

Automated Physical Testbeds for Emulation of Wireless Networks

Axel Sikora, Jubin Sebastian E, Artem Yushev, Edgar Schmitt and Manuel Schappacher

Institute of Reliable Embedded Systems and Communication Electronics (ivESK), Offenburg University of Applied Sciences, 77652 Offenburg, Germany

Abstract. Institute of Reliable Embedded Systems and Communication Electronics, Offenburg University of Applied Sciences, Germany has developed an automated testing environment, Automated Physical TestBeds (APTb), for analyzing the performance of wireless systems and its supporting protocols. Wireless physical networking nodes can connect to this APTb and the antenna output of this attaches with the RF waveguides. To model the RF environment this RF waveguides then establish wired connection among RF elements like splitters, attenuators and switches. In such kind of set up it's well possible to vary the path characteristics by altering the attenuators and switches. The major advantage of using APTb is the possibility of isolated, well controlled, repeatable test environment in various conditions to run statistical analysis and even to execute regression tests. This paper provides an overview of the design and implementation of APTb, demonstrates its ability to automate test cases, and its efficiency.

1 Introduction

Over the last decade, wireless networks have been deployed at almost all the areas. The exciting capabilities made possible by the inexpensive ubiquity of wireless networking technology, has led to a massive amount of research activity to improve the performance of existing wireless networks, and develop new applications based on this technology such as wireless sensor networks and internet of things. Even though many wireless standards are available, to meet the specific requirements in terms of low energy, scalability etc, we need to develop optimum solutions. Conducting such type of wireless network research is, none the less, a challenging effort due to the distributed and short lived nature of wireless signal propagation. There are many challenges in wireless network research which are nicely listed in [1], [18]. The major approaches used for performance evaluation of wireless networks are simulation, virtualization, outdoor field test (test beds), and emulation.

Simulation is used to test the behaviour of the protocol and its implementation, i.e. the firmware on hardware abstraction layer level. The advantage of simulations is its scalability, but simulation is not helpful in debugging the underlying hardware and is totally missing the effects that hardware has on RF characteristics [1].

Virtualization of networks also used to analyse the protocol stack in a virtual environment of intended network. Network virtualization has the advantage that enables to emulate connection between applications, services, dependencies, and end users in a test environment without the need to test in real hardware. But, distinctive properties of wireless environment such

as time variant channels, mobility, broadcast, attenuation, and specific access technologies makes convergence, sharing and abstraction difficult to achieve [10].

Outdoor field tests play a vital role for testing the capabilities and reliabilities of wireless systems, but the major disadvantages are irreproducible tests due to the uncontrollable nature of the outdoor environment. Also, outdoor field tests are time consuming and therefore expensive [1].

Emulation is the combination of simulation and testbed. As in the simulation approach emulators can generate predefined emulated network conditions and traffic dynamics for different use cases. Hardware-based emulation clearly achieves the most physical layer realism; practical considerations such as ease of development, control, and experimental repeatability have made emulation the dominant experimental technique [1]. The requirements of the development of an emulator can be seen in [2].

Authors have developed first generation of our automated RF emulator for distributed wireless sensor networks for analysing EnOcean Radio Protocol (ERP) [2], and the second generation developed to test Ko-TAG subsystem [1]. In this paper we describe the architecture and implementation details our third generation fully automated emulator for wireless networks. This includes the physical networking nodes, RF elements to allow the controlling of RF environment between the participating nodes, and automated control software. We call this set up as Automated Physical Testbeds (APTb) for emulation of wireless networks.

APTb is an emulator to perform reproducible tests with static and dynamic attenuation between the

participating wireless nodes as real laboratory set up. The nodes are connected over their antenna outputs to RF-waveguides. It is also possible to cut or switch the connections of the network, in this way many different topologies can be set up. APTB promises to achieve much of the flexibility of wireless emulators while maintaining great realism of real wireless networks [4].

This contribution is structured as follows: Ch. 2 gives a review on the previously developed emulator platforms by other researchers and the previous generation of emulator by author's Ch. 3 presents the APTB architecture and implementation details, whereas Ch. 4 gives an overview of experience with APTB, Ch. 5 concludes the paper.

2 State of the art

An overview of the previously proposed emulators by other work groups and authors team is given in this chapter.

2.1 Review of emulators

In the field of wireless network emulation we observed different approaches like a). Central control based wireless emulators, in which the wireless nodes connect to central server as seen in ONE [5], MIT's Click modular router [6], Dummynet [7] and NIST NET [8]. b). Simulation integrated wireless emulators, which combines the network emulation with existing network simulators which has rich models and libraries as VINT/ns [9]. c). Trace based wireless emulation, which collect the traces from the target network and then construct a wireless network model with collected traces, and then reproduces the traced network effect in a wired network and provide a reproducible environment [3].

Clearly these emulators provide a controllable and reproducible environment, but it generally lacks the support for network topology, mobility and an interactive interface for users. In some of the emulator setup the node mobility implemented using remotely controlled robots as in CMU [11], APE [12], Mobile Emulab [13], m-MiNT [14]. Associated with the mobility features a tracking system [14]-[16] is used to determine the position and orientation of each node. But such type of emulator platforms is not flexible and efficient to test the wireless system in the development stage. Using a wired connection of RF elements we overcome these drawbacks. EMWIN [3] is an IP based mobile wireless network emulation system; provide a unique feature of emulating multiple mobile hosts in one single emulator node, also to emulate the mobility using a wired network of computers.

To the best of our knowledge, we have not seen a system which uses wired connection of RF elements like attenuators, splitters, switches etc., to imitate the RF environment and manipulate the path characteristics. So, we decided to develop such type of emulator for wireless networks. Next section reviews the details of our previous generation of emulators.

2.2 Review of our previous emulators

The first generation consists of elements like network nodes, management nodes, a client computer, a deployment support network (DSN), with an embedded web server running web 2.0 technology for remote monitoring and control of its main elements. The emulator originated from the implementation of the EnOcean Radio protocol (ERP), the details of this emulator and experience are described in [2]. This presented an approach of management and sniffing tools for distributed wireless networks. The emulator was designed flexible and generic to extent this to other wireless protocols.

The second generation of our emulator developed for highly scalable IEEE 802.11p communication and localization subsystem in the Ko-FAS project [1]. This includes physical networking nodes, but models the RF environment using RF-waveguides. The RF emulator allows the controlling of path losses and connectivity between any of the nodes with the help of RF attenuator and programmable RF switches, while it's shielded against its surrounding RF environments in the lab. Details and experience of the system design can be seen [1].

These two versions of our emulator was designed for specific projects, we extend this development to easily reuse it for other wireless network research projects.

3 Automated physical testbeds (APTB)

We have developed a wireless emulator, APTB that enables both realistic and repeatable wireless experimentation by supporting accurate wireless signal transmission, propagation, and reception in an emulated physical space. APTB uses physical networking nodes, but where the RF environment is still modelled using RF-waveguides and RF elements like attenuators, splitters, switches, cables, terminators etc. In this environment, it is well possible e.g. to control the path loss between any of the nodes with the help of RF attenuators and to control the connectivity between any of the nodes with the help of programmable RF switches. APTB maintains the repeatability, configurability, the ability to modify wireless device behaviour, automated tests, support for a large number of nodes, manageability of simulation while retaining the support for real applications, and much of the realism of hardware testbeds. As a result, this APTB emulator can be a superior platform for wireless network experimentation [4].

3.1 Architecture

APTB consists of both custom and commodity hardware that must act in concert to enable users to accurately emulate arbitrary signal propagation environments and efficiently execute experiments. Developing hardware to achieve this has been a challenging task. Figure 1 describes the architecture that enables our physical layer APTB to achieve accurate emulation while enabling

efficient experimentation. The hardware elements of the APTB are APTB nodes, RF elements and Controller units.

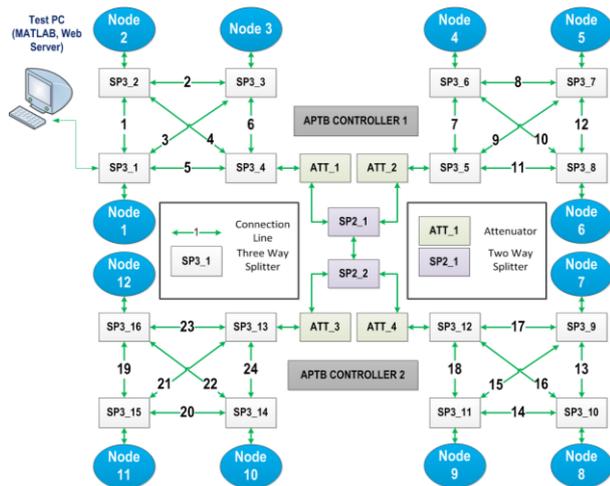


Figure 1. Architecture of APTB.

Any of the wireless modules with an antenna output used as APTB nodes. We encapsulate these APTB nodes in to shield boxes. The RF nodes are shielded from each other using RF shield boxes so that less communication occurs over air. Such types of 12 shield boxes are present in the APTB emulator as shown Figure 1. The RF elements used in the APTB are two way splitters/combiners (SP2), three way splitters/ combiners (SP3), variable and fixed attenuators (ATT), high isolation switches (SW), SMA cables, terminators and RF shielded test enclosure. All the communication between RF nodes occurs through the signal propagation environment modelled using these different RF elements within the APTB emulator. The controller units present in the APTB are two ARM Cortex-M3 based MCU to control the RF elements.

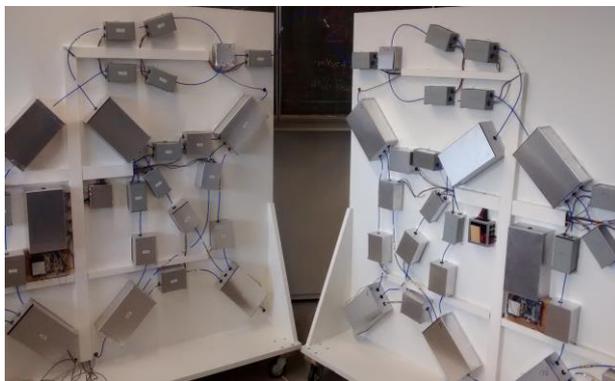


Figure 2. Rear side of APTB.



Figure 3. Front side of APTB.

In the author's Lab APTB is developed in two identical physical boards, the rear side of the testbed is shown in Figure 2 with splitters, attenuators, switches, SMA cables and APTB control unit. The front side of the APTB is shown in Figure 3 consist of shield boxes where we can encapsulate the APTB nodes.

3.2 APTB hardware

The hardware elements of the APTB are APTB nodes, RF elements and controller units.

- APTB Nodes

Any type of wireless nodes with antenna output can be used with APTB. For example, @ANY900-2 modules used as APTB nodes, which contains Atmel microcontroller ATmega 1281V (128 bytes of FLASH memory, 8K bytes SRAM, 4K bytes EEPROM), IEEE 802.15.4 radio transceiver AT86RF212 (868/915 MHz band) and 2Mbit serial data flash (AT25F2048), which can store the flash images of the used MCU [19]. We encapsulate these APTB nodes in to shield boxes. The RF nodes are shielded from each other so that no communication occurs over air. Such types of 12 shield boxes are present in the APTB emulator.

- RF elements

The RF elements used in the APTB are provided by "Mini Circuits" company follow: two way splitters/combiners (ZFRSC-183+), three way splitters/combiners (ZB3PD-63+), variable attenuators (ZX76-15R5-SP+), fixed attenuators (VAT-30+), high isolation switches (ZASWA-2-50DR+), SMA cables (141-6SM+), RF shield boxes (STE2200) and terminators (ANNE-50+). All the communication between physical nodes occurs through the signal propagation environment modelled using these different RF elements within the APTB emulator.

- Two way Splitters/combiners (ZFRSC-183+): are two-way, 0 degree, resistive (50Ω) splitter/combiner, which operates in wideband from DC to 18000MHz, and has coaxial connector. This provides low insertion loss and high isolation [20].

- Three way splitters/combiners (ZB3PD-63+): are three-way, 0 degree, in-phase splitter and combiner covers a wide frequency range (150-6000 MHz), making this splitter suitable for GPS, GSM applications, in addition to WiFi, Bluetooth and 802.11a uses. This model also features low insertion loss, amplitude and phase unbalance [21].

- Fixed attenuator (VAT-30+): is coaxial SMA fixed 30 dB attenuator of 0.5W, and has wide coverage from DC to 6000 MHz [22].

- Variable attenuators (ZX76-15R5-SP+): is a 50Ω RF digital step attenuator that offers an attenuation range up to 15.5 dB in 0.5 dB steps. The control is a 5-bit serial interface. The model operates on a single +3 volt supply [23].

- High isolation switches (ZASWA-2-50DR+): is 50Ω SPDT, coaxial high isolation switch, operate at DC to 5GHz, and has an integral TTL driver. It possesses high isolation of 82dB typical at 2 GHz [24].

- SMA cables (141-6SM+): low loss precision test, 141 series hand-flex coaxial cables are used for interconnection RF elements. It has wideband frequency

coverage from DC to 18GHz and low Loss, 0.43dB, at 18GHz. Also it has excellent return loss, 23 dB at 18GHz [25].

➤ RF shield boxes (STE2200): is an RF shielded test enclosure [26] used to connect the APTB nodes and have a shield RF environment for test set up. In STE2200 heavy, rugged 0.090 and 0.125 aluminium is used throughout and double lip woven Monel flexible gaskets are used at all joint locations assuring a reliable RF tight closure. Oversized hinges and latches are used to provide a physically tight seal every time. RF absorbent foam lines the interior to provide a typical RF attenuation of -90dB @ 3GHz.

➤ Terminators (ANNE-50+): SMA Male, 50Ω terminator [27] used for test set up of APTB.

Table 1. RF elements overview.

type	attenuation [dB]	frequency	No. used in APTB
two way splitter/combiner	6	DC-18000MHz	2
three way splitter/combiner	5	150-6000MHz	16
fixed attenuator	30	DC-6000MHz	12
variable attenuator	1-15	DC-4000MHz	22
high isolation switch	82	DC-5GHz	24
terminator	-	DC-18GHz	12
RF shielded test enclosure	90	-	12

All the communication between RF nodes occurs through the signal propagation environment modelled using these different RF elements within the APTB emulator.

• **Controller Units**

APTB Controller unit is an ARM Cortex-M3 based MCU (STM32F107VCT6) running together with a web interface on the PC to control paths and attenuation between different nodes. With these we can vary the topology and attenuation between the APTB nodes.

Based on the experience from earlier emulator setups [2] [1], all elements have additional shielding and are carefully mounted whereas links were fixed with a torque wrench. All elements and connections are characterized after they have been built-in with regard to attenuation, reflection, and phase shift (delay).

3.3 APTB software

The main software implementations for APTB are control software for RF elements and MATLAB scripts to automate, reproduce the tests. All the involved devices like APTB nodes, RF elements and controller units are automatically controlled by MATLAB program from the PC, integration with the APTB controller unit.

To control the RF elements of the APTB emulator uses web interface running together with APTB controller. In experiment setup the APTB consists of two parts, which helps on one side to control data traffic flow

better and on other side to extend physical topology as much as it required for the test case. Two identical parts create a local network which is managed externally by an Ethernet based control interface. Used infrastructure with such devices helps to add different controlling extensions which may automatically perform tests and grab output data [1].The controlling devices are equipped with an embedded TCP/IP stack including an embedded webserver. This provides basic access mechanisms e.g. using an embedded website as shown in Figure 4. In APTB Low layer commands are used to directly access RF elements of the APTB. For example,

- APTB:RADElements:ATTenuator:SETAttenuation <id>,<val>
- APTB:RADElements:SWitch:SETEnable <id>,<val>
- APTB:NETements:NOde:SENDDData <id>,<data>

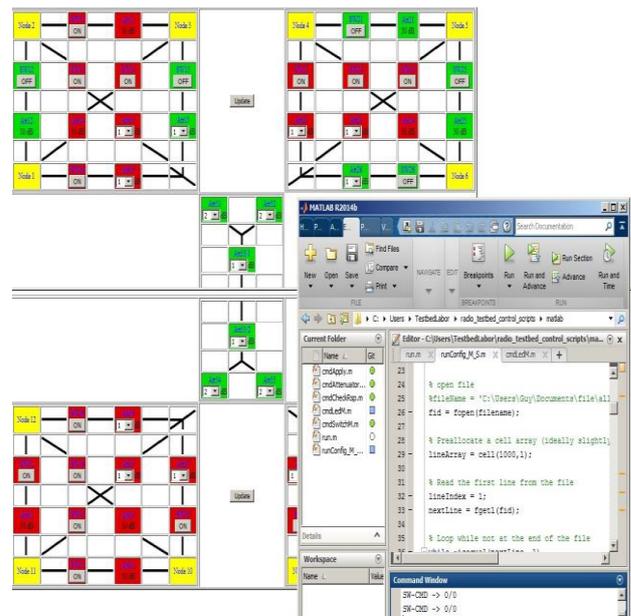


Figure 4. Screenshots for the programming and the web interface for APTB automated tests

These commands are interpreted and executed by the APTB Controller which includes an according “Standard Commands for Programmable Instruments” (SCPI) parser/handler. The RF Switch Control is fitted with the STZEDN’s embedded TCP/IP protocol suite (emBetter) which is expanded by an SCPI module.

Furthermore the software of the controlling devices have been extended with a software module able to interpret the SCPI protocol commonly used to control and to monitor measurement devices. This makes it possible to control and to automate the network topology e.g. with MATLAB that is using Java-IO libraries to send the SCPI commands to the controlling devices. Self-defined configuration scripts can be used to create a time-controlled sequence of different topologies and therefore also provides the possibility to reproduce scenarios. MATLAB programs are used for high level user interface to describe the use cases and topologies. A sample MATLAB script for a scenario of link loss add reconnection after a specific amount of time can be

described in a human readable manner (CSV, XML) as in the Table 2.

Table 2. MATLAB script for test setup.

0 ,SW,12,1<LF>	Enable Switch-ID 12
100,SW,12,0<LF>	Disable after 100ms
200,SW,12,1<LF>	Reconnect after another 100ms

Figure 4 shows the automated test setup together with webservice using MATLAB, the green colour between the nodes shows the enabled paths and red shows the disabled path.

4 Experiences with APTB

To analyse the accuracy of the emulator we have performed many tests like creating different topologies using automatic control of the attenuator and switches using MATLAB, and we have measured the attenuation between different nodes, compared this with the calculated attenuation. These measurement results confirmed the calibration accuracy of the APTB. We have extensively used APTB for testing and verification of the protocol stacks developed by author's team. These testing involve various steps: test case definition, configuration of APTB nodes, test execution and result analysis. The measurements of different test cases were performed for different topologies like chain, mesh and tree. Each test executed many times in a time span of 24 hours to one week. More details related to these experiments and results are seen in [4], [17].

5 Conclusion and outlook

In this paper we presented the state of art of our wireless network emulator APTB, which helps the wireless network researchers to set up different wireless time varying topologies and path characteristics automatically. The major advantage of using APTB against traditional mount and measure approach is concentrated in striving for isolated, well controlled, repeatable test environment which may help to prove a sustainable and reliable behaviour of wireless network protocols in various conditions. The authors would be happy to share more details about the implemented APTB, so that other work groups can build identical or extended testbeds, keeping compatibility especially on the software side.

References

1. A. Sikora, M. Schappacher, L. Möllendorf, "Automated RF Emulator for a highly scalable IEEE802.11p communication and localization subsystem," in *6th Intl Workshop on Communication Technologies for Vehicles*, Offenburg, (2014)
2. L. Möllendorf, D. Schauenberg, N. Braun, D. Rahusen, A. Sikora, "A Distributed Embedded Web2.0 Based Automated Testbed for Wireless Mesh Networks," in *1st IEEE Int'l Workshop On AWN*, Leipzig, (2009)

3. Ni, Pei Zheng and Lionel M., "EMWIN: Emulating a Mobile Wireless Network using a Wired Network," in *In Proceedings of the 5th ACM international workshop on Wireless mobile multimedia*, (2002)
4. A. Yushev, P. Lehmann, A. Sikora, V.F. Groza, "Extended Performance Measurements of Scalable 6LoWPAN Networks in an Automated Physical Testbed," in *IEEE Int'l Conference I²MTC, Pisa*, (2015)
5. M. Allman, A. Caldwell, and S. Ostermann, "ONE: The Ohio Network Emulator," Ohio University, *Tech Report TR-19972*, August (1997)
6. E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek, "The Click Modular Router," *ACM Transactions on Computer Systems*, vol. **18**, pp. 263-297, (2000)
7. L. Rizzo, "Dummysnet: a simple approach to the evaluation of network protocols," *ACM Computer Communication Review*, vol. **27**, (1997)
8. <http://snad.ncsl.nist.gov/itg/nistnet>
9. K. Fall, "Network Emulation in the VINT/NS Simulator", In *Proceedings of 4th IEEE Symposium on Computers and Communications*, (1999)
10. https://en.wikipedia.org/wiki/Network_virtualization
11. D. Maltz, J. Broch, and D. Johnson. "Experiences Designing and Building a Multi-Hop Wireless Ad-Hoc Network Testbed". In *CMU TR99-116*, (1999)
12. H. Lundgren, D. Lundberg, J. Nielsen, E. Nordstrom, and C. Tscudin. "A Large-scale Testbed for Reproducible Ad Hoc Protocol Evaluations". In *Proc. of WCNC*, (2002)
13. D. Johnson, T. Stack, R. Fish, D. M. Flickinger, L. Stoller, R. Ricci, and J. Lepreau. "Mobile Emulab: A Robotic Wireless and Sensor Network Testbed" In *IEEE Infocom*, (2006)
14. <http://www.ecsl.cs.sunysb.edu/mint/>
15. L. Cremean, W. Dunbar, D. van Gogh, J. Hickey, E. Klavins, J. Meltzer, and R. M. Murray. "The Caltech Multi-Vehicle Wireless Testbed" In *Proc. of Conference on Decision and Control*, (2002)
16. D. Johnson, T. Stack, R. Fish, D. M. Flickinger, L. Stoller, R. Ricci, and J. Lepreau. "Mobile Emulab: A Robotic Wireless and Sensor Network Testbed" In *IEEE Infocom*, (2006)
17. A. Yushev, P. Lehmann, A. Sikora, "6LoWPAN with RPL Performance Measurements in an Automated Physical Testbed", *2nd IEEE IDAACS SWS, Offenburg, Germany*, pp. 29-33, (2014)
18. M. Ringwald, K. Römer, "Deployment of Sensor Networks: Problems and Passive Inspection", *Proc. of WISES '07, Madrid, Spain*, (2007)
19. http://www.ansolutions.de/documents/@ANY900_2_Module_Leaflet.pdf
20. <http://194.75.38.69/pdfs/ZFRSC-183+.pdf>
21. <http://194.75.38.69/pdfs/ZB3PD-63+.pdf>
22. <http://194.75.38.69/pdfs/VAT-30.pdf>
23. <http://194.75.38.69/pdfs/ZX76-15R5-SP+.pdf>
24. <http://194.75.38.69/pdfs/ZASWA-2-50DR+.pdf>
25. <http://194.75.38.69/pdfs/141-6SM+.pdf>
26. <http://www.rfshieldbox.com/datasheets/STE2200.pdf>
27. <http://194.75.38.69/pdfs/ANNE-50+.pdf>