Wireless propagation emulator for virtual wireless testbed

Hiroshi Mano ,†1,†2 Toshihisa Yamada ,†2
Satoshi Funada ,†3 Masanori Uno ,†3 Mineo Takai †4
and Shigeru Kaneda †5

Recently, the rapid growth of wireless broadband communication has necessitated an extensive fusion of radio and network technologies. Particularly in digital packet switching communication, there are more cross-layer trade-offs than in simple legacy wireless communications.
Due to the small liaison activity between these two research domains, implementation-based system evaluation is the only way to evaluate new ideas for wireless broadband systems. However, these implementation methods have many restrictions, and the result is always restricted by the particular conditions employed.
The purpose of this research was to propose an architecture and perform the actual implementation of a complete general-purpose virtual wireless system testbed, instead of a legacy implementation-based evaluation method. The purpose of this paper is to describe the proposed architecture for this virtual wireless testbed and report on the implementation of and results for a programmable wireless propagation emulator that is a main part of the virtual wireless testbed. As results, wireless propagations over at least 10 wireless nodes were emulated in real time using 10 FPGAs that had programmable parameters consisting of the channel model, path loss model, antenna model, and emulation scenario..
In conclusion, the proposed wireless propagation emulator enabled the virtual wireless testbed and will make environment-independent wireless system evaluation possible.

1. Introduction

In this section we describe the requirements for today's wireless testbed and the limitations of the existinged wireless testbed. And then in the following section "Proposal of virtual wireless testbed" is described in Chapter 2 first. And "Implementation of a programmable wireless propagation emulator" is explained in Chapter 3. The "Evaluation of the programmable wireless propagation emulator" is reported in Chapter 4. Chapters 5 and 6 provide a discussion of the virtual wireless testbed and the conclusion.

†1 University of Yamanashi
∗1 Presently with Interdisciplinary Graduate School of Medicine and Engineering
†2 Root,Inc.
†3 e-trees Japan,Inc.
†4 Waseda University/University of California, Los Angeles
†5 Space-Time Engineering Japan, Inc.

(1) Requirement of wireless testbed

Recently, the purpose of wireless communication has changed from a single, specific application to general digital applications. Most of the innovations related to this change have been achieved by combining wireless and network technologies. Particularly in digital packet switching communication, there are trade-offs between the radio and network layers. However, there has never been a crossover for these technology domains in the form of a black box. Therefore, we have to evaluate some of the new ideas using specific environments in field trials. For example, the high-speed hand-off idea is evaluated using a particular radio link (PHY and MAC) technology, even though the design does not depend on a radio link. On the other hand, even for radio technologies (antenna, modulation, etc.) that do not depend on upper layers such as the network, transport, and application layers, the final evaluation is performed using specific high-layer technologies (TCP/IP, Mobile IP, EAP, etc.). In addition, there are limitations on the license and the geographical environments for the evaluation of a wireless system, which increase the evaluation cost. Therefore, if a virtual wireless testbed can maintain the independence of each item given in Table 1 while providing an evaluation environment, it enables the improvement of new wireless system innovations.

(2) Limitation of existing wireless testbed

Some of the wireless testbeds described in Table2 exist for the purpose of researching a new generation of networks. However, all these testbeds were designed from the viewpoint of either a wireless or a network environment. For example, the SDFR-based testbed uses a felxible MAC/Link layer technologies, but the actual system evaluation should be done in a particular field environment. The Wi-Fi-based testbed is available for a general-purpose examination of the higher protocol, but it can only be used with Wi-Fi radio characteristics (2.4 GHz/5 GHz, DSSS, OFDM, etc.). The DSP-based testbed only emulates the wireless propagation of a particular wireless system. As shown in this comparison, no wireless testbed has been realized that has wide adaptability from the physical layer to the application layer.

Table 1 Required independency of wireless testbed

Geometric location independent
Environment model independent (urban, suburban, rural, indoor, outdoor)
Moving or Fixed
Propagation model independent (line-of-sight, non-line-of-sight)
Physical layer independent
Frontend characteristics independent (frequency, power, antenna, filter, noise figure etc,.)
Modulation independent (OFDM, DSSS, PSK, CDMA, etc.)
MAC-layer independent
Duplex (FDD, TDD)
Multiple access (CSMA, TDMA, FDMA, CDMA, etc.)
Link-Layer topology independent
Mobile system (mover, base station)
Fixed wireless system (base Station, CPE)
Broadcasting system

Table 2 Comparison with existing testbed

Layer	Condition	SFDR	Wi-Fi	DSP	FPGA+SFDR
Geometric	Propagation	Fixed	Fixed	Programmable	Programmable
	Environment	Fixed	Fixed	Programmable	Programmable
Physical	Frontend	Fixed	Fixed	Fixed	Programmable
	Modulation	Reconfigurable	Selectable	Fixed	Reconfigurable
MAC	Duplex	Reconfigurable	Selectable	Fixed	Reconfigurable
	Multi Access	Programmable	Fixed	Fixed	Reconfigurable
Link Layer	Topology	Programmable	Selectable	Fixed	Reconfigurable
Existing case		GNU radio	Orbit	CMU	Proposed

2. Proposal of virtual wireless testbed

(1) Architecture of virtual wireless testbed

The proposed virtual wireless testbed architecture is shown in Fig. 1. This testbed incorporates a scenario generator, reconfigurable wireless node, and programmable wireless propagation emulator. Details on each of the listed modules are given below.

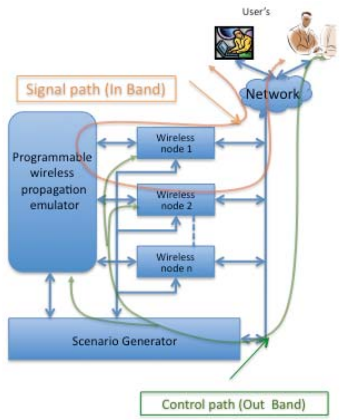

Fig. 1 Architecture of virtual wireless testbed

(a) Scenario generator

The scenario generator provides the functions for a user interface, resources management (setting/selecting parameters and scenarios), and the calculation of the dynamic propagation scenario that will be executed by the emulator. A commercial off-the-shelf personal computer is used as the scenario generator.

(b) Wireless node

The wireless node shown in Fig. 2, which is implemented on FPGAs, provides a fully reconfigurable PHY, MAC, and Frontend (roll-off filter, AGC, etc.). In addition, this wireless node incorporates an embedded computer, which will be used for the link or higher layer control. In the case of a virtual network application, a user may also implement a virtual host on this embedded computer.

(2) Programmable wireless propagation emulator

The programmable wireless propagation emulator shown in Fig. 3 emulates the propagation path between wireless nodes. The channel and path loss model are implemented using an FIR filter on FPGA hardware. All wireless propagation behavior can be described as a combination of delay, amplitude, and phase. The channel model is implemented with a 7-tap FIR that also emulates frequency

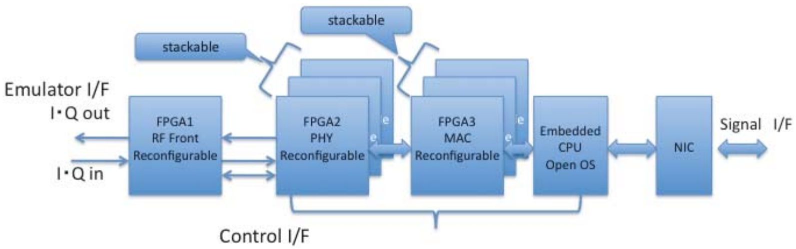

Fig. 2 Wireless Node

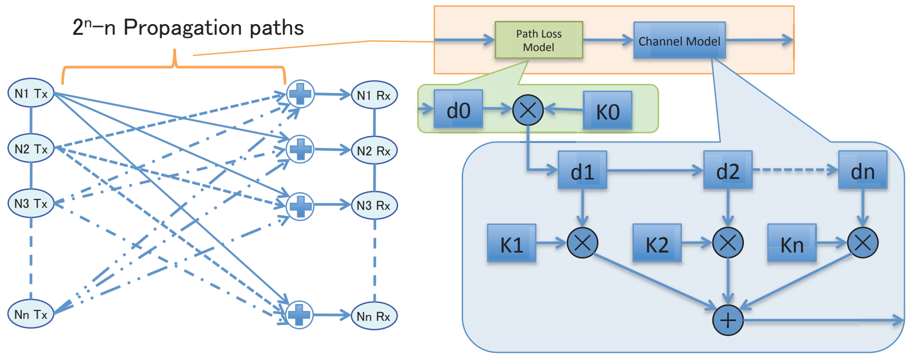

Fig. 3 Programmable wireless propagation emulator

selective fading. The multiplication value (Kn, dn in Fig. 3) for the signal is calculated beforehand using a scenario generator, and it is updated based on the time with the help of a user-defined scenario. All the wireless nodes are connected to each other through this propagation emulator by I/Q complex signals. Therefore, $(2^n - n)$ propagation paths are required for this propagation emulator.

(3) Prospective uses for virtual wireless testbed

Because of the wide adaptability of the proposed virtual wireless testbed, there are numerous prospective uses with no regulatory limitations. This will allow users to evaluate their own new ideas without requiring radio licenses and test fields. The use case shown in the Fig. 4 are some of the conceptual prospective examples.

For example, we spent much time and money to test a high-speed hand-over protocol over Wi-Fi at an actual motor sports course with a professional driver and a race car. However, this only demonstrated the performance under a particular set of conditions (2.4 GHz, DSS, single vehicle, closed circuit environment). If we wanted to perform the same evaluation using different preconditions (OFDM, 5.8 GHz, multiple vehicles, etc.), we would have to spend the same volume of resources. In addition, and most importantly, the user only needs to program or configure the target part to be evaluated. All the other parts will be described by ideal parameters. For example, if a user wants to evaluate a new idea for an antenna, they need to program only the antenna characteristics, while selectable predetermined values are used for all of the other layers such as the PHY, MAC, LINK, and transport. .

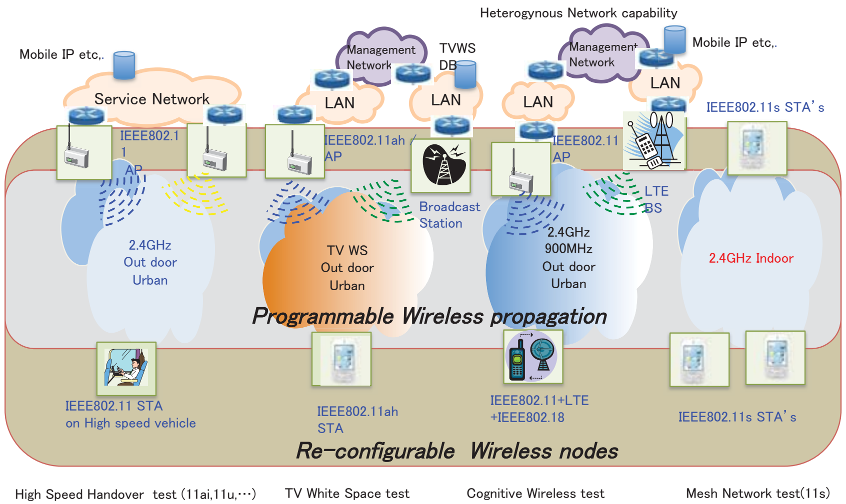

Fig. 4 Prospected uses

3. Implementation of programmable wireless propagation emulator

We implemented the scenario generator and programmable wireless propagation emulator for 10 wireless nodes, as explained in this section. The scenario generator is a software on a Linux system, and the programmable wireless propagation emulator is a hardware board that integrates FPGAs. A 1 Gbit/s Ethernet connects the scenario generator and programmable wireless propagation emulator. A system overview is shown in Fig. 5.

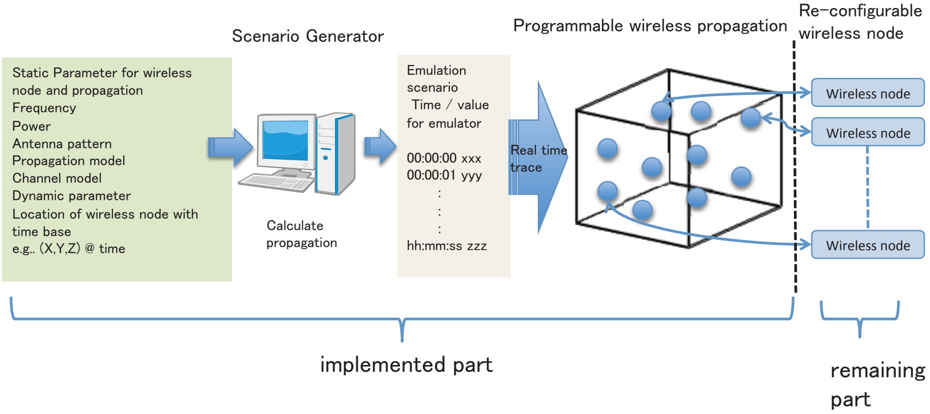

Fig. 5 System overview

(1) **Scenario generator**

The scenario generator is a software running on a Linux system, and the following functions were developed and implemented on it.

(a) **GUI**

We implemented the entire user interface as Flash and PHP applications, while considering system sharing. This GUI provides a user-friendly interface that shows the scenario behavior and all the parameters for wireless nodes.

(b) **Propagation characteristics generation**

The scenario generator calculates the propagation fluctuation parameter using the conditions that were configured by the user through the GUI. The implemented prototype supports the configurable conditions for the propagation path shown in Table 3.

(c) **Path loss calculation**

Table 3 Configurable conditions

Path loss model	Channel ITU-R M.1225	Fading Model	Antenna
Two ray	Indoor A/B	Classic	Isotopic
COST231Hata	Pedestrian A/B	Flat	Dipole
	Vehicular A/B	User defined	ITU-R M.2135
	PDSekm		ITU-R F.1245
	User defined		User defined

The scenario generator calculates the path loss between nodes using the very familiar formulas described in the following. In the case of the two-ray model, if the distance is short enough, the path loss is given by (1), and if it is longer path loss is given by (2).

$$L = 20 * \log_{10}\left(\frac{4\pi d}{\lambda}\right) \tag{1}$$

$$L = 40 * \log_{10}(d) - 20 * \log_{10}(ht) - 20 * \log_{10}(hr) \tag{2}$$

In the case of the Cost 231 Hata model, the path loss is given by

$$\begin{aligned} &L = 46.3 + 33.9\log(f) - 13.82\log(ht) - a(hr) \\ &+ [44.9 - 6.55\log(ht)] * \log(d) + 3 \\ &a(hr) = (1.1\log(f) - 0.7) * hr - (1.56\log(f) - 0.8) \end{aligned} \tag{3}$$

where d = distance, λ = wave length, ht, hr = antenna height of transmuter and receiver f = frequency.

(d) **Fading calculation**

The scenario generator calculates the vector data for the Rayleigh fading wave r(t) using

$$\begin{aligned} &r(t) = x(t) + j * y(t) \\ &= \left[\sqrt{\frac{2}{N_1+1}} \sum_{n=1}^{N_1} \sin\left(\frac{\pi n}{N_1}\right) cos\left\{2\pi f_d cos\left(\frac{2\pi n}{N_1}\right) t\right\} + \frac{1}{\sqrt{N_1+1}} cos\left(2\pi f_d t\right)\right] \\ &+ j\sqrt{\frac{2}{N_1}} \sum_{n=1}^{N_1} \sin\left(\frac{\pi n}{N_1}\right) cos\left\{2\pi f_d cos\left(\frac{2\pi n}{N_1}\right) t\right\} \end{aligned} \tag{4}$$

where fd = Maximum fading frequency, $N1$ = number of FIR taps.

(e) **Emulation scenario**

The evaluation scenario is given by the time and location (x, y, z, and azimuth) of the wireless node using the GUI, as shown in Fig. 6. The scenario generator calculates the time-based fluctuation of the path loss using the process described above.

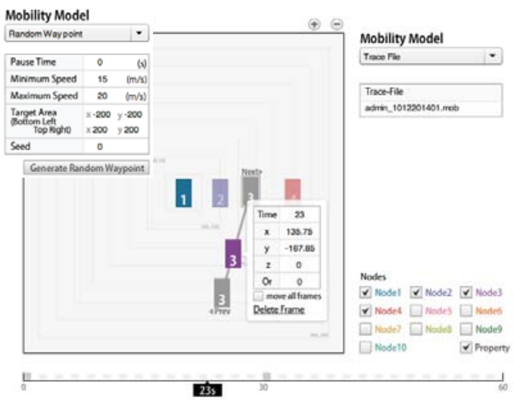

Fig. 6 Emulation scenario GUI

(2) **Programmable wireless propagation emulator**

The programmable wireless propagation emulator is a hardware that integrates 10 FPGAs (Vertex-6 XC6VLX130T /1156pin/200MHz). e implemented this 10-node capability in the virtual wireless space with a 49% slice, 74% DSP and 60% RAM consumption. After calculating the vector data of the path loss and channel model using the scenario generator, the data are transferred to the programmable wireless propagation emulator by Gigabit Ethernet. The programmable wireless propagation emulator emulates real time propagation by combining the delay memory and complex multipliers, as shown in Fig. 7.

4. Evaluation of programmable wireless propagation emulator

We evaluated the accuracy, process delay, and behavior of the implemented programmable wireless propagation emulator. In this section, the evaluation methodology is explained first, and then, the results of the evaluation are described.

(1) **Evaluation methodology**

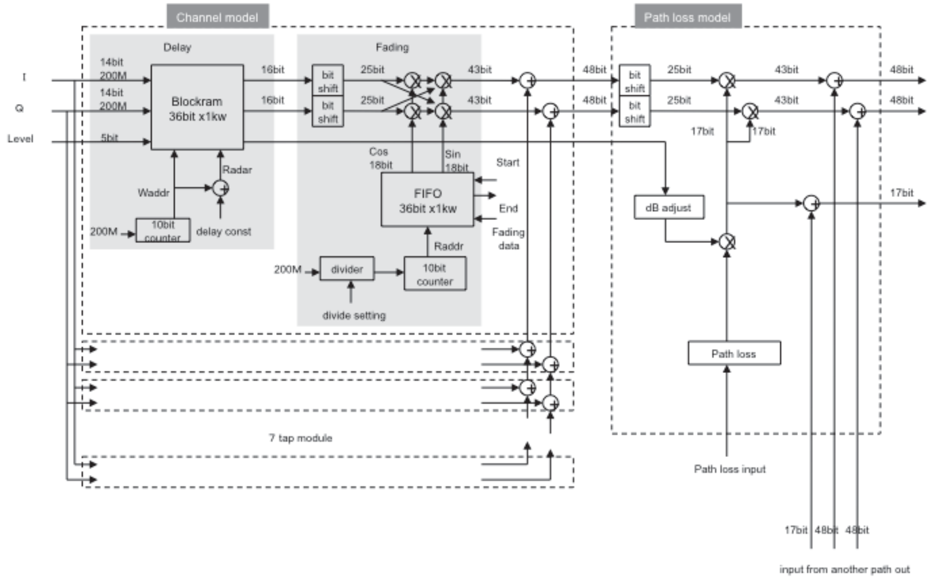

Fig. 7 Block diagram of programmable wireless propagation emulator

We measured and evaluated the accuracy, process delay, and behavior using the existing legacy radio measurement equipment such as a spectrum analyzer, signal generator, and commercial Wi-Fi products. Because the programmable wireless emulator only had digital I/Q interfaces, it was not suitable to verify with visible inspection. Therefore, we prepared an A/D (analog-digital) and D/A (digital-analog) converter for use between the measurement equipment and programmable wireless emulator, as shown in Fig. 8.

(2) **Evaluated characteristics and result**

We evaluated the following characteristics using 2.4GHz analog signals.

(a) **Attenuation accuracy**

We measured the attenuation accuracy of the emulator for CW and OFDM signals. These measurements confirmed that the propagation wireless emulator attenuates the input signal with an accuracy of ±1.0dB and a resolution of 0.1dB.

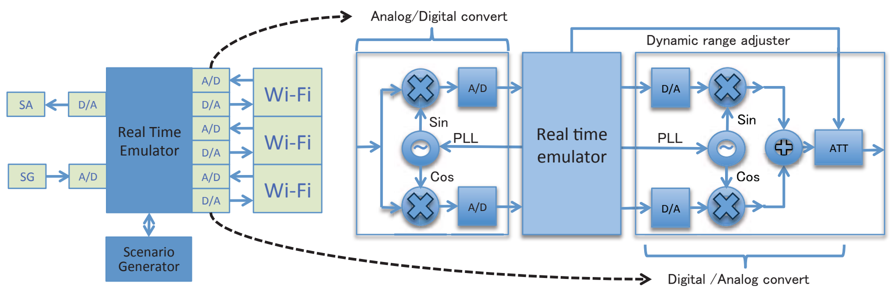

Fig. 8 Evaluation block diagram

(b) **Channel model execution**
We verified the channel model fading with an OFDM signal, as shown in Fig. 9. The left side shows the effect of flat fading, and the right side shows the effect of frequency selective fading.

(c) **Emulation delay**
We measured the signal processing delay of the programmable wireless propagation emulator using a single pulse signal. This measurement confirmed that the processing delay of the propagation wireless emulator is less than 1 us. This 1-us delay is sufficiently smaller than the typical delay of standard radio.

(d) **Multi channel capability**
We installed 2 sets of Wi-Fi (AP and STA) links over the programmable wireless emulator using 2 different channels. Both these Wi-Fi link sets were established over the emulator. We programmed the emulator so that all the signals were added to an output port, and then measured the output using the spectrum analyzer. The measured spectrum shown in Fig. 10 verifies that the programmable wireless propagation emulator has multi-channel capability.

(e) **Dynamic emulation**
We programmed a scenario that increases and decreases the distance between 2 wireless nodes based on time to emulate a vehicular scenario. We then measured the attenuated signal level with the CW signal. Fig. 11 shows the measured output signal levels for different distance-changing

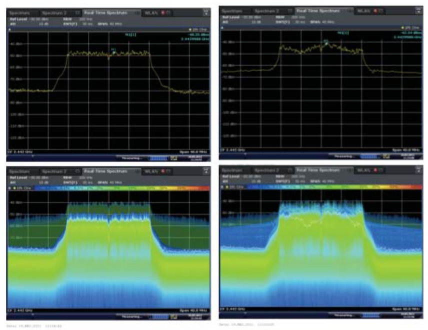

Fig. 9 Fading effect

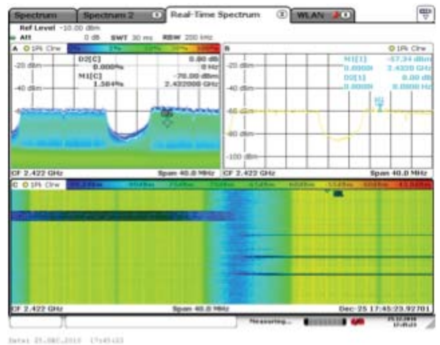

Fig. 10 Multi channel capability

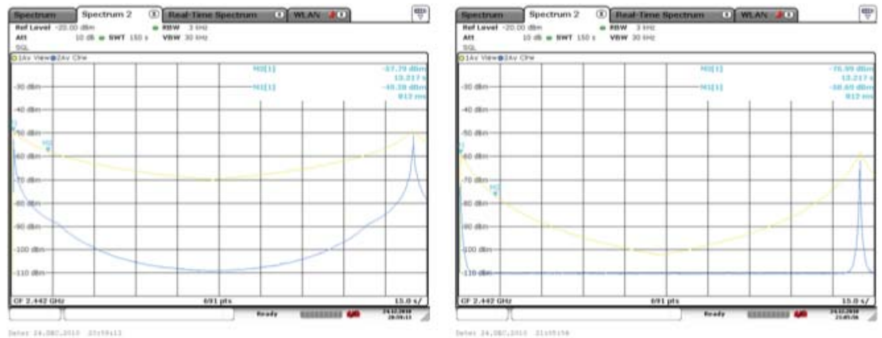

Fig. 11 Signal output level with vehicular scenario

speeds and path loss models. The left side shows the result for a speed of 5 km/h, while the right side shows a 200-km/h speed. The upper line shows the two-ray model and the lower line shows the COST-231 Hata model.

(f) **EVM (error-vector-magnitude)**
We measured the internal effect of the EVM quality using an ideal OFDM signal. Table 4 lists the measured EVM data of the original signal and a signal after the emulator passage. As can be seen from these measured data, the EVM quality was decreased. However, it is reasonable to assume

that these changes were caused by the characteristics of the analog converters because the emulator part performs only digital operations and the typical digital operation error is not as small as that which is measured. In addition, the results showed that there was no significant quality change that would violate the IEEE802.11 standard.

Table 4 EVM over emulator

Item	Measurement point		Difference	Unit	IEEE802.11
	Input	Output			Limit
EVM All Carr	1.03	2.95	+1.92	%	5.62
	-39.76	-30.61	-9.15	dB	-25.00
EVM Data Carr	1.04	2.98	+1.90	%	5.62
	-39.69	-30.53	-8.84	-25.00	-25.00
EMV Pilot Carr	0.91	2.57	-1.6	%	39.81
	-40.77	-31.80	-9.46	dB	-8.00
IQ Offset	-69.50	-58.08	-11.42	dB	-15.00
Gain Imbalance	-0.09	-0.07	-0.02	%	-
	-0.01	-0.01	0	dB	-
Quadrature Err	0.00	-0.04	0.02		
Freq Err	-149.13	-152.60	-3.60	Hz	±48840
Symb Clock Err	0.09	-0.03	0.12	ppm	±20

5. Discussion

In this study, we proposed the basic architecture for a virtual wireless testbed and reported the implementation of a wireless propagation emulator. The results of the actual implementation of the programmable wireless propagation emulator showed the following:

- The wireless propagation emulator emulates wireless propagation with sufficient accuracy (±1.0dB).
- The emulation delay is small enough for this system to provide a real-time air interface instead of actual air (1.0μSec) .
- The wireless propagation emulator emulates the typical behavior of flat fading and frequency selective fading.
- Complex vector signals with at least a 85MHz bandwidth can be emulated simultaneously.
- A 10-node emulator can be realized using an existing commercial FPGA product.

The results of this implementation demonstrated that the combination of a scenario generator and that the propagation emulator is capable of performing propagation emulation with wide adaptability. Compared to existing testbeds, the proposed programmable wireless propagation emulator can be implemented without depending on wireless node characteristics (PHY, MAC, Link, and upper layers). The scalability for large number of nodes by proposed architecture shall be limited by the number of I/O pins of FPGA, because we divided a propagation module into 10 FPGAs. However the large integration FPGA may solve this scalability problem to reduce the number of FPGAs.

6. Conclusion

In conclusion, the architecture for a virtual wireless testbed was proposed, and the technical feasibility of the actual implementation of a programmable wireless propagation emulator was demonstrated. The performance of the programmable wireless propagation emulator enabled the virtual wireless testbed and will provide environment-independent wireless system evaluation. We will implement the reconfigurable wireless node, which is the remaining part of the proposed virtual wireless testbed architecture, in the next work. In addition, we are going to evaluate our idea of a fast initial link setup protocol, which will be proposed as IEEE802.11ai, using the virtual wireless testbed. In case of IEEE802.11ai, we have to evaluate a large number of STA accesses to AP using a vehicular model, which is a typical use for a virtual wireless testbed.

Acknowledgments
The assistance received from Tetsuya Shigeyasu, Hiroshi Matsuno, Akihiro Nakao and Masanori Hanawa in the preparation of this manuscript is gratefully acknowledged.

References

1) E. Blossom,GNU Radio: Tools for exploring the radio frequency spectrum, Linux Journal, vol.2004, no.122, June 2004.
2) http://gnuradio.org/

3) D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu and M. Singh, "Overview of the ORBIT Radio Grid Testbed for Evaluation of Next-Generation Wireless Network Protocols", Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC 2005)
4) http://www.orbit-lab.org/
5) A Using Physical Layer Emulation to Understand and Improve Wireless Networks, Glenn Judd, PhD Thesis, Department of Computer Science, Carnegie Mellon University, October 2006. Also published as technical report CMU-CS-06-164.
6) http://www.cs.cmu.edu/emulator/intro.html
7) ITU-R M.1225
8) Two ray ground reflection model "Wireless Communications, Principles, and Practice", T. Rappaport
9) COST 231 "Digital mobile radio towards future generation systems", COST Action 231 Final Report, EUR 18957, 1999.
10) Hiroshi Harada and Ramjee Prasad , Simulation and Software Radio for Mobile Communications