

SIMULATED RADIOS FOR STRESS TESTING OF INET LINK MANAGER

Mariusz Fecko¹, Heechang Kim¹, Andrzej Cichocki¹, Larry Wong¹,
Mark Radke², Tom Young³, Thomas Grace⁴

¹Perspecta Labs,
Basking Ridge, NJ

²Bevilacqua Res. Corp.,
Edwards AFB, CA

³USAF AFMC,
Edwards AFB, CA

⁴NAVAIR,
Patuxent River, MD

ABSTRACT^{†‡}

IP-based telemetry systems such as iNET require extensive lab testing prior to fielding. Current iNET testbeds typically use several real radios and a mix of pre-recorded and live telemetry traffic. However, stress testing the iNET Radio Access Network (RAN) is hard to perform experimentally because of the limited number of available telemetry radios. To scale up the testing to a larger number of radios, we developed portable, low-cost VM-based telemetry radio simulators that interact with the iNET Link Manager (LM) and real radios to provide additional (emulated) links and queue depth reports. This approach makes it possible to establish the upper limit on the numbers of Test Articles that the LM can handle under various conditions. It also allows for fast reconfiguration of the number and set-up of simulated radios to test out specific use cases. The simulated radios free up real radios for off-site tests while preserving the ability for the continuous testing of LM features until real radios can be reclaimed. The developed testbed is portable owing to its lightweight set-up on the low-cost mac-mini computers. This makes it easy to use simulated radios also in off-site testbeds and field experiments.

I. INTRODUCTION

The iNET [1] Radio Access Network (RAN) segment provides a multiple-access telemetry network with RF links shared across geographically dispersed nodes. Within RAN, the iNET Link Manager (LM) [2] is responsible for allocating capacity [3] among Test Articles (TAs) and Ground Stations (GSs) to achieve QoS guarantees for multiple mission priority levels. Figure 1 shows a conceptual depiction of the LM system architecture. In the LM configuration, every QoS class is assigned a “class weight” based on DSCP, and every link is assigned “link priority weight” based on Mission Service Level Profile (MSLP) Weight/Priority. The LM instance at a ground node obtains the current per-mission/per-QoS class queue depths from Queue Managers (QMs) deployed at each airborne Test Article (TA) and each ground network node. Using these inputs, LM acts as a TDMA controller to allocate slots by assigning the RF channel capacity. It does so through generation of Transmission Opportunities (TxOps) messages that establish uplinks and downlinks and allocate transmission resources based on traffic and mission priority.

[†] This project is funded by the Test Resource Management Center (TRMC) Test & Evaluation/Sciences & Technology (T&E/S&T) Program through the U.S. Army Program Executive Office for Simulation, Training and Instrumentation (PEO STRI) under Contract No. W900KK-13-C-0027.

[‡] DISTRIBUTION STATEMENT A. Approved for public release; Distribution is unlimited. 412TW-PA-18338

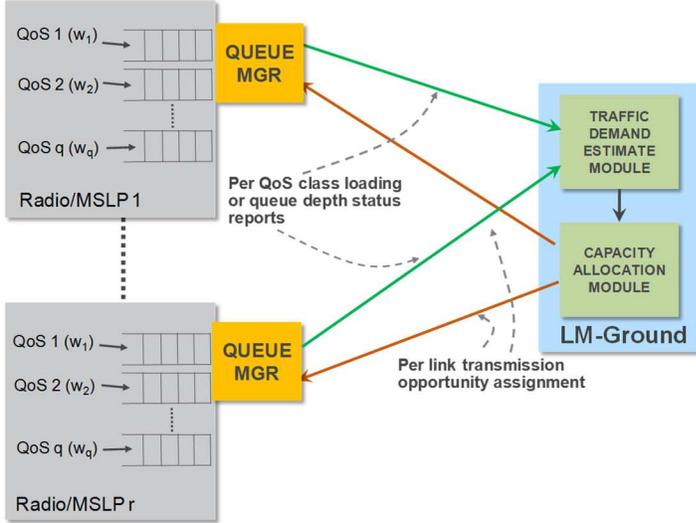


Figure 1: Conceptual LM system architecture

The LM functionality requires extensive lab testing prior to fielding. Current iNET testbeds typically use several real radios and a mix of pre-recorded and live telemetry traffic. However, stress testing the iNET Radio Access Network (RAN) is hard to perform experimentally because of the limited number of available telemetry radios. The concept of VM-based radio simulators is meant to scale up the testing to a larger number of nodes. The simulated radios interact with the LM by providing additional RF links and queue depth reports. The benefits of this approach are as follows:

- *Scalability*: Stress test the LM by establishing upper limit for the numbers of telemetry radios/queues that LM can handle.
- *Fast test reconfiguration*: Quickly change the number and set-up of simulated radios to test out specific use cases for LM.
- *Backup*: Free up real radios for off-site tests while preserving the ability to test new LM features using simulated radios until real radios can be reclaimed.
- *Portability*: Lightweight setup with mac-mini computers makes it easy to use simulated radios in off-site testbeds and field experiments.

The *eDLEP* (*enhancing Dynamic Link Exchange Protocol for iNET*) is a T&E/S&T program that supports the above objectives through the development of an experimental testbed in which an IP-based real Telemetry Radio is emulated with two pieces of software modules: Signaling Message Converter (SMC) and the custom Kernel. These modules emulate radio's TE (Traffic Engineering) queues and dynamic TDMA scheduling behavior. In the current testbed, they run on two separate virtual machines that are typically installed on the same virtual machine host. We migrated the simulated radios to a set of mac-mini computers located at the iNET system integrator's lab, where they are able to interface with the live LM. The latest testbed deployment contained two emulated ground radios and two emulated airborne radios. We also completed the system validation activities as part of eDLEP Phase 3 effort.

The rest of the paper is organized as follows. Section II describes the simulated radios architecture and design. Performance evaluation study is presented in Section III. Section IV describes future work and concludes the paper.

II. SIMULATED RADIOS ARCHITECTURE

The Signaling Message Converter (SMC) communicates with the LM (Link Manager) by exchanging Radio Frequency Link Management Messages (RFLMMs) related to TxOP assignment and queue management. It receives a per-link TxOp message from the LM for the emulated-source radio (the consumer of the TxOp assignment for transmitting packets over the

air) and translates it into per-queue TxOp assignments required by the fine-grained queue control in the custom Kernel [4]. The computed per-queue TxOp assignments are then sent to the custom Kernel of the same radio, which uses this information to schedule transmission of packets accrued in each TE queue.

The SMC also receives a Queue Status Report containing the current layer 2/3 queue sizes from the custom Kernel, converts it to an appropriate queue management message whose format the LM can parse, and sends the message to the LM. The contents of the message are used by the LM to calculate the next TxOP assignments. The SMC supports two types of the transport layer connection to the LM: UDP- and TCP-based. The TCP-based signaling protocol version was added to align the signaling with the latest iNET WG document [5] (Chapter 28, Section 28.4). The LM replaced UDP messages with TCP messages to communicate with TAs, introduced the concept of TxOp time-lease for increased robustness, and defined heartbeat messages to maintain an RF link in the absence of user traffic. We updated the simulated radios to use the above message semantics and modified the internal time bookkeeping accordingly.

A. TCP-based SMC–LM Signaling

Figure 2 illustrates the high-level software architecture of an emulated Telemetry radio. A TxOp assignment message received by the SMC from the LM contains a timeout value. When the SMC receives a new message, it updates its old per-link TxOp assignment and timeout values already stored in a global memory structure. The ‘epoch timer thread’ wakes up every epoch, converts per-link TxOp assignment to per-queue TxOp assignments, sends them to the custom kernel, retrieves the queue depth status from the custom kernel, and sends it to the LM. This message exchange takes place once every epoch.

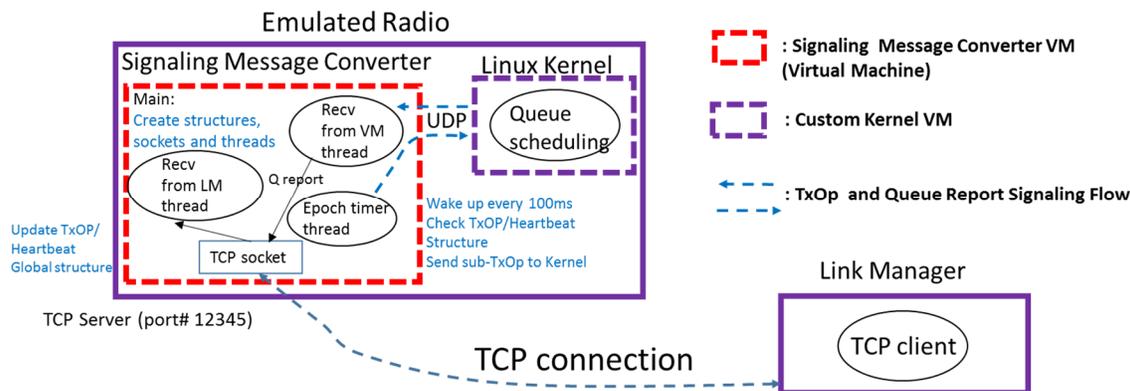


Figure 2: High-level software architecture of an emulated Telemetry radio and TCP-signaling between the emulated radio and LM

The SMC also processes a Heartbeat message received from the LM, whose timeout value controls all TxOp assignments for a given source radio. A Heartbeat timeout value is stored and updated in the global memory structure of the SMC and decremented every epoch. The SMC uses this value to stop TE queue scheduling in the custom kernel when the timeout occurs. We also implemented a secure TCP connection using OpenSSL Linux Library as required by the standard.

The custom Kernel emulates the TE (Traffic Engineering) queues of a Telemetry radio and their scheduling behaviors by using the Linux kernel HTB (Hierarchical Token Bucket) TC (traffic Control) mechanism. Our approach provides fine-grained queue control developed in earlier phases of the eDLEP program, where the single control knob (per-link capacity allocation) is

replaced with enforceable per-queue capacity allocations. This approach is able to drain the exact amount of traffic from each queue as specified by the LM per-queue capacity allocation [4]. A per-queue TxOp message contains information about when and how long each queue needs to be served in a given TDMA epoch for transmitting packets accrued in the queue. The custom Kernel is based on the 32-bit Linux Kernel version 3.2.0.

B. Testbed Configuration

Figure 3 shows the physical connectivity of the on-site testbed composed of four emulated radios and a Precision Time Protocol (PTP) server. The PTP server synchronizes the system times of the emulated radios and the LM. The testbed is also connected to the LM residing in the same LAN. We built virtual networks on top of the physically connected testbed LAN to emulate a dynamic TDMA Telemetry radio network (between TA radios and Ground radios) and a wired IP ground network (between Ground radios and the LM). The emulated networks are realized by configuring network interfaces and IP route settings of each virtual machine. Since an emulated radio is composed of VMs and emulated networks are created virtually by configuring VMs' interfaces without changing Apple Mac mini's physical network interface configuration, the test-bed can easily be extended to a larger network with more emulated radios. This can be done by connecting additional Apple Mac minis to the same LAN, installing VMs on each Apple Mac mini that emulates a Telemetry radio, and configuring VMs' network interfaces for creating emulated telemetry networks. An example network configuration and virtual network settings are described in the later part of this section.

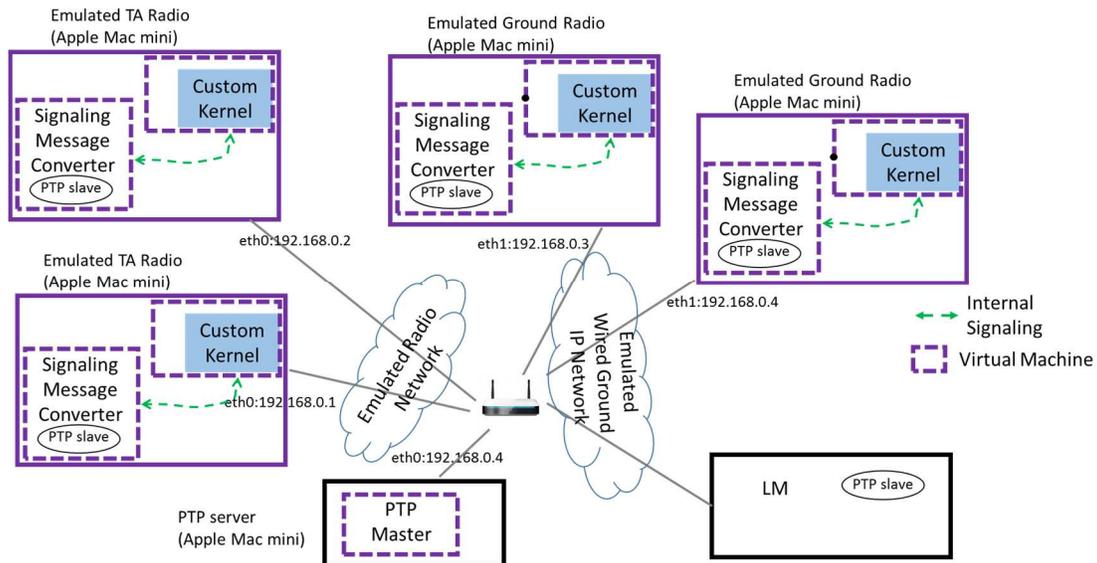


Figure 3: Testbed network and the LM

Figure 4 illustrates the configuration of an example testbed consisting of four emulated radios (two TAs and two Ground Stations) and the LM. In this example, TA1 radio communicates with Ground1 radio while TA2 radio communicates with Ground2 to send and receive user traffic data. Virtual networks are set up to create emulated networks by configuring IP addresses and route settings in each VM. The 10.1.3.x virtual LAN was set up for the Emulated Ground IP network between Ground radios and the LM. The 10.1.1.x virtual LAN and 10.1.2.x virtual LAN emulate the Telemetry radio network between TAs and Ground Stations. The 10.1.3.x virtual LAN is created for PTP time synchronization among radios and LM.

Emulated radio’s internal VM architecture and its network interface information are hidden to the LM, which needs to know only one IP address (IP address of eth0 interface of the SMC) per emulated radio for sending signaling messages. These are 10.1.1.1, 10.1.2.1, 10.1.3.1 and 10.1.3.2 for TA1, TA2, Ground1 and Ground2 radios, respectively. Since a Ground radio works as a gateway router between the LM and a TA radio for exchanging signaling messages (i.e., TxOps and queue depth reports), proper IP route settings are required in both the LM and the VMs.

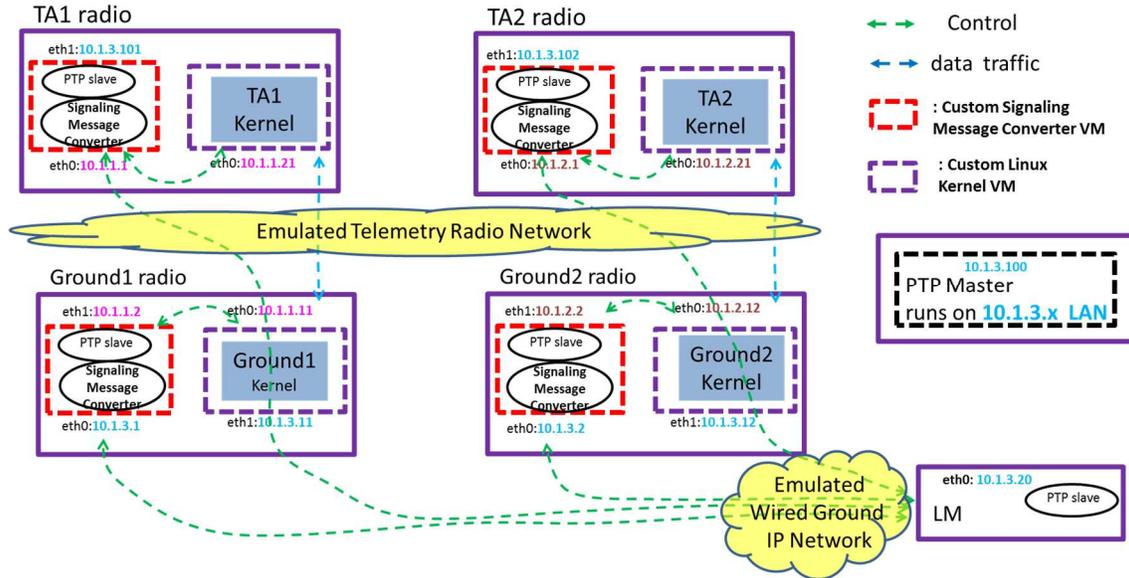


Figure 4: Example test-bed architecture and setup

The LM needs to be configured such that the default gateway IP addresses can be 10.1.3.1 and 10.1.3.2 for 10.1.1.1 (the IP address of TA1) and 10.1.2.1 (the IP address of TA2), respectively. TE queues are implemented in the custom Kernel VM. This requires the SMC of a Ground radio to forward TxOps messages for TAs to the custom Kernel VM of the same Ground radio. Accordingly, in the SMC VM, the default gateway IP address for the destination TA should be the IP address of the custom Kernel VM’s eth1 interface connected to the ‘Emulated Wired Ground IP network’. The custom Kernel VM then forwards TxOps to the other interface (eth0) of the same kernel. Eventually the TxOps messages are delivered to a TA via the ‘Emulated Telemetry network’ when the Ground radio gets a chance to transmit the messages.

Simulated radios communicate with other simulated radios and the LM (via a wired connection between host machines and/or within the host machine as VMs), while real radios communicate with other real radios via RF and the LM. This allows for real radios and radio simulators to coexist on the same network and for the LM to interact with both. We established that real radios, simulated radios, and the LM all inter-operate at the signaling level. We also verified the correctness of the signaling message flows among the SMC, the LM, and the custom Kernel.

III. PERFORMANCE VALIDATION

We validated that simulated radios approach the performance of real radios up to certain limits. We have successfully tested sample traffic scenarios using a combination of MGEN traffic, VoIP applications, and FTP file transfer, and confirmed that simulated radios contend for channel bandwidth, at both up and down links. We demonstrated that applications such as VoIP and FTP work correctly with both the UDP-based and the TCP-based LM software, i.e., UDP with absolute

time TxOps and TCP with relative time TxOps. In particular, we used a VoIP softphone and a SIP server (both installed at the TA) and a softphone (installed at the ground station). We verified the ability to make a phone call from TA to ground while sending ftp from TA to the ground. We made sure that voice gets the BW even when BE ftp TCP traffic “fills the pipe.”

A. MGEN Traffic Scenarios

Our initial approach was to verify performance using MGEN traffic that approximates FTP file transfer and voice flows. This section provides a summary A of the obtained experimental results.

In the first test scenario, we loaded 2Mbps UDP traffic on each link as shown in Figure 5. In this scenario traffic is properly marked so that packets are in queue Q3. Queue weights are distributed evenly among all 8 queues. In all three MGEN traffic scenarios, we used 8 Mbps as a total telemetry radio raw bandwidth.

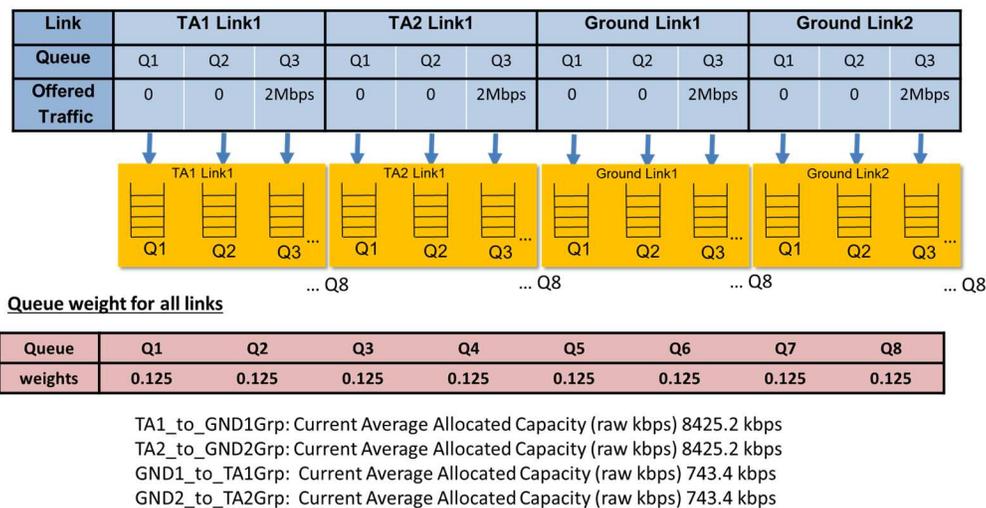


Figure 5: Test 1 traffic loading

The top left plot in Figure 6 shows queue depths of four links. The queues in both ground radios are saturated (the max queue size is about 100 Kbyte) while the queues in both TA radios are not. This is because TxOp assignments for both up links are constrained to a pre-configured number by the LM. We could not verify the uplink TxOp assignment threshold value, but this value is extrapolated from the middle left and bottom left plots that present TxOp1 assignment and TxOp2 assignment of the four links. For example, the LM allocated about 2 msec per TxOp for the uplink 1 (Ground1 to TA1).

With two TxOp assignment per 100 msec epoch, about 320 kbps raw radio bandwidth is expected to be allocated to Ground1 radio. Considering overheads in each protocol layer, the actual application throughput will be smaller than this number. The experimental result shows 190 kbps application throughput for the Ground1 up link traffic as illustrated in the right plot of the figure. In a typical telemetry network configuration, uplink TxOp assignments are configured to be much smaller than downlink TxOp assignments. This is because a ground radio normally sends only signaling messages while a TA radio sends user data traffic as well as signaling messages. The LM allocated more than 20 msec per TxOp for both down links. This allocation is large enough to serve 2 Mbps offered traffic and the resulting throughputs of both links are about 2 Mbps, as shown in the right plot of Figure 6.

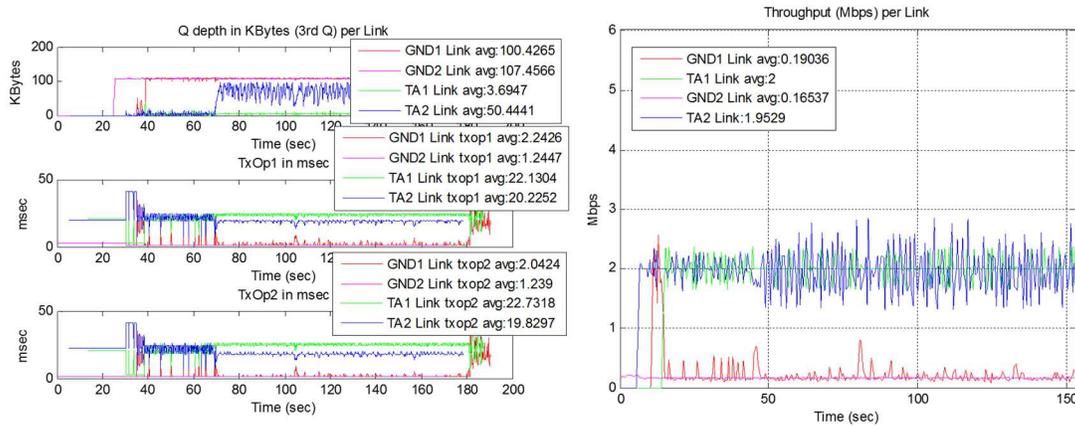


Figure 6: Test 1 results

Test 2 traffic scenario is a case of an overloaded traffic condition where TA1 TxOp assignments cannot handle the total offered load (80Kbps + 4Mbps) of TA1. From this traffic load experiment, expected results are as follows: Throughputs for down link 2 (TA2 to Ground2) and all up links are the same as Test 1. Q3 of the down link1 (TA1 to Ground1) will get 80Kbps throughput with no loss while Q4 will get less than 4Mbps throughput with some loss. If 4Mbps in Q4 does not induce loss, we continued increasing the loading in Q4 until loss occurred. Actual results in Figure 8 match the expected performance.

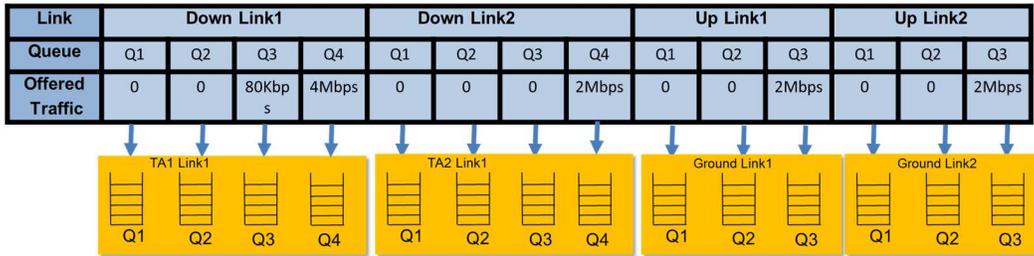


Figure 7: Test 2 traffic loading

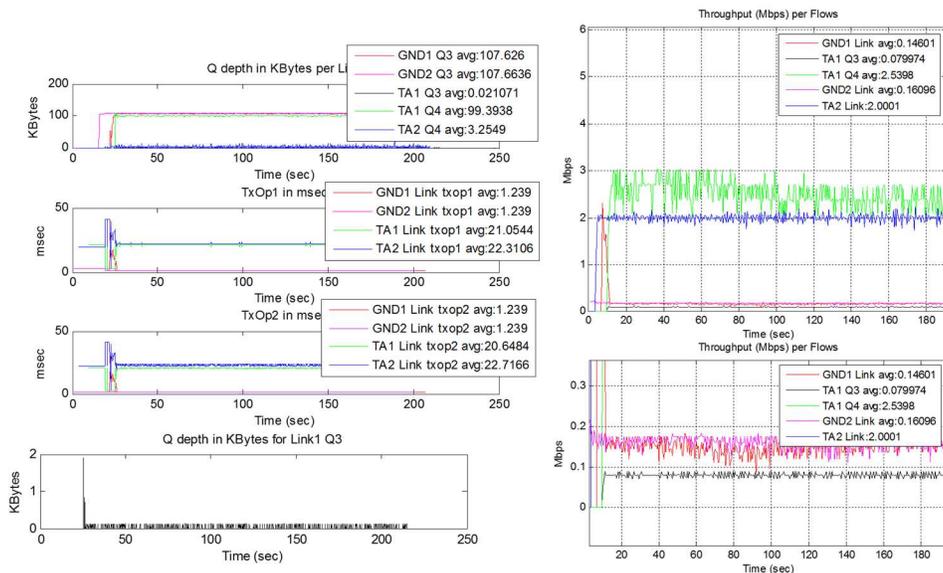


Figure 8: Test 2 results

B. VoIP/FTP Traffic Scenario

The VoIP/FTP traffic scenario was created to match closely the test environment at the iNET integrator's lab. We use the following queue configuration and traffic flows.

- Radio IP layer rate equal to 9 Mbps, TE queue size depending on policy (for VoIP EF, queue size meeting 100 ms E2E latency; for Best Effort FTP, default Linux configuration);
- We sent 80Kbps voice over IP traffic stream marked as Expedited Forwarding, and a ftp traffic stream marked as best effort from TA to ground;
- VoIP softphone and a SIP server installed at the TA and a softphone installed at the ground station. We verified the ability to make a phone call from the TA to ground while sending ftp from TA to the ground at the same time. We made sure that voice gets the BW even when BE ftp TCP traffic “fills the pipe”.

We have installed the SIP server and a softphone on two kernels within apple-19 (Ground radio) and configured them so that a call could be made and answered automatically, playing a 100kB sound file. We confirmed that the generated traffic is a stream of 50 UDP packets per second, each of length 172 bytes, for the total bandwidth consumption of about 69kbps. We have run tests using Linphone VoIP application. Figure 9 shows traffic loading in which Q3 is used for VoIP, Q4 carries TCP (FTP), and Q5 carries UDP (generated by MGEN).

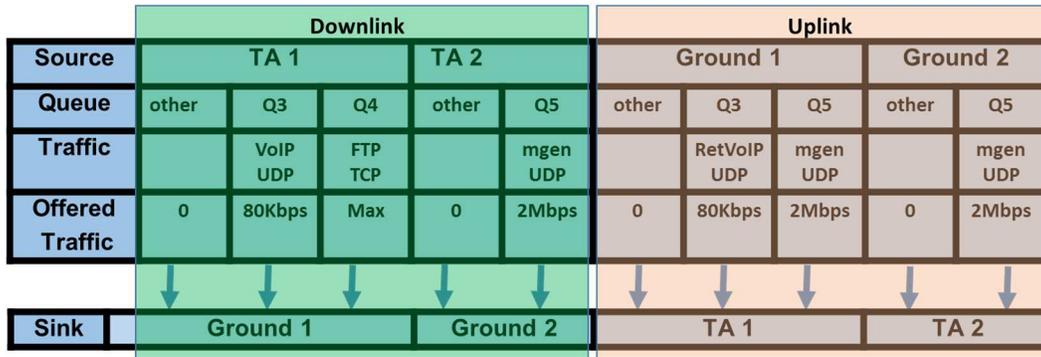


Figure 9: Traffic loading of VoIP/FTP scenario

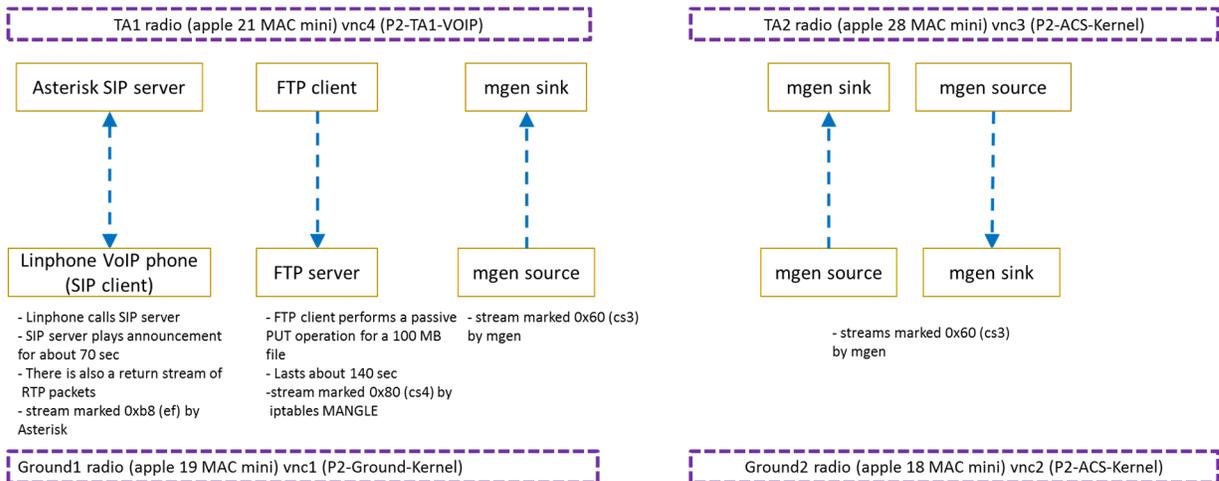


Figure 10: Network configuration for VoIP experiments

Figure 10 shows the current network configuration for VoIP experiments, where the Linphone calls the SIP server, which plays announcement for about 70 sec. There is also a return stream of RTP packets (disregarded). Voice stream is marked 0xb8 by Asterisk. FTP client performs a passive PUT operation for a 100 MB file, which lasts about 140 sec. FTP stream is marked 0xe0 by iptables MANGLE to make the associated packets go into an appropriate queue.

The performance of UDP MGEN flows in Q5 is shown in Figure 11. The results are consistent with the LM capacity allocations and traffic demands. This means that the emulated radios correctly share the allocated capacity among each other and among individual queues.

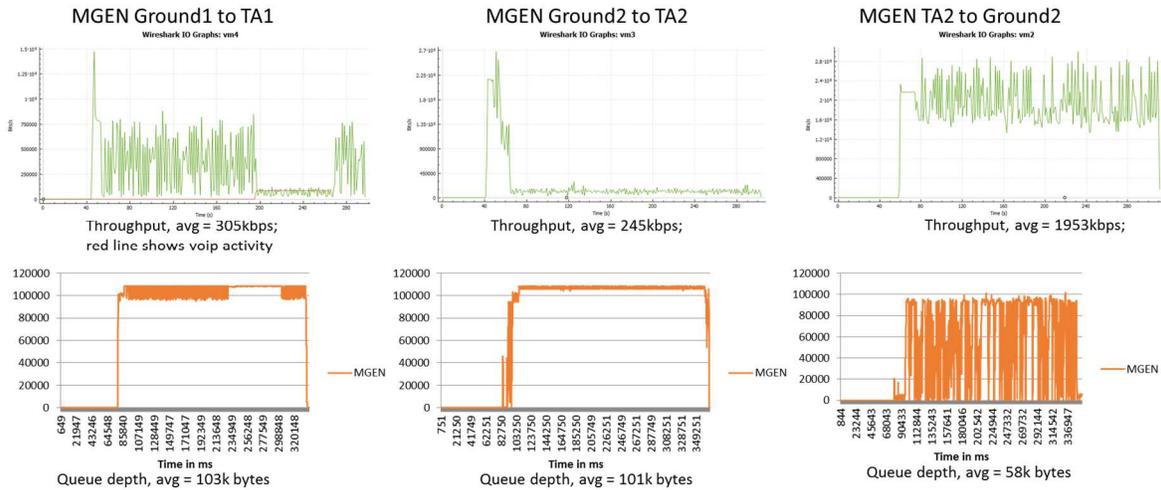


Figure 11: Performance results of MGEN traffic

Performance results shown in Figure 12 confirm that the VoIP traffic is protected in the presence of FTP traffic, experiencing no loss and small jitter.

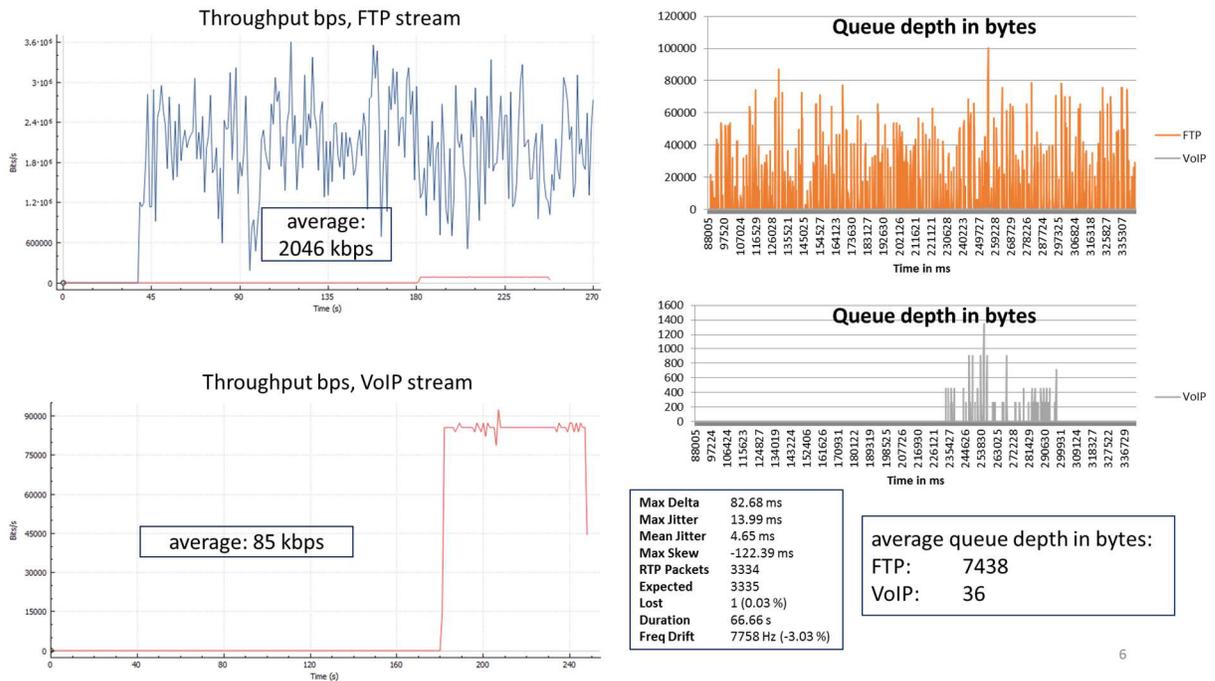


Figure 12: VoIP and FTP performance results

IV. CONCLUSION AND FUTURE WORK

To enable stress testing of the iNET Link Manager, we developed portable, low-cost VM-based telemetry radio simulators that interact with the LM by emulating additional RF links and queue depth reports that trigger capacity allocation by the LM. The iNET system integrator has performed an initial assessment of the radio-simulator VMs deployed on the integrator's main servers. The assessment confirmed that the LM is able to interface with both simulated and real radios in a unified testbed in the control and data planes. A number of future modifications and enhancements also were identified to maximize the effectiveness of the approach:

- The simulated radio implementation will be brought up to the latest iNET standard IRIG 106-17 Chapter 24, which includes new messages (TxOp ID Acknowledgement Report, Link Metric Report, Link Transmit Statistics Report) and up to 100 capacity allocations per epoch.
- New capabilities will be implemented with respect to how the simulated radio testbed is integrated with the physical radio network. For example, IP forwarding rules and routes will be set in order to receive traffic on one interface (e.g., wired) and forward out of another (e.g., simulated RF network). A new configuration option will allow TLS support to be easily enabled/disabled.
- Support for IP multicast will be added to the simulated RF network. This can be realized by emulating the RF MAC header processing, e.g., use an IP multicast protocol to send telemetry data to multiple simulated GS radios and derive the multicast group ID from TxOp Center Frequency field.

We will also increase the scalability and flexibility of the current simulated radio setup by deploying more mac-mini computers and by rewiring the connectivity among various VMs. In particular, a new testbed configuration will provide physical network interface separation between a radio simulator's "wired network interface" and its "RF network interface". Such a topology is easily scalable when desiring to add new radio simulators into the testbed. The "RF network interface" of each radio simulators will connect to a switch representing the Simulated RF Network. The "wired network interface" may connect to the ground network or it may connect to a new subnet representing a new TA.ⁱ

REFERENCES

- [1] B. Abbott, M. Araujo, M. Moodie, T. Newton, T. Grace. iNET System Design Concepts. In *Proc. ITC/USA International Telemetry Conference (ITC)*, 2011.
- [2] R. O'Connell. Telemetry Network System (TmNS) Link Management Algorithm Verification. In *Proc. ITC/USA International Telemetry Conference (ITC)*, 2013.
- [3] M.A. Fecko, K. Chang, M. Barton, H. Kim, S. Gadgil, M. Sarraf, L. Wong, S. Samtani, R. O'Connell, B. O'Neil, M. Rauf, M. Radke, T. Young, T. Grace. Dynamic Capacity Allocation Algorithms for iNET Link Manager. In *Proc. ITC/USA International Telemetry Conference (ITC)*, 2014.
- [4] M.A. Fecko, K. Chang, A. Cichocki, L. Wong, S. Samtani, R. O'Connell, M. Radke, T. Young, T. Grace. Priority-aligned flow control for IP-based telemetry systems. In *Proc. ITC/USA*, 2015.
- [5] integrated Network Enhanced Telemetry (iNET) Radio Access Network Standards Working Group IRIG 106 Part 2 Chapter 28, Radio Access Network Management Draft, January 2016.

ⁱ Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Test Resource Management Center (TRMC) Test and Evaluation/Science & Technology (T&E/S&T) Program and/or the U.S. Army Program Executive Office for Simulation, Training and Instrumentation (PEO STRI) under Contract W900KK-13-C-0027.