

CRC: Collaborative Research and Teaching Testbed for Wireless Communications and Networks

Samer S. Hanna
Eng. Math. and Phys. Dept.
Alexandria University, Egypt
samer.hanna@alexu.edu.eg

Arsany Guirguis
Comp. and Sys. Eng. Dept.
Alexandria University, Egypt
arsany.guirguis@alexu.edu.eg

Mahmoud A. Mahdi
Zagazig University, Egypt
qursaana@gmail.com

Yaser A. El-Nakieb
ECE, Virginia Tech MENA
ynakieb@vt.edu

Mahmoud Alaa Eldin
Alexandria University, Egypt
mahmoud_3laa_eldin@yahoo.com

Dina M. Saber
Comp. & Comm. Eng. Dep.
Alexandria University, Egypt
dina.saber11@gmail.com

ABSTRACT

The validation of wireless communications research, whether it is focused on PHY, MAC or higher layers, can be done in several ways, each with its limitations. Simulations tend to be simplified. Equipping wireless labs requires funding and time. Remotely accessible testbeds present a good option to validate research. The existing testbeds have gone a long way in building the infrastructure for managing and operating themselves. Yet, there is still space to improve the administration of resources whether it is nodes, frequency spectrum or storage space. In this work, we present the Collaborative Radio Cloud (CRC), a wireless testbed that enables web/remote access to facilitate research experiments in the field of wireless communications and networks. CRC is built upon the contributions made by current testbeds and incorporates new features to improve the user experience and ameliorate the management of resources. It enables multiple simultaneous users to coexist in the testbed by enforcing an elaborate node isolation policy. Software dedicated to the enforcement of spectrum assignment was developed to further ensure the isolation between concurrent users. A tool that is capable of realizing a desired topology using gain control was developed. CRC also operates a scheduler that optimizes node assignment to maximize testbed utilization. Enhancements to disk image sav-

ing are being developed to reduce the required storage. Other than research, CRC targets contributing to the education of wireless communications and networks by providing experiments and a web interface dedicated to students and instructors.

CCS Concepts

•Networks → Network experimentation; Network simulations; Network measurement;

Keywords

Testbed; Wireless experimentation; Network experimentation

1. INTRODUCTION

The credibility of research results, new governmental policies/standards and prototype testing depends on the method of research validation. Several options exist for the evaluation of wireless research. The most popular option is simulation because of its relative ease and low cost. The problem with simulations is that they make many assumptions to simplify calculations, which might compromise the credibility of results. In addition, large-scale simulations require resources or take durations that could go beyond what is feasible on personal computers. The second option is to acquire radio-frequency (RF) hardware. This option has several drawbacks. First, it incurs a cost that is outside the abilities of many research groups. Then, it adds an overhead of hardware and software setup. Additionally, it is not economical, as RF hardware purchased will often suffer from low utilization. Remotely accessible testbeds offer researchers a more realistic tool for research validation, while avoiding most of the problems associated with the management of hardware. Thus, it can be used for increasing the trust of the proposed systems in an economically feasible manner.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

WiNTECH'16, October 03-07, 2016, New York City, NY, USA

© 2016 ACM. ISBN 978-1-4503-4252-0/16/10...\$15.00

DOI: <http://dx.doi.org/10.1145/2980159.2980161>

Several remotely accessible wireless testbeds exist like ORBIT [15], CORNET [10] and NITOS [12]. Their operators already developed the infrastructure and tools needed for testbed management. Still, there are many shortcomings. The concept of node virtualization, despite being popular in wired testbeds, has been neglected in wireless testbeds which only allow physical node reservations. Scheduling is often performed using a naive first come first served (FCFS) manner, neglecting the optimization that can be done to increase the number of concurrent users. Current disk imaging methods work by saving the entire disk content regardless of the amount modifications made, which leads to redundancy in storage.

In the SmartCI research center at Alexandria University, we implemented CRC, a wireless testbed with a scalable and modular architecture. Hardware-wise, the testbed consists of a controller server, a simulation server and a number of testbed nodes. In this stage, due to budget limitations, only eight nodes are available. This number is to be increased in the future. The controller runs the management software, the simulation server provides users with resources to run simulations with high requirements, while the nodes operate wireless interfaces which are available to the users.

CRC has many new features that aim to improve the user experience and attempt to optimize the utilization of the limited available resources. While other testbeds relied on geometric or arbitrary node placement, CRC nodes were placed based on careful study and data collected during multiple placement attempts. This placement, along with a tool developed to realize topologies by varying transmitter power and receiver amplification, allows users to realize the connections they desire. In CRC, testbed nodes are virtualized based on wireless interface, enabling multiple users who need different wireless hardware to use the same physical node at the same time. CRC's scheduler tries to maximize the number of simultaneous users, by mixing and matching their resource requirements until the best possible combination is found. CRC also developed the first frequency enforcement tool that works with Software Defined Radio (SDR) kits.

Standard tools in testbeds, like experiment controller and disk imaging, are used. At CRC, an enhancement for the disk imaging tool is being developed to avoid storing redundant data. For the simulation server, disk images containing the most notable simulation software are provided. A shared code library started by CRC administrators will be available for experimenters to use or to contribute to. A web portal with an easy-to-use interface allows beginners to access most of the testbed capabilities, while, an interface dedicated to education services also exists.

The rest of the paper is organized as follows. In Section 2, the existing testbeds are briefly discussed along with their limitations. Section 3 presents the new features in CRC. Afterwards, the architecture of the

testbed and its main components are described in Section 4. Section 5 evaluates some of the new features in CRC. The paper is concluded in Section 6.

2. RELATED WORK

Several testbeds are deployed in different places around the world. Some of them target wired experiments or wireless experiments or both. In this section, we briefly present some of the existing *wireless* testbeds. Then, we give a brief overview of the new features we provide in CRC.

ORBIT [15] is a wired and wireless testbed that is designed to achieve reproducible experimentation, while also supporting realistic evaluation of protocols and applications. Its core component consists of 400 nodes placed in 20×20 grid of programmable radio nodes. ORBIT allows only one user to reserve all 400 nodes at a time regardless of the number of nodes he actually needs. Thus, this makes the testbed nodes underutilized most of the time.

CORNET [10] is a wireless networking testbed. It provides a platform through which researchers can develop and test Cognitive Radio (CR) algorithms and applications on SDR nodes. Several problems exist in this testbed. Firstly, it does not provide a disk imaging tool. Hence, software modifications by a user affects the rest of the users. This also makes software across their nodes inconsistent. Secondly, it lacks a reservation system. Meaning that any user can access any free node at any time with no proper management.

NITOS [12] is a testbed that targets wired and wireless experiments. It has a reservation system that allows users to reserve some nodes. It also allows several users to work concurrently. However, it does not guarantee frequency isolation between users, but it relies on a gentlemen's agreement between users that each user only uses the channels he reserves. Also, it does not support any means of virtualization, where any physical node can be used by only one user at a time. Moreover, the scheduling system for resource reservation used is a naive FCFS which may not optimal from the utilization point of view.

CogFrame [16] and CRESCENT [3] target higher layer protocols. By abstracting the MAC and PHY layers, they let designers focus on the routing protocols. They leverage the functionalities of standard computers and Wi-Fi cards, saving the cost of special purpose RF-frontends.

CRC offers new features that are not implemented by any of the mentioned testbeds. It enables isolated concurrent experiments, topology realization based on node gains. Its scheduler optimizes the testbed utilization. It uses an enhancement to disk imaging to reduce storage. It also provides a code library to improve collaboration between researchers. A web portal dedicated for educational purposes is integrated in CRC.

3. CRC NOVEL FEATURES

CRC builds on the pioneering efforts of the current testbeds and introduces a number of new features that aim at improving the testbed utilization, saving resources and providing users with new tools. Some of these features has been implemented while others will be in the future.

3.1 Current Features

In this section, we introduce the new features we added to CRC.

3.1.1 *Isolated Multiple Concurrent Users*

One of the new features in CRC is its ability to have multiple concurrent users access its resources while guaranteeing no interference. To be capable of accommodating concurrent users two challenges exist; the first is limiting each user to his reserved nodes and preventing him from unauthorized access either by mistake or deliberately. The second challenge is to enforce frequency assignments. As the wireless medium, unlike wired, is of broadcast nature, two users conducting experiments on the same frequency can compromise the integrity of each others work. Thus, ensuring that users work on their allocated band is important. NITOS handles only the first challenge. CRC, on the other hand, handles both challenges, making it truly support concurrent experimentation by different users.

3.1.2 *Enforcing Frequency Assignment*

Users, while reserving nodes, are required to reserve frequencies. Confining each user to his reserved spectrum can be done by monitoring for frequency violations. Once a violation is detected, the offending user can be forced out of the testbed and perhaps have his account suspended. The approach we took at CRC to detect such violations is a software based approach. Three wireless interfaces exist currently at CRC; USRP, RTL-SDR, and WiFi. The RTL-SDR is a receiver only, so, it does not cause interference. The WiFi standard enables multiple users to share the same channel and monitoring its frequency usage can be performed through the operating system. This makes the most challenging source for interference is the USRP. The USRPs available at CRC communicate with the virtual machines through USB using packets with a predefined structure. By sniffing these packets from the physical nodes, we can tell what frequency band is being used and hence detect violations.

3.1.3 *Testbed Nodes Virtualization*

Virtualization of nodes is common in wired testbeds and has been used to extend the scale of experiments like in Emulab [19]. None of the existing wireless testbeds uses node virtualization. The smallest unit to be reserved in these testbeds is the physical node. This is explained by the following; in wired networks, virtualization of network resources like interfaces and links is

possible without jeopardizing integrity of results due to the controlled nature of wires. In wireless communication, although some work has been done on the virtualization of WiFi [18], reliable virtualization of SDR peripherals, which are physical layer devices, is difficult. At CRC testbed nodes are virtualized based on interfaces; each physical node runs three virtual machines and each virtual machine is assigned one of the three interfaces. Hence, this setup allows multiple simultaneous users, who need different interfaces, to use the same physical machine.

3.1.4 *Topology Control using Gains*

Some wireless experiments depend on the existence of a specified topology between the nodes. For example, some cognitive radio scenarios require certain connectivity between the nodes. Changing the transmitter power and the amplification at the receiver by changing the transmit and receive gains of the nodes can be used to realize a given topology. Selecting suitable nodes and finding suitable values for the gains using trial and error is a time consuming process. In CRC, we developed a system, that given a desired topology, will find values of the transmit and receive gains for the USRPs to achieve the desired topology.

To build this system, the relevant characteristics of the USRPs were studied, like the relation between the gains and the received power, and the relation between received power and the quality of the link. From these measurements, the required power levels for a connection to exist or not were determined. Using these characteristic values and the values of channel coefficients (which can be measured), the problem of finding the values of the gains to realize a topology can be formulated as an optimization problem. The set of gains that realizes the user's desired topology can be obtained by solving this problem. The exact formulation and the limitations of this method are discussed in [17].

3.1.5 *Maximal Utilization Scheduler*

One of the new features incorporated in our testbed is the use of an automated offline scheduler for reservation, which adopts many techniques to maximize testbed utilization. Several dimensions exist for our scheduler to optimize. In CRC, we allow sharing the testbed nodes between multiple users. The day is divided into 24 one-hour time slots, and each user can request one or more slots for his experiments. Also, we allow simultaneous users to reserve and work on different frequencies. In addition, the node virtualization implemented in CRC adds another dimension to be considered. Our scheduler also performs mapping of the requested topologies to our testbed nodes with the help of the topology control using gains system.

The goal of this reservation system is to handle the incoming requests specifying time preferences, topology, interfaces, requester type, and other characteristics, and to select the largest subset that can be fulfilled,

thus, maximizing the testbed utilization. The proposed scheduler is the first automated mapping-scheduling algorithm on a virtualized wireless testbed. The offline reservation has two stages. First, it finds all possible node-mappings for each requested topology, which is done using an induced sub-graph isomorphism based algorithm. Second, it decides which requests, with which mappings, to be served in each time slot on which nodes, in a slot-by-slot manner, using a Genetic Algorithm (GA) heuristic. Each combination of requests are encoded as a chromosome (possible solution), with their mappings, and a fitness function values the quality of a solution by the achieved requirements and priorities of requests, and penalizes any unsatisfied constraint (Hardware, frequency, time, . . . etc.). Detailed description can be found in [4]. Another module is implemented that allows reservation of remaining resources in-time in a FCFS manner.

3.1.6 Storage Efficient Disk Imaging System

The users' images are always derived from the base image provided by the administrators of CRC. Users may perform some changes on the image and then save it back to the storage server using Frisbee [7]. The drawback of Frisbee is that it stores the entire disk image which leads to having duplicate data. To avoid this, we use a proven data deduplication software which is Venti archival storage system [13]. Venti is a block level storage system, it identifies duplicate blocks and avoids data duplication. Our work was on the integration of Venti and Frisbee to get the benefits of both; Venti avoids duplication, while Frisbee enables fast and scalable image transfer.

3.1.7 Code Libraries

The collaboration between different researchers from different projects and countries can increase their productivity and competitiveness. Currently, there is no well-known environment that combines all the work of researchers in the wireless communications and networks fields.

At, CRC we encourage the collaboration between researchers by offering the option of saving any newly implemented technique, whether it is a simulation or a testbed experiment, as part of a shared library in CRC. This experiment database is initiated with some popular and available open-source protocols for simulations and testbed experiments ready to be used by any user. This enables researchers around the world to use any newly implemented work for learning, running comparisons, or extending it. The presence of such facility is expected to have a desirable impact on the wireless research.

3.2 Proposed Features

In this section, we present features that are meant to be included in CRC in later stages of the project.

3.2.1 Resumable Simulation

Large-scale simulation, even on a high specification machine, takes long periods that can span weeks or even longer. These durations might be bigger than the maximum allowed simulation server time slot or simply might exceed the duration the user had reserved. Other sources of unexpected failures such as power outages exist and threaten the continuity of such experiments. To secure users against such cases and allow them to recover from any lapses, at CRC, we aim to provide a backup system which takes snapshots of the execution state of the virtual machines at scheduled times. Once a failure occurs or a user is preempted due to the expiration of his reservation, the user will have the option to recover the system to the last working state.

3.2.2 Unified Scripting Language for Simulation and Testbed

Currently, the simulation tools and experiment controller use different scripting languages. This means that a user after simulating his scenario, in order to test it using hardware, will have to reimplement his work. He will also have to be exposed to the different software tools used in testbed experiments and the hardware parameters. Such additional effort can deter users from real world experimentation. At CRC, we intend to develop a Unified Scripting Language. Using this language the user will code his experiment scenario only once abstracted from the hardware details. Then, he will be capable to run it as a simulation or a real world experiment with ease.

3.2.3 Node Emulation

In some experiments, researchers are interested in probing and evaluating the behavior of a small number of nodes in a system consisting of a large number of nodes. For example, in evaluating a wireless routing protocol, the researcher might want to physically test a scenario on a few nodes that are part of a bigger wireless network. Running the entire scenario on real hardware can be resource exhaustive and of little value as the user is not interested in the exact behavior of every single node but the outcome of the big network on a subset of nodes. For such scenarios, combining real world and simulation experiment can be advantageous. At CRC, we intend to develop a node emulation module. This module will allow the user to combine simulation with experimentation on physical nodes using the unified scripting language.

4. TESTBED ARCHITECTURE

CRC testbed is designed to serve both research and education. For researchers, it enables them to test their work either as a simulation or using real hardware. For education, it gives students the opportunity to gain a hands-on experience with the concepts they study in communications and networking courses. It also pro-

vides an intuitive interface that enables easy management of labs for instructors.

The architecture of the testbed is shown in Figure 1. The controller server runs all the software used for the administration of the testbed. The simulation server provides users with computational resources along with preinstalled simulation tools, while testbed nodes provide access to wireless interfaces. The web portal and SSH (the Secure shell) are used to provide access to users according to their permissions, whether they are researchers, instructors or students. The portal additionally provides an interface for the scheduler which is used to grant or deny users' requests according to resource availability. An authentication module enforces scheduler decisions by banning unauthorized users from accessing the testbed whether using the web or SSH. Once a request has been granted, the user can use the existing tools; the experiment controller, disk imaging or topology control. Other features like mobile nodes and the mobility control software, the node emulation and the support of integration with other testbed will be added in later stages of the project.

4.1 Hardware

4.1.1 Nodes

A testbed node consists of a PC connected to a number of wireless interfaces. Each testbed node runs an equal number of virtual machines (VMs), each of them is connected to one of these wireless interfaces. The users don't have access to the physical machines and can access the wireless devices through the VMs. This setup allows having multiple users, wanting different interfaces, to work on the same node simultaneously. In the current stage three wireless interfaces are attached to the PC, a USRP B210 [5], an RTL-SDR [1] and a WiFi module. In future stages, mobile nodes will be added.

4.1.2 Node Placement

Several factors contribute the wireless topologies that can be realized in the testbed like the RF characteristics of the wireless interfaces (for example maximum transmitted power), the distance between the nodes, and the obstacles between them. Some testbeds like ORBIT relied on a uniform geometric placement of nodes. This placement causes a high level of interference between nodes, limiting the topologies that can be realized. In order to place nodes in a way that permits a wide range of topologies to be realized, the RF interfaces characteristics were carefully studied. Several positions were tested and many measurements were performed before selecting the placement of the nodes.

4.1.3 Simulation Server

Using this server, testbed users can do simulations in CRC. The simulation server offers users several benefits. First, thanks to the high processing power of

this server, many users can deploy and run large-scale experiments (involving large number of nodes, connections and other network parameters) each in his VM. In addition, CRC administrators offer some prepared disk images with the most popular network simulation tools (including NS2 [11], NS3 [2] and Mininet [9]) to be used by the testbed users. Using these images alleviates the overhead of preparing the simulation environment which can be time consuming. Finally, for the educational purposes, CRC simulation server is equipped with tutorials for students and beginners on the basic concepts and protocols in networks subject. This can give students the required exposure to the newly learnt concepts.

4.2 Software

4.2.1 Web Portal

The purpose of testbed portal is to provide web access to both the testbed infrastructure and/or simulation server. Through the portal, users can check available testbed nodes and simulation virtual machines, reserve and access them.

Additionally, a dedicated education interface is integrated in CRC. Its purpose is to help instructors manage labs for courses on the testbed. Among the functionalities it provides are bulk or mass resource reservation, tracking students' progress and controlling their accounts. It also enables instructors to create courses, develop experiments and assign them to students. Students, on the other side, will have access to the instructions and files provided by the instructor. They will be able to run the required experiments, visualize the results and submit the requirements using the portal.

4.2.2 Experiment Controller

OMF [14] is used as the testbed management and measurement framework. It allows users to write an experiment script, through which a user can specify the nodes participating in their experiment, what software they will run and with what parameters. They can also describe the sequence of events needed to realize the experiment. Using OML (OMF Measurement Library), all results are collected and stored in a SQLite database. It was first developed to be used in ORBIT testbed, now it is deployed in many testbeds worldwide and has support for federations with other testbeds.

4.2.3 Disk Imaging

Disk imaging allows users to load and save an entire disk image on their reserved nodes. Its purpose is to allow users to install any software or make any modifications they want on the nodes without affecting other users or having to repeat the changes every time. This is explained by the fact that users will start their session by loading their images, thus, erasing all the changes made by previous users.

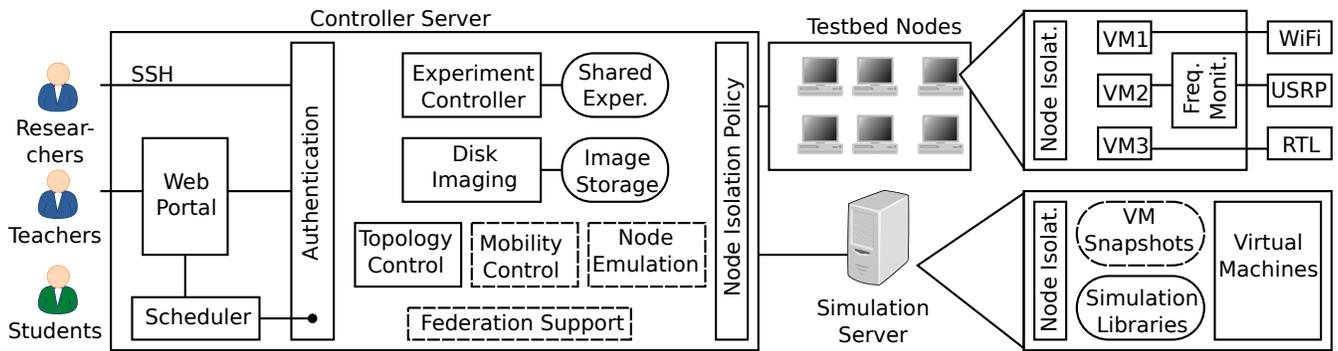


Figure 1: Architecture of Testbed. Dashed blocks represent prospective features.

4.2.4 Node Isolation Policy

The purpose of node isolation is to limit users to the nodes they reserved. Unauthorized access of nodes can occur from the controller or from the user’s reserved nodes. To prevent it, the isolation policy is implemented using firewall rules on the controller and the physical nodes. These rules are updated according to the reservation schedule.

5. EVALUATION

In this section, we evaluate the node virtualization which introduced in CRC. The evaluation also demonstrates the benefits of having a simulation server.

5.1 Testbed Nodes Virtualization

Virtualizing testbed nodes increases the possible number of concurrent users in the testbed, as multiple users who need different interfaces can work simultaneously on the same node. This setup, despite being advantageous from the point of view of testbed utilization, needs to have its effects evaluated from the user perspective, which is what we will do in this section.

In a testbed that operates SDR kits, most of the processing takes place in the computer. The computational capabilities of the PC can be the bottleneck for achieving high data rates. To evaluate this, the system depicted in Figure 2 was used. It consists of data source that generates a continuous stream of zeros, which are modulated using QPSK then demodulated and then passed to a buffer that measures the average throughput over a period of 2 minutes. This system was implemented using GNU Radio [6]. The results obtained from this system estimate the maximum rate at which the user can transmit and receive data simultaneously. When operating this SDR system on the physical machine using all 8 cores a rate of 2.82 Mbps was obtained. While when limiting it to use only 2 cores on the physical machine, the throughput was 1.77 Mbps compared to 1.73 Mbps when operating from the VM (which has 2 cores assigned to it). These results show that a small degradation in performance occurs when

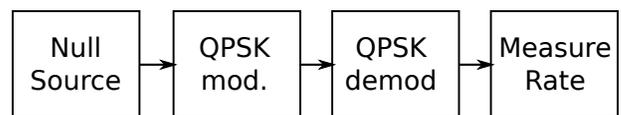


Figure 2: Block diagram used to evaluate processing power.

moving from the physical machine to the VM using the same number of cores, while having access to all cores can give a performance boost.

We then evaluate the effect of simultaneous users on the same node. The testbed node consists of PC, with 8 cores and 8 GB of RAM, running three virtual machines. Each VM is assigned two cores and 2 GB of RAM. The throughput of the previously described system was measured from one of the VMs, while the PC was loaded as follows: (1) The physical machine OS was running an application that is fully utilizing a single core to simulate tasks running on the physical machine (2) The other two virtual machines were running an application that is utilizing their two cores at a percentage that was changed throughout the experiment to simulate the load made by concurrent users on the same node. The result of varying the percentage utilization of the two VMs (load by other simultaneous users) over the throughput of the first VM are shown in Figure 3. From the curve we can see that at small processor loads of 10% there are no degradation in performance. As the load increases, performance is affected negatively.

The results obtained from the evaluation of processing power show that the virtualization and sharing of testbed nodes can negatively affect performance when simultaneous users are running computationally intensive task. Despite this, these conditions are not likely to occur because the USRP is the only high data rate full duplex SDR interface that requires high processing power. The RTL has a small sample rate and is a receiver only and the WiFi module is not computationally

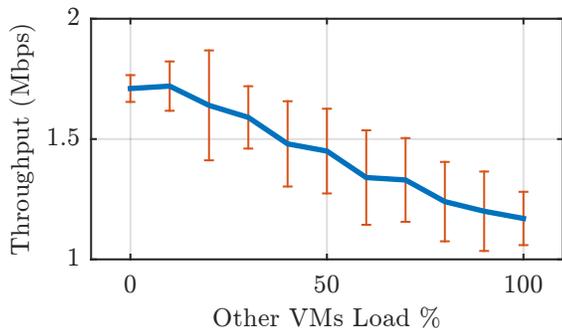


Figure 3: Effect of adding a processing load on the two other VMs on the throughput of the system shown in Figure 2 running in the third VM.

Table 1: Comparison between the specifications of the used laptop and the VMs of the simulation server.

Specification	Laptop	Sim. VM 1	Sim. VM 2	Sim. VM 3
Number of cores	4	2	4	8
CPU Speed (GHz)	2.5	2.6	2.6	2.6
RAM (GB)	3	4	8	32
Disk storage (GB)	512	512	512	512

ally intensive as most of its processing is done on chip. This makes the conditions that causes degradation in performance less likely to occur.

In later stages of the project, the virtual machines will be allowed to have configurable specs (like number of cores) according to the user’s requirements. This can allow users with high computational needs to have a machine with higher specs, thus alleviating the downsides of virtualization.

5.2 Simulation Server

To evaluate the benefit of the simulation server, we have performed a set of experiments on a personal laptop and on three different virtual machines (VMs) on the simulation server. The specifications of all machines are shown in Table 1. The comparison is performed in terms of time required to finish each experiment.

We start by describing the experiments setup. The simulation tool used on both platforms is NS2. The selected experiment was inspired from those used to evaluate the CellCheck system [8]. In this experiment, we have n nodes that are deployed uniformly in a square deployment area. One of these nodes is located at the center of the deployment area and works as a transmitter where, all other $n - 1$ nodes are receivers.

Figure 4 shows the time taken by all experiments replications on the server VMs and on the laptop. We can see that the server, using any of the tested instances, outperforms the laptop. The reason for this is the bigger size of main memory in the server VMs. Also, the

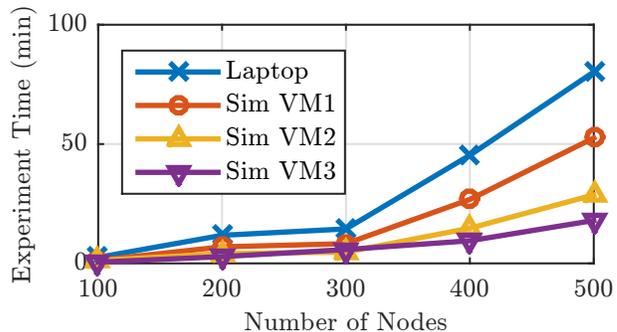


Figure 4: Effect of changing the experiment size on the elapsed time on both the simulation server virtual machines and the personal laptop.

the server processor has a higher clock frequency than the laptop. Additionally, the third VM of the server has twice as many cores as the laptop. All these factors make the simulation server performance superior than that of the laptop.

More tests showed that in some cases the laptop would fail to complete the simulation. When attempting the same experiment with 2000 nodes, the laptop failed to finish the simulation due to running out of memory.

5.3 Summary

In the evaluation section, the effect of node virtualization and the benefits of the simulation server were evaluated. From this section, we conclude that: (1) the existence of simultaneous users can affect the performance for all users if they were all running computationally extensive programs simultaneously. This is not likely to occur due the nature of the interfaces assigned to VMs. (2) The simulation server can, at least, reduce simulation time for users when compared to a normal laptop. For big scenarios, it works where a regular computer fails.

6. CONCLUSION

In this work, we presented CRC, a remotely accessible wireless testbed that is built to accelerate wireless research and enable collaboration. CRC introduces several new features over its predecessors. It supports having multiple users access its resources concurrently while guaranteeing isolation. Additionally, it virtualizes testbed nodes to improve utilization. It provides topology realization based on gain control. A scheduler based on genetic algorithm tries to maximize testbed utilization while realizing users desired topologies. Improved disk image saving enables users to restore all their nodes to a previous state with minimal storage usage. Code libraries are provided and are open for users contributions to enhance collaboration. CRC also targets improving education using an easy-to-use GUI and already-developed experiments for students. An evalu-

ation of the effect of the newly introduced node virtualization on performance was performed. The speed up achieved for simulation experiments using the powerful simulation server within CRC was demonstrated.

7. ACKNOWLEDGEMENT

This work is supported in part by a grant from the Egyptian National Telecommunication Regulatory Authority (NTRA).

8. ADDITIONAL AUTHORS

Moustafa Youssef, (Wireless Research Center, Egypt-Japan Uni. of Sci. and Tech., email: moustafa.youssef@ejust.edu.eg), Mustafa Y. ElNainay (Computer and Systems Eng. Dept, Alexandria University, Alexandria, Egypt., email: ymustafa@alexu.edu.eg), Amr A. El-Sherif (Wireless Intelligent Networks Center (WINC), Nile University, Giza, Egypt. Department of Electrical Engineering, Alexandria University, Alexandria, Egypt, email: aelsherif@nu.edu.eg) and Karim G. Seddik (Electronics and Communication Engineering Department, American University in Cairo, email: kseddik@aucegypt.edu).

9. REFERENCES

- [1] *RTL-SDR*. <http://www.rtl-sdr.com/about-rtl-sdr/>.
- [2] E. Altman and T. Jimenez. Ns simulator for beginners. *Synthesis Lectures on Communication Networks*, 5(1):1–184, 2012.
- [3] A. Asal, A. Mamdouh, A. Salama, M. Elgendy, M. Mokhtar, M. Elsayed, and M. Youssef. CRESCENT: a modular cost-efficient open-access testbed for cognitive radio networks routing protocols. page 179. ACM Press, 2013.
- [4] Y. A. Elnakieb, M. Azmy, and M. El-Nainay. Genetic algorithm-based mapper to support multiple concurrent users on wireless testbeds. In *Wireless Communications and Networking Conference*, pages 267–273. IEEE, 2016.
- [5] Ettus Research, <https://www.ettus.com/product/details/UB210-KIT>. *USRP B210 USB Software Defined Radio (SDR) - Ettus Research*.
- [6] GNU Radio Website, accessed July 2016.
- [7] M. Hibler, L. Stoller, J. Lepreau, R. Ricci, and C. Barb. Fast, scalable disk imaging with frisbee. In *USENIX Annual Technical Conference, General Track*, pages 283–296, 2003.
- [8] Y. Khazbak, M. Izz, T. ElBatt, A. Fahim, A. Guirguis, and M. Youssef. Cost-effective data transfer for mobile health care. *IEEE Systems Journal*, PP(99):1–12, 2016.
- [9] B. Lantz, B. Heller, and N. McKeown. A network in a laptop: rapid prototyping for software-defined networks. In *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, page 19. ACM, 2010.
- [10] T. R. Newman, A. He, J. Gaeddert, B. Hilburn, T. Bose, and J. H. Reed. Virginia tech cognitive radio network testbed and open source cognitive radio framework. In *Testbeds and Research Infrastructures for the Development of Networks & Communities and Workshops, 2009. TridentCom 2009. 5th International Conference on*, pages 1–3. IEEE, 2009.
- [11] The Network Simulator NS-2. <http://www.isi.edu/nsnam/ns/>.
- [12] K. Pechlivanidou, K. Katsalis, I. Igoumenos, D. Katsaros, T. Korakis, and L. Tassioulas. Nitos testbed: A cloud based wireless experimentation facility. In *Teletraffic Congress (ITC), 2014 26th International*, pages 1–6. IEEE, 2014.
- [13] R. Pike, D. Presotto, K. Thompson, H. Trickey, et al. Plan 9 from bell labs. In *Proceedings of the summer 1990 UKUUG Conference*, pages 1–9. London, UK, 1990.
- [14] T. Rakotoarivelo, M. Ott, G. Jourjon, and I. Seskar. OMF: a control and management framework for networking testbeds. *ACM SIGOPS Operating Systems Review*, 43(4):54–59, 2010.
- [15] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremono, R. Siracusa, H. Liu, and M. Singh. Overview of the orbit radio grid testbed for evaluation of next-generation wireless network protocols. In *Wireless Communications and Networking Conference*, volume 3, pages 1664–1669. IEEE, 2005.
- [16] A. Saeed, M. Ibrahim, K. A. Harras, and M. Youssef. A low-cost large-scale framework for cognitive radio routing protocols testing. In *Communications (ICC), 2013 IEEE International Conference on*, pages 2900–2904. IEEE, 2013.
- [17] Samer S. Hanna, Karim G. Seddik, and Amr A. El-Sherif. Topology Realization using Gain Control for Wireless Testbeds. In *Proceedings of the 10th international workshop on Wireless network testbeds, experimental evaluation & characterization*. ACM, 2016.
- [18] G. Smith, A. Chaturvedi, A. Mishra, and S. Banerjee. Wireless virtualization on commodity 802.11 hardware. In *Proceedings of the second ACM international workshop on Wireless network testbeds, experimental evaluation and characterization*, pages 75–82. ACM, 2007.
- [19] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar. An integrated experimental environment for distributed systems and networks. pages 255–270, Boston, MA, Dec. 2002.