Demo: Mobile Network Performance Evaluation Using The Radio Frequency Network Channel Emulation Simulation Tool(RFnest™)

Justin Yackoski, Babak Azimi-Sadjadi, Ali Namazi,
Alexey Bogaevskiy, Jason H. Li, Yalin Sagduyu, and Renato Levy
jyackoski, babak, anamazi, abogaevskiy, jli, ysagduyu, rlevy@i-a-i.com
Intelligent Automation, Inc., Rockville, MD, USA

ABSTRACT

In this demonstration, we present new capabilities of Intelligent Automation Inc. (IAI) Radio Frequency Network Emulation Simulation Tool (RFnest™) [1][1]. RFnest™ is a Field Programmable Gate Array (FPGA) based network channel emulator that allows all of the channels of a full network of radio nodes to be emulated in real time, with all communication nodes experiencing a realistic channel impulse response. This allows RFnest™ to be used for protocol testing, replaying field tests, and model validation. RFnest™ has a modular design with three main capabilities: 1) FPGA based emulation hardware with RF front ends that allows nodes with real radios to send their RF signal over an emulated channel without any modification to the radio, 2) modeling of time-varying channel impulse responses within the emulation hardware, with channel properties based on mobility defined with a scripted or interactive Graphic User Interface (GUI) environment, and 3) integration with network emulators and monitoring functionality that allows the user to instantiate, manage, and monitor real and virtual network nodes within the scenario. In the new version of RFnest™ we now support:

- Frequency programmability: The new RFnest™ allows users to change the center frequency through software from 800 Mhz to 2.7GHz.
- Increased bandwidth: RFnest™ supports a bandwidth of 60 MHz.
- Higher fidelity channel modeling: Through software and hardware updates RFnest™ support wireless channels with up to 20 taps and a wide range of Doppler frequencies.
- MIMO: RFnest™ now supports different combination of MIMO channel between transmitters and receivers.

We will demonstrate the capabilities of RFnest™ with scenarios, including multi-hop bulk data transfer in mixed real virtual network, bulk data transmission at 900MHz and video transfer between two real nodes.

[1] $www.i-a-i.com/rfnest$

MobiCom'13, September 30–October 4, 2013, Miami, FL, USA.

http://dx.doi.org/10.1145/2500423.2505299 .

Categories and Subject Descriptors

B. Hardware
B.4 INPUT/OUTPUT AND DATA COMMUNICATIONS
B.4.4 Performance Analysis and Design Aids**
Subjects: Simulation**

Keywords

Network, Emulation Simulation, FPGA, Radio Frequency

1. INTRODUCTION

Wireless network evaluation is mainly done either through software simulation such as ns-2/3, Opnet and Qualnet or hardware testbeds such as ORBIT in WINLAB. Both of these approaches lack fidelity needed for accurate and realistic evaluation of a network. Software simulations usually use simplistic physical layer modeling that ignores effects such as nonlinearity, filtering, and inter modulation that are caused by hardware. On the other hand testbeds are not repeatable and controllable and cannot emulate a representative geometry of a specific scenario.

The goal of RFnest™ is to provide a repeatable and controlable propagation environment for a network of wireless nodes (currently 12 nodes and in near future up to 96 nodes) so that radios with real MAC, Network, and Transport protocols communicate with each other in a dynamic RF propagation environment. In short RFnest™ provides:

- network evaluation in a **controlled, repeatable, and realistic mobility model environment** with the **same radios used in the field.**
- a **hybrid** software/hardware network emulator to provide **scalability** as well as **high fidelity**.
- **replaying of field tests** with all their complexity in a lab environment.
- **model validation** by creating identical scenarios for real radios and radio models.

Figure 1: RFnest™setup.

2. HARDWARE

Several radios can be hooked up to RFnest™and the channel between all of them (full mesh) is emulated by RFnest™. Essentially, RFnest™is a bank of Software Defined Radios (SDRs) that are all connected to a central powerful processing unit (Virtex7 FPGA). The SDRs interface with the radios under test, and the central FPGA emulates the channel between them. Figure 1 shows the RFnest™hardware and software setup with connected radios.

The 12 node RFnest™currently consists of one main Xilinx Virtex7 FPGA for emulating the wireless channels and 12 RF interface nodes. Each RF interface node includes a Spartan6 FPGA, an analog-to-digital Converter (ADC), a digital-to-analog converter (DAC), and an RF front end. This hardware (HW) interfaces with software components, including the channel models, through a Gigabit Ethernet link.

The current HW has the following specifications:

- RF Band: 800MHz to 2.7GHz
- Instantaneous Bandwidth: 60 Mhz
- Sample Resolution: 16-bit DAC, 12-bit ADC
- Output Dynamic Range: 96 dB (digital)

3. SOFTWARE

[2]The RFnest™software architecture is shown in Figure 2. In a wireless network, there are essentially two aspects of each node that must be considered. First, the operating system stack, meaning the network layer and above, must exist or be emulated. Second, each node must have either a real wireless radio or a virtual radio via MAC and PHY layer simulation model. Building upon existing methods of creating and connecting these aspects, we have created additional components which allow an integration between the real and virtual radios [1].

CORE (Common Open Research Emulator) [2] is used to manage the scenario being tested, including creating the emulated virtual nodes via Linux network namespaces, configuring EMANE (Extendable Mobile Ad-hoc Network Emulator), allowing the user to move nodes via the GUI, and configuring the routing and other network behaviors on both the real and virtual nodes. CORE uses EMANE's event notification API and PHY/MAC models.

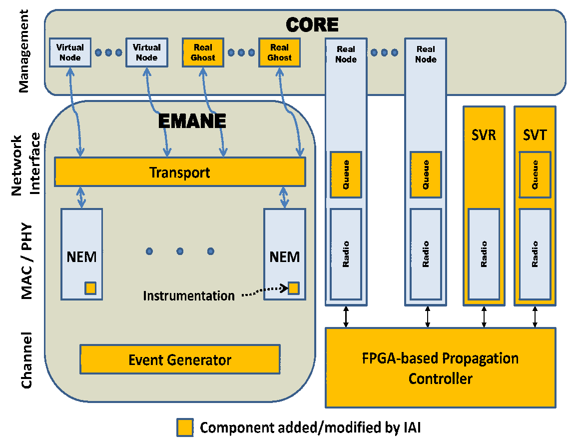

Figure 2: Our EMANE- and CORE-based architecture.

EMANE [3] provides a simulation infrastructure for wireless communication that includes a library of various PHY and MAC models. These models can be either used directly or modified to simulate new protocols. EMANE provides several key features that are heavily used by our emulation architecture. EMANE has an event service API for use by other emulators (such as CORE), mobility generators, terrain data, as well as various PHY, MAC, and other models provided both by EMANE and by third parties. This allows us to easily integrate with existing radio models and other software written for EMANE with little to no changes. For example, we use the EMANE event service to receive node position updates from CORE.

EMANE provides virtual network devices which allow emulated nodes (i.e., those created by CORE) to communicate over what appears to be a normal network device. The virtual network device serves as an entry point for providing packets to and from the PHY and MAC models. We have made significant use of this extendable nature to develop new or modified EMANE components.

[2]Parts of the text in this section are taken from [1]

To allow interaction between real nodes using our emulation hardware and virtual nodes created by CORE and EMANE, we have created an EMANE transport which serves as a bridge between data sent between the EMANE radio simulation models and the Surrogate Virtual Receiver (SVR) / Surrogate Virtual Transmitter (SVT) (see [1] for more details). In EMANE, a transport is responsible for transferring packets between each node's OS stack and the corresponding simulated MAC and PHY layer. By monitoring the MAC addresses specified in packets, our EMANE transport decides whether to keep each packet sent by a virtual node within the "virtual world" (i.e., within EMANE's models) or to give the packet to the SVT. Similarly, packets received from the SVR which are addressed to a virtual node are injected into the EMANE simulation.

4. DEMONSTRATION

In this demonstration we will connect several radios with different frequency band and bandwidth in a dynamic propagation environment and we show the performance of the wireless network as a function of time. The overall demonstration scenario is shown in Figure 3.

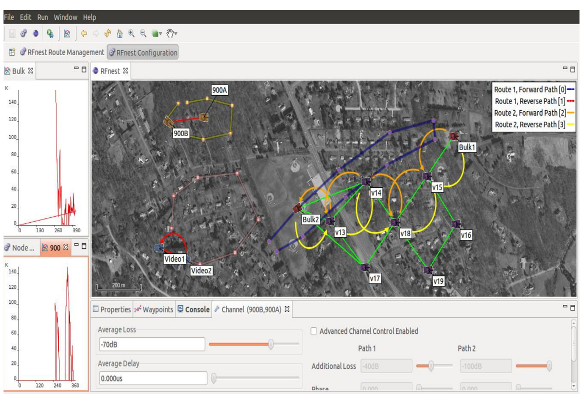

Figure 3: Emulated scenarios using RFnest^{TM};multiple nodes and scenarios are demonstrated at the same time

The details of the demonstration are described below:

- **Multi-hop Bulk Data Transfer in Mixed Real Virtual Network at 2.4 GHz:** In this scenario, two real radios are sending bulk data (e.g. a file transfer) and we want to observe the behavior of the network at a larger scale. To do this, we add a number of virtual nodes (created using CORE and EMANE) to the network. These virtual nodes communicate with the real nodes via surrogate radios controlled by RFnest^{TM}. The virtual nodes become part of the routing topology and actually carry the packets. As the real node moves throughout these virtual nodes, the routing protocol adjusts appropriately. In this scenario neither the real nor the virtual nodes are aware if they or their neighbors are real or virtual. This scenario allows us to see that the behavior of particular routing protocols which can be unstable when many routing options are present.
- **Bulk Data Transmission at 900MHz:** In this scenario, two radios are configured to operate at 900MHz frequency and the are sending bulk data. As one of the nodes moves back and forth, we can observe the changes in throughput shown on the graph labeled "900".
- **Video Transfer Between Two Real Nodes at 2.4 GHz:** In this scenario, live bi-directional video is being sent between the two blue nodes (Video1 and Video2). The real-time throughput is shown on the left side of the figure, the routing paths being attempted are shown by blue arrows, and the signal strength is indicated by the straight lines between nodes. We demonstrate how an aircraft flying over a mountainous terrain communicates a video capture of the environment to a ground station. The wireless multipath channel is estimated using IAI's ray-tracing based channel-modeling software that accounts for the effects of terrain and antenna patterns of aircraft and ground station. The multipath channel coefficients are then communicated to RFnest^{TM}that emulates the channel so that real radios operating within 2.4 GHz representing the aircraft transmitter and ground station receiver communicate over the estimated wireless multipath channel. The effects of the channel, such as blockage of line of sight, are captured in the quality of the video feed received by the ground station receiver.

5. REFERENCES

[1] J. Yackoski, B. Azimi-Sadjadi, A. Namazi, J. H. Li, Y. Sagduyu, and R. Levy, "RF-NEST: Radio Frequency Network Emulator Simulator Tool," MILCOM 2011.

[2] J. Ahrenholz, C. Danilov, T.R. Henderson, and J.H. Kim, "CORE: A real-time network emulator", Proc. of IEEE MILCOM, Nov. 2008, pp 1-7, San Diego, CA USA.

[3] , Naval Research Laboratory, "Extendable Mobile Ad-hoc Network Emulator (EMANE)", url = "http://cs.itd.nrl.navy.mil/work/emane/.