WARPnet: Clean Slate Research on Deployed Wireless Networks

Siddharth Gupta, Chris Hunter, Patrick Murphy and Ashutosh Sabharwal
Electrical and Computer Engineering
Rice University
Houston, Texas 77005
{sgupta, chunter, murphpo, ashu}@rice.edu

ABSTRACT

In this demo we present the Wireless Open-Access Research Platform for Networks (WARPnet), a research testbed aimed at performing experiments at the network level. The platform is designed to support not only conventional research areas but also new modalities like cooperative coding and network coding. It is based on the Virtex-4 FPGA, which provides the resources to implement novel MIMO physical and MAC layer algorithms. Additionally, the platform adds an orthogonal wireless backdoor network that allows remote programming, control and monitoring of the nodes.

1. INTRODUCTION

The current research emphasis in wireless networks is shifting from a bottoms-up approach (which views a network as a combination of point-to-point links) to a top-down paradigm which views the whole network holistically. Examples of such concepts include network coding [1, 4, 3] and cooperative coding [2, 6], where nodes collaborate to use more than one route in the network at physical layer and above. These concepts are already being actively researched to be utilized in the next generation of wireless standards. However, there is little or no experimental evaluation of many of the proposed protocols due to lack of high-performance platforms which can implement the novel sophisticated algorithms at any realistic scale at real speeds.

We present the architecture for a wireless platform which is built from ground up to allow clean-slate prototyping of real-time wireless networks, which can both be tested in the lab and in a deployed network. The new platform, labeled WARPnet, extends our current platform WARP [5] in three significant directions. First, it increases the computational resources available in the hardware to allow more complex MIMO designs. Large coding systems such as LDPC also become a possibility. Second, it adds significant memory onboard to allow full Linux implementations as well as memory for data collection and storage. Lastly, it adds an orthogonal wireless backdoor network which allows remote programming, control and monitoring of *deployed* WARPnet-based networks.

2. PLATFORM

To enable reasearch on deployed wireless networks, the platform is designed to address two major requirements. First, it should allow clean-slate programmability of any subset of nodes in the network. Second, the platform should have extensive support for in-depth management and observation at all layers of the experimental networking stack. The overall system architecture is shown in Figure 1.

Controlling the entire stack is crucial as new architectures for nodes are designed, so that the user is not restricted to specific parameters as in off-the-shelf hardware. This control enables a fundamental re-design to take advantage of new ideas and algorithms as they are developed by all wireless research communities.

This requires the development of a board that provides resources to implement complete wireless networks. Each node must have processing power to allow physical layers to run in real-time. These resources must be local on the board for every node to decode data and return acknowledge them on timescales that are realistic for packet based implementations.

We designed the WARP Virtex-4 FPGA Board to achieve the goals. As expected it is based around a Xilinx Virtex-4 FPGA that has 27% more slices than our previous generation boards. It can sustain 50% higher clock rates along with 2GB of memory. There are up to 3 gigabit Ethernet connections for faster in and out flow of data through the network. Up to four flexible radio interfaces are available and perform the baseband to RF conversion without adhering to any particular standard. This allows any wireless algorithm to be prototyped on the system.

Once we have developed an experimental wireless stack on the nodes, observation becomes the primary concern. In a lab scenario this is not an issue as the user is in direct contact with the nodes to extract data. However, once a network is deployed, data extraction from the nodes becomes a stumbling block in design and further development.

To address this we designed the WARP Backdoor Board that gives the user remote control of the experimental stack at every layer. It is designed around a Linux System-on-Chip by

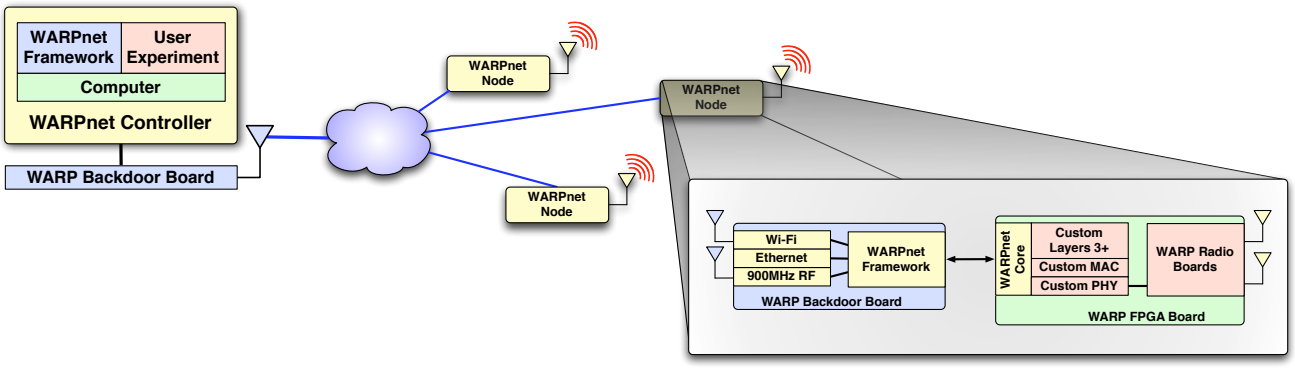

Figure 1: *Overall System Architecture, with WARPnet hooks for control and measurement at every layer*

Axis Communications. There are three major backdoor interfaces: Ethernet, USB WiFi and 900 MHz RF. The board also has reconfiguration and power control of the FPGA Board. New firmware can be sent on any backdoor interface and reconfigure the FPGA. Thus, once the nodes are deployed the user can upload new designs and experiments without disassembling the network.

While control is an important aspect, the Backdoor Board also provides a measurement network. As an experiment is running, real-time data updates can be provided to the WARPnet controller. As each backdoor interface varies in terms of throughput and latency, the update rate of information must be adapted based on the interface used. In addition to real-time updates, stored data can also be transferred once an experiment has neared completion.

3. DEMO

We showcase the abilities of WARPnet in our demo. The setup of the system is shown in Figure 2. The experimental link under observation is between two WARPnet nodes. In this case, it is our wireless reference design running OFDM with 64 subcarriers at 2.4GHz. The data source and sink are laptops connected at either end. The experimental link acts as a wired-to-wireless bridge.

While the experiment is in progress, we will be gathering statistics on this experimental link using the backdoor network. All the three backdoor interfaces will be utilized. A WARPnet controller must have a Backdoor Board connected to it to communicate over the 900 MHz link but can connect to any of the nodes when using Ethernet or WiFi.

The designs in the FPGA are unaware of the presence of the backdoor network. The data of interest is written to and read from a buffer. The Backdoor Board can read and write data to the same buffer. Hence the experiment running is transparent to the observation and control.

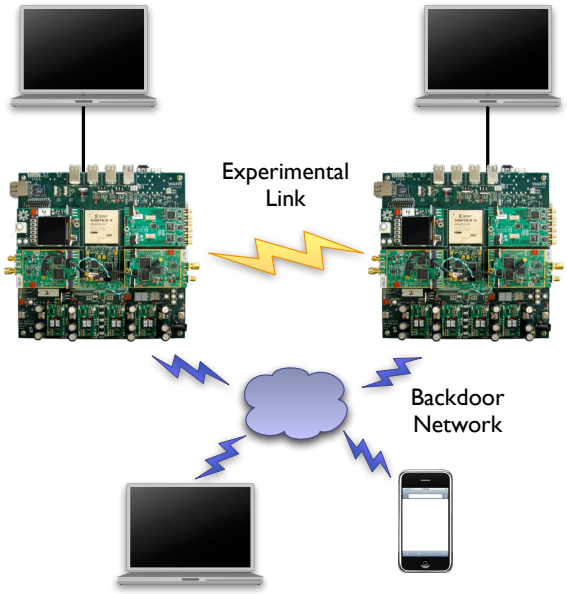

Figure 2: *Demo Layout*

4. REFERENCES

[1] R. Ahlswede, C. Ning, S.-Y. Li, and R. Yeung. Network Information Flow. In *IEEE Transactions on Information Theory*, 2000.

[2] A. Chakrabarti, E. Erkip, A. Sabharwal, and B. Aazhang. Code Designs for Cooperative Communication. In *IEEE Signal Processing Magazine*, volume 24, September 2007.

[3] P. A. Chou and Y. Wu. Network Coding for the Internet and Wireless Networks. In *IEEE Signal Processing Magazine*, volume 24, September 2007.

[4] P. A. Chou, Y. Wu, and K. Jain. Practical Network Coding. In *Allerton Conference on Communication, Control, and Computing*, 2003.

[5] P. Murphy, A. Sabharwal, and B. Aazhang. Design of WARP: A Flexible Wireless Open-Access Research Platform. In *Proceedings of EUSIPCO*, 2006.

[6] A. Stefanov and E. Erkip. Cooperative Coding for Wireless Networks. In *IEEE Transactions on Communications*, volume 52, September 2004.