

# Demonstration of Plug-and-Play Cognitive Radio Network Emulation Testbed

Sohraab Soltani\*, Yalin E. Sagduyu\*, Jason H. Li\*, Jared Feldman<sup>‡</sup> and John Matyjas<sup>‡</sup>

\* Intelligent Automation, Inc., Rockville, MD, USA

<sup>‡</sup> U.S. Air Force Research Laboratory, RITF, Rome, NY, USA

Email: {ssoltani ysagduyu, jli}@i-a-i.com, {jared.feldman, john.matyjas}@us.af.mil

**Abstract**—In this demonstration, we present the capabilities of a comprehensive cognitive radio system, CREATE-NEST. We provide hardware demonstration of a Cognitive Radio nEtworKing ArchiTecturE (CREATE) over the Network Emulator Simulator Testbed (NEST). CREATE is a distributed and scalable network architecture operating without a common control channel and is equipped with cross-layer design functionalities. NEST provides plug-and-play cognitive network implementation with software-defined radios (SDRs). In this demonstration, we use eight USRP N210 radios, each running full network protocol stack and communicating with each other over multiple hops and different frequency bands. Instead of over-the-air transmissions, NEST uses RFnest (Radio Frequency Network Emulator Simulator Tool) to generate the air environment by changing channel properties digitally. We demonstrate the capability of cognitive network nodes to discover local neighborhood, sense the spectrum, estimate channels, dynamically access the spectrum, select channels, route traffic, and avoid congestion, primary users and RF interferers. We present different topology and mobility scenarios and show the cognitive radio performance (throughput, delay, energy, and overhead) with interactive GUI.

**Keywords**—*Demonstration; cognitive radio network; software-defined radio; implementation; testbed; USRP; GNU Radio.*

## I. INTRODUCTION

This demonstration presents the implementation of plug-and-play software-defined radio (SDR) emulation testbed for distributed cognitive radio networks. We evaluate cognitive network functionalities of a complete end-to-end Cognitive Radio nEtworKing ArchiTecturE (CREATE) on a high-fidelity network emulation simulation testbed (NEST). Cognitive networks provide the flexibility to adapt system parameters with respect to spectrum dynamics and therefore they require extensive evaluation in realistic wireless setups.

There are mainly two approaches for wireless network evaluation: software simulations or hardware testbeds. Both of these approaches lack the fidelity needed for accurate and realistic evaluation of a wireless network. Software simulations typically use simplistic physical layer modeling that ignores hardware effects such as nonlinearity, filtering, and intermodulation. Examples of software simulators include ns-2/3, OPNET and QualNet. On the other hand, testbeds are typically static and cannot represent general, and, in particular, mobile

network topologies. In addition, testbeds use over-the-air radio transmissions that cannot be reliably controlled or repeated over time. Examples of cognitive radio or SDR testbeds [1] include the ORBIT testbed in WINLAB [2] and the CORNET testbed in Virginia Tech [3].

To fill the gap of realistic cognitive radio network testing and evaluation, we develop a repeatable, high-fidelity network emulation and simulation capability, NEST, that can reflect realistic physical channel and dynamic spectrum effects in cognitive radio networks. NEST builds upon an open source SDR platform with GNU Radio [4] and Universal Software Radio Peripheral (USRP) [5] implementation to support the large-scale cognitive radio network testbed experimentation with asynchronous and distributed spectrum negotiation and coordination channels. Using Radio Frequency Network Emulator Simulator Tool, RFnest<sup>TM</sup> [6], real signals are sent over emulated channels among cognitive nodes with dynamically controlled topology, mobility and channel impulse response properties (RFnest<sup>TM</sup> is a commercial off-the-shelf product previously developed by Intelligent Automation, Inc.). NEST allows replaying field tests in a lab environment with SDRs perceiving and learning dynamic environments thereby adapting to end-to-end goals. RFnest provides a repeatable and controllable propagation environment for a network of wireless nodes so that radios with real network protocols communicate with each other in a dynamic RF propagation and mobility environment that we can replay in RFnest.

## II. TESTBED

NEST consists of eight USRP N210 radios communicating over RFnest. Each radio is connected to a laptop that runs an identical code on GNU Radio. The implementation is mature and all signals are confined to RF cables without any over-the-air transmissions. Therefore, there will be no issue of causing or receiving destructive interference from other concurrent RF demonstrations. The testbed can replace some of the USRPs with other radios (RouterStation Pros) to implement primary user or RF interferer effects. This demonstration requires a table with an approximate size of 3m by 0.75m. The testbed setup is shown in Figure 1. The air environment is provided by RFnest to digitally control the wireless channel properties (path loss, fading, delay, multipath, Doppler, etc.). We will use RFnest GUI to set up the dynamic network scenario (with different mobility effects) and run end-to-end traffic flows between source-destination pairs. Each radio runs an identical code implementing CREATE. The performance is monitored with an interactive GUI (shown in Figure 2) at each node.

DISTRIBUTION A. Approved for public release; distribution unlimited. (AFRL/RITF; 88ABW-2013-5048). Supported by USAF/AFRL under contract FA8750-12-C-0228. Supported by the Air Force Office of Scientific Research under STTR Contract FA9550-12-C-0037. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Air Force.

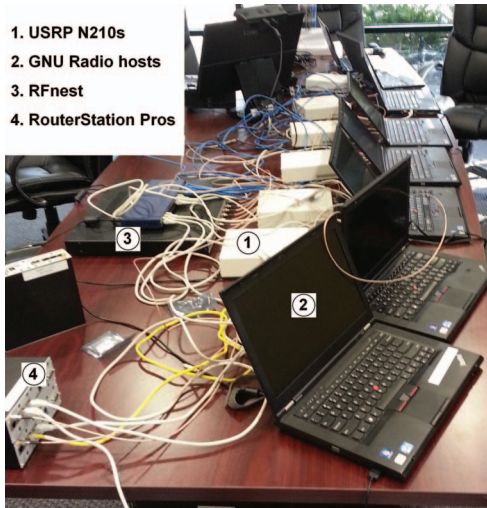


Fig. 1: CREATE-NEST testbed.

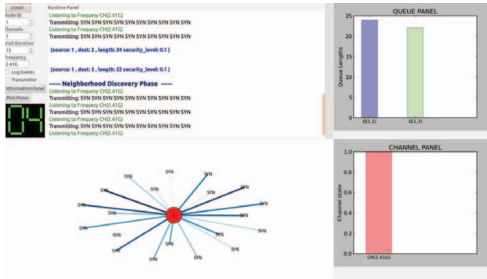


Fig. 2: A snapshot of CREATE-NEST GUI.

In this GUI, we can demonstrate channel and queue properties over time, and measure different performance metrics, including throughput, overhead, and energy consumption. We also demonstrate the total network performance on a separate RFnest GUI that acts as a network monitoring unit.

### III. DEMONSTRATED CAPABILITIES

We demonstrate the capability of cognitive nodes to self-organize into a multi-hop network without a central controller and optimize multiple objectives (such as throughput, delay, and energy) with cross-layer design. A backpressure algorithm [7] is used in CREATE with spectrum awareness to make joint routing and channel access decisions at individual nodes. These decisions are made locally after sensing, estimating and tracking real physical channels without end-to-end path computations and maintenance. The main components of CREATE and its integration with NEST are shown in Figure 3.

The demonstration does not use a dedicated common control channel [8] for information exchanges. Instead, distributed negotiation and coordination channels are implemented asynchronously. Cognitive nodes are then shown to adapt to network dynamics by discovering the local neighbor relations and updating the channel and queue information. Full network protocol stack is implemented for comprehensive cognitive network testbed deployment. The implementation of this protocol stack is independent from the cognitive radio

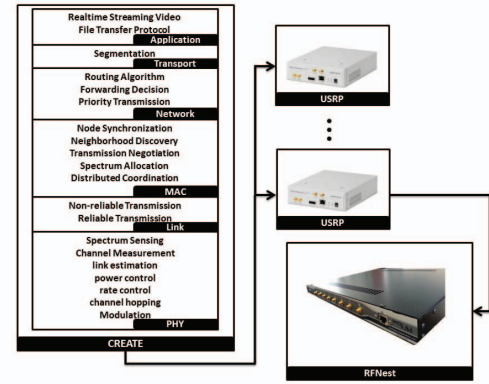


Fig. 3: CREATE system diagram and integration with NEST.

node, each operating with the identical protocol stack. We will demonstrate the following capabilities: 1. *PHY*: Spectrum sensing, channel measurement and link estimation, power control, energy awareness, rate control and frequency channel hopping. 2. *Link*: Channel access with reliable and best effort transmission modes. 3. *Network*: Routing decision and forwarding scheduling, priority transmission. 4. *Transport*: Payload segmentation. 5. *Application*: File transfer.

One specific capability that we demonstrate is how cognitive nodes make decisions locally in response to spectrum dynamics and apply cross-layer design to optimize spectrum allocation and routing jointly. Files are generated at different sources and they are transferred to their destinations. In the meantime, relay nodes select available frequencies for transmission and next hop neighbors to route packets while avoiding interference, congestion, and the potential activities of primary users and RF interferers. This demonstration will show how to set up a cognitive radio network with SDRs and run cognitive network protocols effectively for spectrum awareness.

### REFERENCES

- [1] S. M. Nishra, D. Cabric, C. Chang, D. Willkomm, B. Schewick, A. Wolisz, and R. W. Brodersen, "A real time cognitive radio testbed for physical and link layer experiments," *IEEE DySPAN 2005*.
- [2] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Krems, R. Siracusa, H. Liu, , and M. Singh, "Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols," *IEEE WCNC 2005*.
- [3] "Virginia Tech cognitive radio network testbed (CORNET)," <http://cornet.wireless.vt.edu/trac/>.
- [4] E. Blossom, "Gnu radio: tools for exploring the radio frequency spectrum," *Linux journal*, vol. 2004, no. 122, p. 4, 2004.
- [5] M. Ettus, "Universal software radio peripheral (usrp)," *Ettus Research LLC* <http://www.ettus.com>, 2008.
- [6] J. Yackoski, B. Azimi-Sadjadi, A. Namazi, J. H. Li, Y. E. Sagduyu, and R. Levi, "Rfnet: Radio frequency network emulator simulator tool," *IEEE MILCOM 2011*.
- [7] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1948, 1992.
- [8] B. F. Lo, "A survey of common control channel design in cognitive radio networks," *Journal of Physical Communication (Elsevier)*, vol. 4, no. 1, pp. 26–39, 2011.