.uah.edu

Abstract—This paper presents a system for a virtual wireless sensor network testing that supports real-time hardware in the loop testing of networked sensors. The testbed provides a realistic simulated representation of a tactical network that allows faithful testing of networked systems focusing on hardware in the loop sensors and sensor fusion systems. Systems can be tested using this method in a controlled repeatable environment not feasible in field testing. The system design combines dedicated High Performance Computing (HPC) resources with a scalable, high fidelity network emulation and a Computer Generated Forces (CGF) model to virtually represent the tactical network, force movement, interactions, and communication loads to systems under test. The system is implemented using QualNet network emulation and OneSAF CGF running on a parallel Linux blade system and integrates sensors to the simulated virtual environment through wireless gateways and Ethernet connections. In this paper we present the design framework, current implementation of the system, and real-time performance measurements.

Keywords: *Modeling & Simulation, Wireless Systems, Network & Communications*

I. INTRODUCTION

Realistic testing of tactical wireless sensor networks requires a mixture of live and simulated network nodes and interconnections. Performing a live field test is impractical due to the size of the terrain required and the sheer number of tactical network nodes needed to completely represent the full force structure and equipment (i.e. networks, radios, sensors, weaponry). The testbed approach combines dedicated High Performance Computing (HPC) resources with a scalable, high fidelity network simulation and a Computer Generated Forces (CGF) model to virtually represent the tactical network, force movement, interactions, and communication loads to systems under test. The system architecture with the major interfaces between the HPC and the HWIL interfaces is presented in Figure 1.

The software emulation approach to network testing is more cost effective, scalable, and adaptable than hardware

emulation [1]. The network simulation provides the background traffic present in tactical situations and transports sensor data to consuming applications, with realistic estimates of terrain effects, delivery time, packet loss, collisions, and bandwidth availability. By providing interfaces to actual tactical hardware, the wireless sensors can be tested in a reliable and repeatable way not available through typical field level test methods.

II. SENSOR TESTING

Historically, sensor testing has been accomplished using standalone test methods in conjunction with a suite of modeling and simulation tools. Laboratory Hardware-In-the-Loop (HWIL) test methods are considered the highest fidelity alternative to live field testing because of the inclusion of the actual hardware and software in the test as opposed to mathematical models [4]. In-band stimulation of systems with visible, infrared, acoustic, and seismic sensors has become common place in the test and evaluation community. With the increase in complexity of sensors that include network interfaces and sensor fusion algorithms, the test methods must be enhanced.

The real-time sharing of information from multiple sensors across a wireless network coupled with the use of sensor fusion creates a “system of systems” where the combined performance has the potential to be greater than the sum of the individual pieces. The test and evaluation of the system of systems must take into account the performance difference when sensors share information such as detections, identifications, moving object tracks, photos, and live video across the sensor network. The performance of the system of systems depends greatly on the performance of the network linking the individual nodes.

III. TESTBED REQUIREMENTS

The testbed design requirements are broken into two primary areas: a) real-time requirements, covering the speed of the solution and b) problem size requirements.

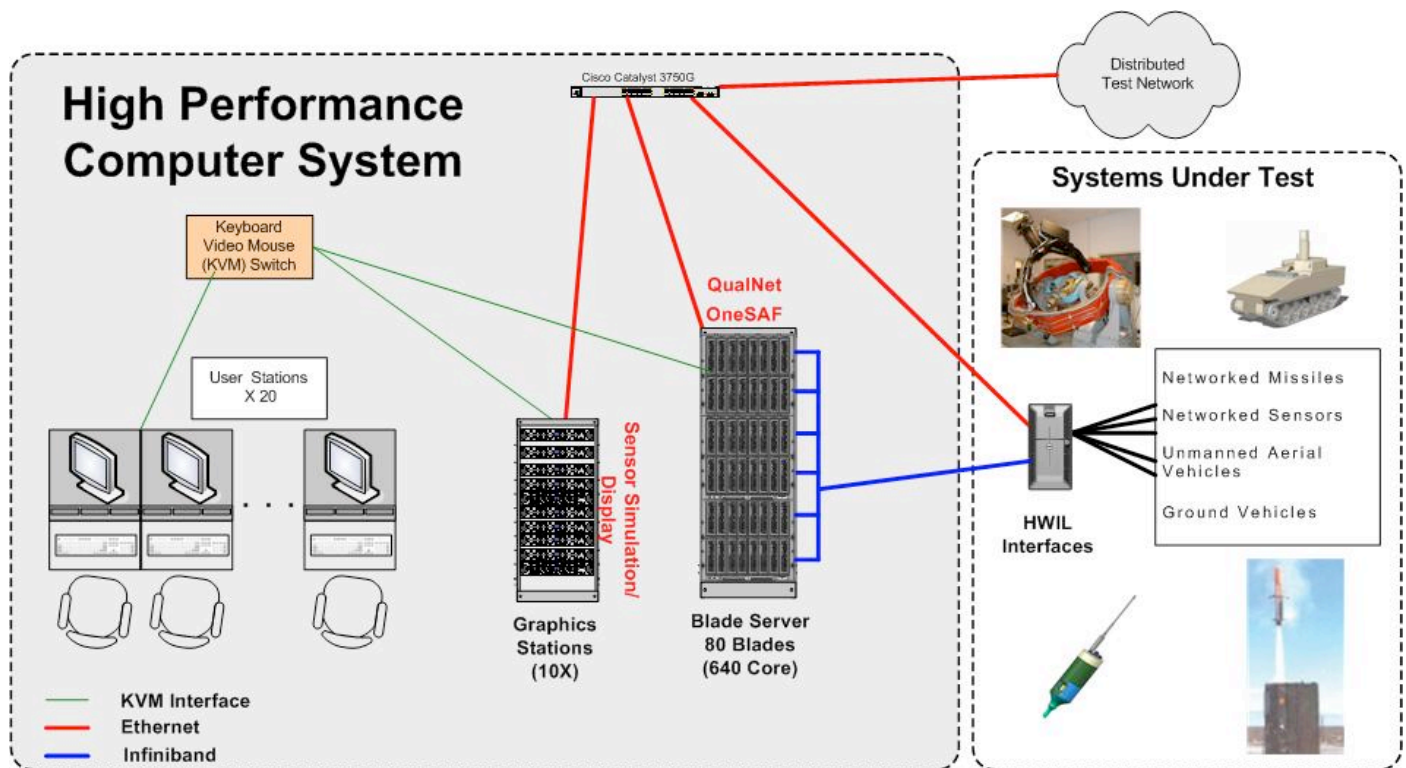


Figure 1. System Interface Architecture

A. Real-Time Requirements

The need for having real wireless sensor hardware connected with the simulated network drives the requirement for operating in real-time. As the size of the modeled network increases, the testbed computing resources must continue to perform all required calculations in real-time. When real-time performance cannot be maintained the test/simulation results are considered invalid.

B. Problem Size Requirements

Network emulation will allow interfacing and testing of a few samples of real hardware with virtual components to produce operationally realistic numbers of network nodes. The future Army Brigade Combat Team (BCT) is anticipated to have between 5,000 and 10,000 networked nodes. The BCT is the target size for the development of the tactical wireless sensor network testbed. Both the network simulation code and the CGF model are required to scale to this magnitude.

IV. REAL-TIME PERFORMANCE

This section of the paper presents the real-time performance evaluation of the wireless sensor testbed using a test scenario where two live sensor systems are exchanging full motion video across a simulated network running on the testbed. Simulated network nodes are added to the environment along with the associated network traffic to provide realistic network loading and computational requirements on the testbed.

The performance testing is divided into three different segments, evaluating the real-time performance as a function of

the number of network nodes, network traffic load, and the fidelity of Radio Frequency (RF) propagation modeling [2]. The size and computational complexity of the problem space is increased until the performance boundaries are established. The real-time performance envelope is determined when the network simulation can no longer keep pace with wall clock time or the HWIL interface linking the live video between the two real network nodes is affected. The live video feed is recorded and monitored for any degradation due to the real-time performance issues associated with the sensor testbed.

The number of processors assigned to perform the network simulation is varied to assess the scalability of the simulation workload [3]. Key network performance parameters such as bandwidth utilization, latency, and jitter are recorded along with the real-time performance boundaries.

The basic function for the HWIL networked sensor systems is providing a video client/server system connected to the simulation computer on the same Local Area Network (LAN). As seen in Figure 2, each of the real computer nodes is represented in the network simulation as green virtual nodes along with the simulated nodes shown in red. All video traffic from the server is transmitted to the client in a User Datagram Protocol (UDP). Each packet is intercepted by the simulation computer, traversed through the required simulated network nodes necessary to link the source to destination. When the packet reaches the client virtual node, the packet is passed to the real client computer over the real LAN completing the interface. VLC media player was used for the testbed transmitting MPEG 4 video at 512 Kbps.

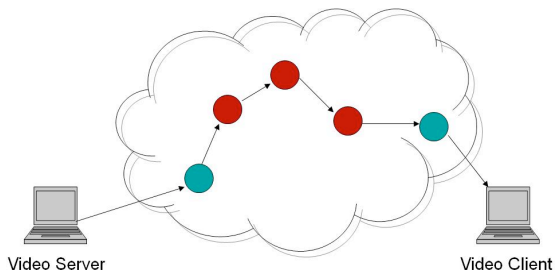


Figure 2. HWIL to Simulation Interface

The HWIL computers and 40 simulated wireless network nodes are considered the baseline experiment case. The baseline scenario includes 5 different subnets with constant rate background traffic generated on each subnet. The 802.11 wireless protocol and a 2 Mbps per channel maximum bandwidth is used exclusively for all testbed experimentation. From the baseline configuration, additional nodes are added in 250 node increments with each grouping assigned to a new subnet and wireless channel. All nodes are added to the region using a uniform placement method. The uniform method divides the region into grids and a node is randomly placed within each grid cell.

Along with each additional node group and subnet, background traffic is added. The background traffic for each subnet consists of two Constant Bit Rate (CBR) sources, providing 3570 byte UDP packets at 33mSec intervals. Each of these packets is set to stagger start in the simulation to create a load on the channel that results in some but not total contention. To remove variability in the test, the subnet containing the two real HWIL nodes does not include additional background traffic loads.

The video server and client applications provide a valuable visible aid, depicting the performance of the network. A more quantifiable approach is also needed in the analysis of the testbed performance. A UDP generator and capture tool called Rude/Crude is used to provide traffic that is time stamped by the server (Rude) and received and logged by the client (Crude). The captured log file is then analyzed to determine if any packets are dropped and to calculate the packet by packet latency. Each of the Rude/Crude computers is connected to a stratum 1 Network Time Protocol (NTP) server to provide synchronized computer clock times. Rude can dispatch UDP packets with a precision of 1 microsecond and Crude can receive and stamp the packets with a maximum error of 200 microseconds [5].

In this early stage of the testbed development, the QualNet network emulation code is run on an Intel 3.0 GHz Quad-core computer with 4 GB RAM running Red Hat Enterprise 5 operating system. The current license of QualNet will support running on one or two processors. For all test cases QualNet is executed in a real-time HWIL mode which does not allow the simulation to run faster than true wall clock time. The real-time execution is required because the testbed includes real network nodes.

During the test execution, QualNet reports the current simulation time and wall clock time enabling the real-time

performance to be evaluated based on the simulation as well as the UDP generator/capture tool. To evaluate parallel speedup in the simulation it is necessary to load the simulation computer with a scenario that cannot be completed in real-time. If the system is not in this overloaded state, the speed of execution using different numbers of processors could not be determined because the simulation will not run faster than real-time when executing in the HWIL configuration.

V. TEST RESULTS

The first step in evaluating the testbed performance is to capture the HWIL interface latency in the baseline configuration (2 HWIL nodes and 40 simulated nodes). The UDP packet generator is programmed to provide four different size and rates for this test. First, packets of 100 bytes at a rate of 64 packets per second is generated then changed to 1000 byte packets at the same rate. Next the rate was changed to 128 packets per second with 1000 and then 500 byte packet sizes. Figure 3 shows the latency results from this test. The 1000 byte packets at 64 packets/sec test case was chosen to match the 512 Kbps generated by the video source. Notice that the latency did not increase when moving from 64 to 128 frames per second for the 1000 byte packets. The performance is as expected because neither rate will drive the channel into saturation. There were no dropped packets recorded for this test case. The UDP packet generator was replaced with the live video stream and the received quality was clear and continuous.

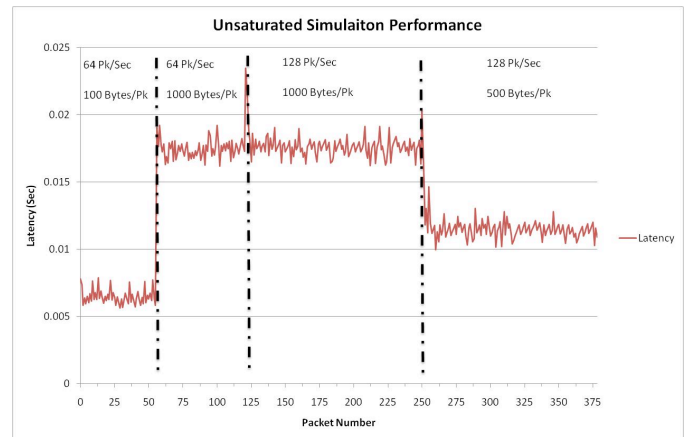


Figure 3. Packet Latency in Unsaturated Simulation

The next step in characterizing the testbed is to find the real-time performance threshold limit with regards to the number of simulated wireless network nodes and background traffic levels. By systematically increasing the number of nodes, subnets, and background traffic while monitoring the real-time performance measures, the threshold is determined to be approximately 750 simulated nodes with the background traffic loads discussed in section IV and executing on 2 processors. Figure 4 shows the latency measurements from the UDP packet generator/logger for this test case.

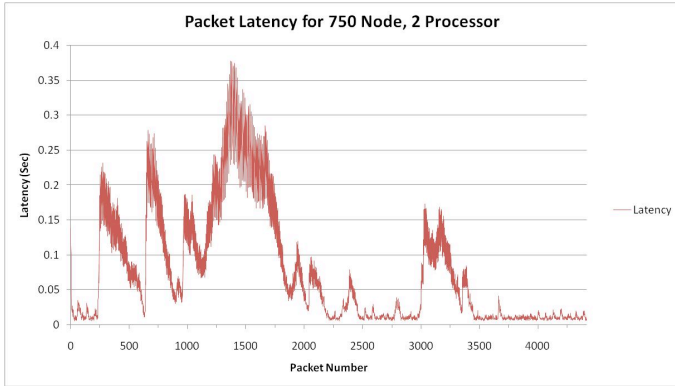


Figure 4. Latency Performance Captured at Real-Time Threshold

Notice that the latency increases beyond the steady state at times but does not grow continuously throughout the test as is seen in the overloaded scenarios presented below. There are no dropped packets in this test case. It is worthy of noting that the measured performance of the testbed is highly dependent on the simulated scenario. For example, the testbed can run 1000 nodes in real-time if the traffic load is not increased from the baseline case. The received video for this test case was perceived to be continuous but included periodic minor flaws.

The next test case increased the simulated wireless network node count to 1000 with the associated increase in background network traffic. This scenario exceeded the real-time capabilities of the testbed and the latencies steadily increased as well as the large blocks of dropped packets, depicted where latency values equal zero. When the live video is transmitted through the HWIL nodes for this test case, only a few video frames are received in a 10 second window and performance degrades over time. This scenario clearly overloads the real-time capabilities of the testbed when running on 2 processors.

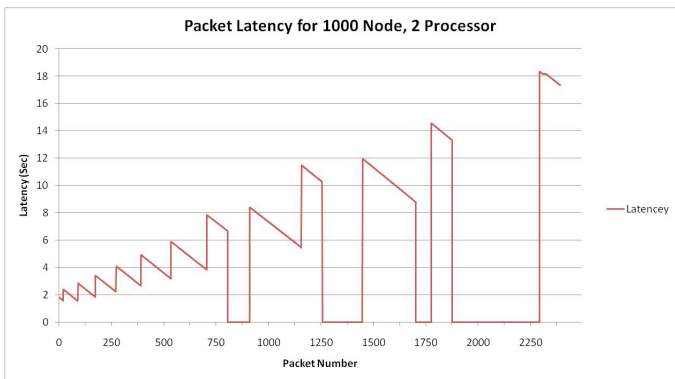


Figure 5. Latency and Dropped Packets from Overloaded Scenario

The next step in the testbed assessment is to evaluate the parallel speedup. Four different test cases are used to evaluate the parallel performance of the QualNet running in the testbed.

The 750 and 1000 network node scenarios are used running each on one and two processors. The simulation time and the real-time (wall clock time) are recorded and are plotted in Figure 6. For the 1000 node case the parallel efficiency is approximately 65%.

$$Eff = \frac{\frac{226sec}{2processors}}{172sec} * 100 = 65\%$$

There are several load partitioning techniques that can increase the parallel performance of the testbed. Again the application scenario has a major effect on the performance with regards to the parallel speedup and efficiency.

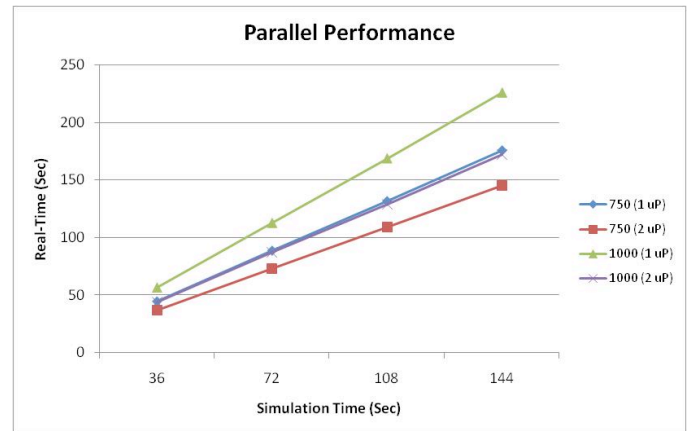


Figure 6. Graph of Parallel Performance

The last real-time performance parameter, fidelity of RF propagation modeling, proved most challenging. Using the uniform node distribution approach generated cases where connectivity between nodes using a low fidelity propagation model could be completed while the same nodes using the higher fidelity model could not communicate. This created test cases that could not be directly compared with regards to real-time performance because the difference in the established wireless connectivity resulted in different load levels on the simulation computer. Extreme care must be taken to generate scenarios that do not change connectivity status when changes are made with the RF propagation model fidelity.

VI. CONCLUSIONS

The augmented wireless sensor testbed performed well for this stage of the development and integration. It is established that using the scenarios outlined in this testing, approximately 750 simulated wireless network nodes can be simulated on a two processor computer while simultaneously interfacing to two external HWIL networked systems passing live streaming video. Through this evaluation it is determined that general performance thresholds can be measured but the results are highly scenario dependent. The true test of real-time performance will be running the intended scenario at the required level of fidelity. Parallel execution proved beneficial

but operator inexperience with load distribution techniques limited the resultant parallel benefit.

Several areas for follow-on research and testing are identified. More elaborate test scenarios are needed to evaluate the performance impact with increases in the complexity of the propagation modeling. Better utilization of parallel load assignments is needed to fully receive the real-time parallel performance enhancements.

The implementation of the tactical wireless sensor network testbed will enhance the test and analysis of system performance in a realistic, real-time, high fidelity simulated environment not achievable through standard test processes. The testbed will allow the community to measure large tactical wireless sensor network performance such as throughput, dropped packets, channel interference, jamming, bottlenecks, power consumption, and reliability just to name a few. The primary advantage of this architecture is the inclusion of live hardware in the test which will be immersed in an augmented environment that allows the item under test to perceive and respond to stimulus just as it would in the real world. Each layer of the network can be tested due to the high fidelity simulation afforded by utilizing parallel processing to maintain real-time performance.

REFERENCES

- [1] G. Werner-Allen, P. Swieskowski, M. Welsh, "MoteLab: A Wireless Sensor Network Testbed", *IPSN 2005 Fourth International Symposium on Information Processing in Sensor Networks*, page(s): 483- 488, April 2005
- [2] B. Hamida, G. Chelius, J. Gorce, "Scalable versus Accurate Physical Layer Modeling in Wireless Network Simulations", *22nd Workshop on Principles of Advanced and Distributed Simulation*, page(s) 127 – 134, June 2008
- [3] Z. Ji, J. Zhou, M. Takai, J. Martin, R. Bagrodia, "Optimizing Parallel Execution of detailed Wireless Network Simulation", *IEEE Proceedings of the 18th Workshop on Parallel and Distributed Simulation*, 2004
- [4] F. Almendinger, K. LeSueur, "Design Tradeoffs in the Development of the Advanced Multispectral Simulation Test Acceptance Resource (AMSTAR) HWIL Facilities", *Proceedings of the SPIE*, Vol. 6544, April 2007
- [5] B. Van den Broeck, P. Leys, J. Potemans, J. Theunis, E. Van Lil, A. Van de Capelle, "Validation of Router Models in OPNET", *OPNETWORK 2002*, Washington D.C., 2002.