Beyond 4G: Open Source Telecom Leveraging MIMO and Distributed P2P Connectivity for Wired Independence

Tirtharaj Bhattacharjee and Brent R. Petersen
University of New Brunswick
Department of Electrical and Computer Engineering
Fredericton, NB, Canada – E3B 5A3
Email: tirtho@ipware.ca, b.petersen@ieee.org

Abstract

With the advent of multi-processing and multi-core systems, parallels in multiplexing radio-frequency (RF) transmission by Multiple-Input-Multiple-Output (MIMO) is advantageous. Convergence architecture for Beyond-Fourth-Generation (B4G) communications is considered utilizing open standards, MIMO and P2P topology. The system is observed as a software host-media-processing (HMP) appliance aggregated with modularized interfacing for distributed network communications. Experiments conducted on the modular interfacing components demonstrate active and passive wireless link properties of 802.11 b, g and n under LOS, NLOS, VLOS scenarios as compared to Ethernet. Codec's used in the system (G711a, G711u, G723.1, G726-32, and G729a) for communication are tested under QoS bandwidth starvation for behavior of bi-directional throughput within the networked system. UDP protocol saturation tests on IPv6 and IPv4 are conducted. The model of the appliance is proposed for voice, data, and video communications anytime, anywhere.

1. Introduction

With the rapid movement towards wireless systems such as Wireless-Digital-Public-Switched-Telephone-Networks (WPSTNs), Wireless-Local-Area-Networks (WLANs), Wireless-Metropolitan-Area-Networks (WMANs 802.16), Voice-over-Wireless-Networks (VoWF), Wireless-Private-Automated-Branch-Exchanges (WPABXs) there is a need to establish wireless Peer-to-Peer (P2P) communication systems capable of transversing large heterogeneous networking environments without need for Network-Address-Translation (NAT). Bottlenecks in throughput on traditional wireless infrastructure and growth in information capacity of the Internet requires wired speeds to be established wirelessly. One such technology promising wired speeds on wireless infrastructure is MIMO multiplexing RF technology. Instead of having one highway using traditional Single-Input-Single-Output (SISO) receiver/transmitter architecture, this paper proposes the utilization of multiple highways, thus multiplexing throughput to make wired independence a reality. Advances in the social open source software movement (GNU) facilitate implementation of wireless inter-connectivity devices for VoIP session exchange. Logical segmentation of Time-Division-Multiplexing (TDM) logic within software is made possible using the Asterisk*® soft-arbiter framework.

Figure 1, shows network topologies over time all converging to MIMO technology in a convergence framework for voice, data and video anytime anywhere.

Figure 2, contains the paradigm for IP evolution over time. The figure shows IPv4-IPv4 communications evolving to IPv6-IPv6 provisioned for increased number of IP-aware devices.

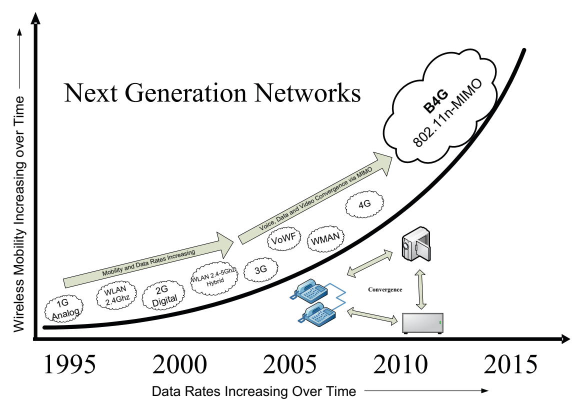


Figure 1. Convergence to B4G with MIMO

978-0-7695-3135-9/08 $25.00 © 2008 IEEE
DOI 10.1109/CNSR.2008.71

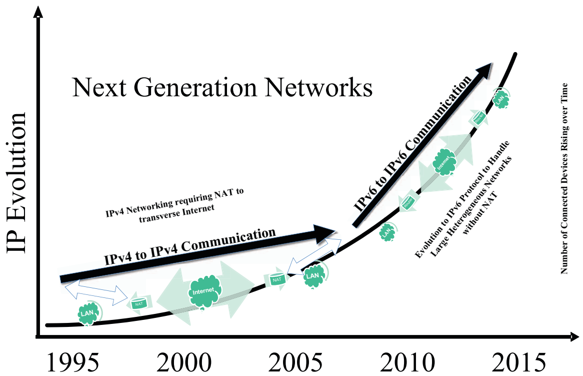


Figure 2. IPv4 Evolving to IPv6

The proposal to use IPv6 becomes evident as the new protocol establishes autonomous auto-configurable P2P terminated networks, as specified in RFC 1883. IPv6 allows for greater transparency in a P2P fashion without need for NAT [1]. IPv6 also allows backwards compatibility with an IPv4 network through encapsulation of IPv6 packets within the IPv4 packets transversing an IPv4 network. IPv6 also has provisions for tunneling capabilities and can establish communications on mixed protocol environments [2]. Session-Initiation-Protocol (SIP) (RFC 3261) [3] has been popularized in VoIP Communications but requires both Real-Time-Protocol (RTP) and UDP to transverse publicly accessible hosts. IPv4-SIP requires publicly accessible IPv4-IPs for IPv4-SIP-to-SIP (IPv4-S2S) inter-operation. In cases of Private-Local-Area-Networks (PLANs), more than one computer/host needs to be translated to a single public IP through NAT. IPv4-S2S inter-access is possible through a demilitarized zone (DMZ) on the Internet gateway. The DMZ simply maps the machine of interest within the PLAN as if it were directly connected to the publicly accessible IP thus allowing IPv4-S2S communication. S2S sessions are controlled through a series of control signals sent through as RTP.

This patchwork approach with IPv4-NAT under PLANs makes IPv4-SIP unattractive to manage large networks of evolving complication. A solution to this problem is the IPv6 protocol (RFC 1883). To demonstrate the appliance proposed, the physical link properties of the modular interconnections need to be considered. The appliance proposed offers Telephony wirelessly at Ethernet (802.3/802.3u) speeds realized by MIMO. The experiments conducted use a flooding technique to saturate the wireless communication link and offer insights to what inter-link technology gives best throughput for use with the proposed appliance.

Experiments 1-7 consider the physical wireless link properties by flooding UDP packets up the RF pipe of a Wireless B, Wireless G, and Wireless N (MIMO) as compared to each other and Ethernet. Measurements of link properties are conducted in Line-of-Sight (LOS), Non/Near-Line-of-Sight (NLOS) and Mobile/Variable-Line-of-Sight (VLOS) scenarios to effectively re-create results observed in real world implementations of the appliance. In Experiment 8, this paper offers VoIP Telecommunications tests for re-creating bi-directional voice communications with the appliance. Effects of bandwidth starvation through recording of a signal through to the appliance are considered. The recorded file is a wave file, in Experiment 8, codecs G711u, G711a, G723.1, G726-32, and G729a are tested. The codecs transverse from an Intranet via gateway or router to an Extranet where the gateway is managed through a graphical user interface and offers QoS bandwidth limitation management. The recorded wave file is composed of a repeated constant signal which is recorded bi-directionally in full duplex. Therefore, the signal generated from the Plain-Old-Telephone-System (POTS) is sent via gateway to the PBX on the Extranet and a signal generated on the Extranet transverses the gateway to the POTS phone on the Intranet. The codec matrix switching is made possible by using Sipura® 3000 Analog-Telephone-Adaptor (ATA) which ties a regular POTS phone over the Ethernet to the Asterisk* PBX [4] where the PBX itself records the signal received as a wave file. In Experiment 9, P2P properties of IPv6 are compared to IPv4, tested on a Sun Microsystems Sunfire® X2100 1U server running Linux installed with Iperf (a UDP saturation testing application) to show the advantages of IPv6 protocol. This paper then proceeds to model the system as an appliance with capability of being scalable and dynamically adaptable leveraging open standards, virtualization and distributed storage over the Internet.

2. Wireless Hardware for Experimentation Introduced

In this paper, we tested a Wireless B [5] to Wireless B (802.11B-802.11B), Wireless G [6] to G (802.11G-802.11G), Wireless N (MIMO) [7] to N (802.11N-802.11N), and Wireless N under B mode to Wireless B (801.11N-MIMO-B-802.11B), Wireless N under G mode to G (802.11N-MIMO-G-802.11G) under LOS, NLOS, and VLOS situations. In our experiments, we used a Linksys® WRT54GS router as the Access Point, and an Alfa USB Wireless B, G card as the client card for Wireless B and Wireless G tests. For MIMO Wireless N tests, we used Linksys WRT300N MIMO router Access Point with Linksys WUSB300N USB MIMO client network card. For Wireless N to Wireless B and G, we used a Linksys WRT54GS router Access Point linked to the Linksys WUSB300N MIMO USB client card.

Figure 3, demonstrates the outline of Experiments 1-7 conducted to measure link properties of 802.11 b, g and n under LOS, NLOS and VLOS scenarios.

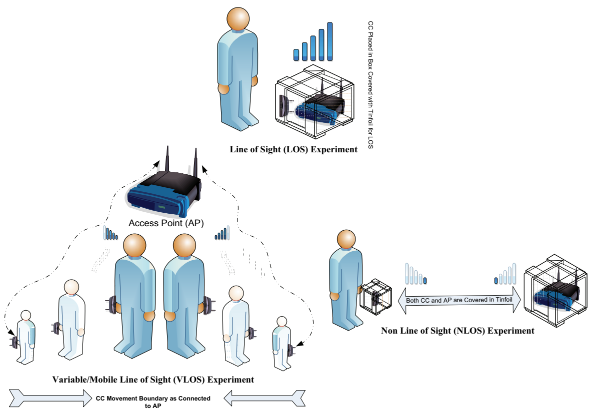


Figure 3. Experiments 1-7 Overview

3. Experiments 1-7: LOS, NLOS and VLOS Explored

For a LOS communication, we placed both the Access Point (AP) and Client Card (CC) in a box and covered the box inside/outside with tinfoil to simulate a controlled environment as best as possible. We measured both active and passive properties of the signal. Active properties of the signal strength were shown on the APs graphical user interface showing 100% signal strength calculated by the router. On a Wireless B LOS link, the channel capacity as stated by the AP was 11 Mbps. To test the throughput of the RF link we used a saturation testing tool called UDP Flooder 2.0 to generate UDP traffic. The tool itself in a single session was only capable of generating approximately 34 Mbps of traffic over 100 Mbps Ethernet link so we could reliably compare this control measurement to what we should see on a 11 Mbps Wireless B link. With Wireless G we used two flooders increasing the throughput going up the RF pipe on the Wireless G link, to exceed the reported LOS rate of 54 Mbps displayed by the active link properties of the router.

For NLOS communications, the AP and CC were placed in separate boxes both individually covered with tinfoil and sections of the tinfoil were opened to allow signal leakage until it was noticed on the active signal properties shown in the APs graphical user interface a rate less than 70% signal strength. Similar to the LOS test we ran the UDP Flooder 2.0 and measured Ethernet UDP flooding throughput over the RF pipe and recorded the rate.

For VLOS communications we left the AP in the same box as for NLOS experiment but attached a 3 meter USB cable connected to the client card. The client card for Wireless B to Wireless B and Wireless G to G was the Alfa network card and for the Wireless N to N, Wireless N to B and Wireless N to G we used the Linksys WUSB300N MIMO network card. In VLOS the CC was moved at a steady rate, the distance of ten feet while a screen capture application was invoked on a PC observing active and passive variations in signal strength. In addition, the UDP Flooder tool was invoked and the rates of passive transmission throughput up the RF pipe was recorded by manipulating the tinfoil to observe various signal strengths.

It should be noted that as the UDP Flooder tool was only capable of generating 34 Mbps of traffic per session. MIMO active RF bandwidth rates reported on the router management console at LOS were 270 Mbps. It was decided to run six sessions of the flooder tool to increase data throughput going up the RF pipe to about 204 Mbps. We could not exceed 6 sessions due to computing limitations of the hardware attached to the MIMO client card to generate additional UDP traffic. Tables 1-3 summarize results observed in LOS, NLOS and VLOS experiments.

Table 1: LOS Experimentation Results

(TCP) Wireless B, G and N in LOS Experimental Results			
	802.11B	**802.11G**	**802.11N**
802.11 B (802.11B)	~ 5.7Mbps		
802.11 G (802.11G)		~ 21.3Mbps	
802.11 N-N AP set to N Mode			~ 80Mbps - ~ 93.4Mbps
802.11 N-B AP set to B Mode			~ 6.7Mbps
802.11 N-G AP set to G Mode			~ 28.8Mbps

Table 2: NLOS Experimentation Results

(TCP) Wireless B, G and N in NLOS Experimental Results			
	802.11B	**802.11G**	**802.11N**
802.11 B (802.11B)	~ 165kbps - ~ 280kbps		
802.11 G (802.11G)		~ 165kbps - ~ 280kbps	
802.11 N-N AP set to N Mode			~ 22Mbps
802.11 N-B AP set to B Mode			~ 4.2Mbps
802.11 N-G AP set to G Mode			~ 6.2Mbps - ~ 12Mbps

Table 3: VLOS Experimentation Results

(TCP) Wireless B, G and N in VLOS Experimental Results			
	802.11B	**802.11G**	**802.11N**
802.11 B (802.11B)	~ 165kbps - ~ 5.7Mbps		
802.11 G (802.11G)		~ 165kbps - ~ 21.3Mbps	
802.11 N-N AP set to N Mode			~ 22Mbps - ~ 93.4Mbps
802.11 N-B AP set to B Mode			~ 4.2Mbps - ~ 6.7Mbps
802.11 N-G AP set to G Mode			~ 6.2Mbps - ~ 28.8Mbps

Figure 4, shows transceiver block diagrams of air interfaces 802.11 B, G, N used in Experiments 1-7 to accompany results from Tables 1-3.

4. Experiment 8: Bi-directional PBX Codec Performance Explored

Experiment 8 was conducted transversing two networks, the first network was the Extranet encompassing the PBX connected on the WAN side of the gateway managed with tree based Hierarchical Token Bucket (HTB) Quality-of-Service (QoS). HTB QoS allows the uplink and down link properties to be adjustable on the gateway thus allowing capping of uplink and downlink transversing both LAN and WAN. The LAN side of the gateway is an Intranet that encompasses a Sipura 3000 ATA hooked up to a POTS phone.

Both the PBX and ATA are able to talk to each other under various codecs within this experiment. To establish bi-directional communication in full duplex the PBX is setup with a conference room extension. The conference room is dialed by the POTS phone from the Intranet and a signal composed of a generated Text-to-Speech word (Test.) spoken five times is transmitted to the PBX on the Extranet. Similarly, the PBX also generates a speech signal (You are the only person in this conference) which is heard by the POTS phone upon entering the conference room on the Intranet. A recorded Pulse-Code-Modulated (PCM) wave file which multiplexes both incoming and outgoing signals was produced. The resulting performance of the codecs varying bandwidth in HTB QoS of 100 kbps, 65 kbps, 13 kbps and 9 kbps are demonstrated in the graphs below.

Figure 7, shows analysis of selected signals in time and frequency domain with call quality depreciation for one period of the spoken word (Test.) for codecs G711u at 13kbps, G711a at 13kbps and G726-32 at 13kbps.

Figure 8, shows analysis of signals in time and frequency domain under several bandwidth rates for codecs G711a, G711u, G726-32, G723.1 and G729a.

5. Experiment 9: IPv4/IPv6 Protocol Performance Explored

Originally, experimentation was conducted using a large array of switches to conduct IPv6 vs. IPv4 NAT performance tests but due to hardware complications the test was scrapped and the core performance of the IPv6 protocol was considered using the loopback device. The loopback device consists of both a link local IPv6 address and a link local IPv4 [8] address which allow simulation of a client server application. An application is invoked as client and server on two different terminals to establish client-server infrastructure in this experiment. On a Sun Microsystems Sunfire® X2100 1U server running SUSE® 10.3 Linux the loopback device has advantages in that as the link is encapsulated on a single machine limiting issues related to interference or link malfunction. A program called Iperf was compiled and invoked within the two terminals to test UDP bandwidth limits of IPv6 and IPv4. FTP transfer via VSFTPD was conducted by copying a 190 MB test.rpm file over the loopback device. VSFTPD was used due to its ability to switch protocols between IPv4 and IPv6. A further test was conducted from two machines separated by a switch to ping/ping6 each other over IPv4 and IPv6 to show their round trip times. Tables 4-8 summarize results observed in IPv4/IPv6 protocol experiments.

Table 4: UDP IPv4-IPv4 Client-Server Iperf Saturation Tests

(UDP) IPv4-IPv4 Test Results	
Client-Server Action: iperf -s (Gbits/sec)	**UDP Saturation Results: (Gbits/sec)**
[4] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 53572	[4] 0.0-10.0 sec 5.04 GBytes **4.33 Gbits/sec**
[5] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 53573	[5] 0.0-10.0 sec 4.96 GBytes **4.26 Gbits/sec**
[4] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 53574	[4] 0.0-10.0 sec 5.85 GBytes **5.02 Gbits/sec**
[5] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 53575	[5] 0.0-10.0 sec 4.99 GBytes **4.29 Gbits/sec**
[4] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 53576	[4] 0.0-10.0 sec 6.15 GBytes **5.28 Gbits/sec**
[5] local 127.0.0.1 port 5001 connected with 127.0.0.1 port 53577	[5] 0.0-10.0 sec 5.33 GBytes **4.63 Gbits/sec**

Table 5: UDP IPv6-IPv6 Client-Server Iperf Saturation Tests

(UDP) IPv6-IPv6 Test Results	
Client-Server Action: iperf -V -s (Gbits/sec)	**UDP Saturation Results: (Gbits/sec)**
[4] local ::1 port 5001 connected with ::1 port 47266	[4] 0.0-10.0 sec 4.94 GBytes **4.24 Gbits/sec**
[5] local ::1 port 5001 connected with ::1 port 47267	[5] 0.0-10.0 sec 4.58 GBytes **3.93 Gbits/sec**
[4] local ::1 port 5001 connected with ::1 port 47268	[4] 0.0-10.0 sec 4.54 GBytes **3.90 Gbits/sec**
[5] local ::1 port 5001 connected with ::1 port 47269	[5] 0.0-10.0 sec 4.62 GBytes **3.97 Gbits/sec**
[4] local ::1 port 5001 connected with ::1 port 44724	[4] 0.0-10.0 sec 4.80 GBytes **4.12 Gbits/sec**
[5] local ::1 port 5001 connected with ::1 port 44725	[5] 0.0-10.0 sec 4.55 GBytes **3.91 Gbits/sec**

Table 6: TCP Ping6-IPv6 vs. Ping-IPv4 Tests

(TCP) IPv6 and IPv4 Ping Test Results Separated by a Switch	
Ping-IPv4: ping6 -I en0 fe80::2e0:81ff:fe5e:9d32	**Ping-IPv4: ping 192.168.99.40**
82 packets transmitted, 82 packets received, 0% packet loss round-trip min/avg/max **= 0.227/7.520/578.707 ms**	68 packets transmitted, 68 packets received, 0% packet loss round-trip min/avg/max/stddev **= 0.282/0.320/0.471/0.021 ms**

Table 7: LFTP-Get TCP Transfer using VSFTPD for IPv4

(TCP) IPv4-IPv4 LFTP-Get TCP Test Results	
Command: lftp -u tirtho,anonymous 127.0.0.1	**Results: (M/s)**
lftp tirtho@192.168.99.40: > get test.rpm	190978562 bytes transferred in 1 second **(121.99M/s)**
lftp tirtho@192.168.99.40: > get test.rpm	190978562 bytes transferred in 1 second **(119.04M/s)**
lftp tirtho@192.168.99.40: > get test.rpm	190978562 bytes transferred in 1 second **(143.52M/s)**
lftp tirtho@192.168.99.40: > get test.rpm	190978562 bytes transferred in 1 second **(132.27M/s)**
lftp tirtho@192.168.99.40: > get test.rpm	190978562 bytes transferred in 1 second **(120.62M/s)**

Table 8: LFTP-Get TCP Transfer using VSFTPD for IPv6

(TCP) IPv6-IPv6 LFTP-Get TCP Test Results	
Command: lftp -u tirtho,anonymous ::1	**Results: (M/s)**
lftp tirtho@::1: > get test.rpm	190978562 bytes transferred in 1 second **(146.41M/s)**
lftp tirtho@::1: > get test.rpm	190978562 bytes transferred in 1 second **(134.32M/s)**
lftp tirtho@::1: > get test.rpm	190978562 bytes transferred in 1 second **(179.62M/s)**
lftp tirtho@::1: > get test.rpm	190978562 bytes transferred in 1 second **(171.50M/s)**
lftp tirtho@::1: > get test.rpm	190978562 bytes transferred in 1 second **(133.04M/s)**

6. MIMO Gains Interpreted

From experimentation, we notice a boost in throughput using MIMO vs. SISO or traditional Wireless B or G systems which can be interpreted by diversity gains. The application of diversity gains are applied to manage fading paths individually over a time frequency/space domain such that we do not incur bandwidth costs. Here diversity gains can be defined as multiple simultaneous paths for reception and transmission. These paths are combined at either the receiver or transmitter by pre-detection and post-detection combiners in the radio equipment so the effects of fading are lessened. Opportunities exist in MIMO channels that fluctuate independently and interestingly offer reduced amplitude of variability as compared to their SISO counterparts through Space Time Coding. To utilize MIMO technology it is necessary to resort to new transmission strategies referred to as Space Time Channel Codes which in addition to the time and spectral domain use the spatial domain to increase performance of a MIMO system. Based on Orthogonal Space Time Block Codes it is possible to show spatial and temporal redundancy achieving full diversity transforming MIMO channels into a series of SISO channels or multiple highways effectively demonstrating the gains of multi path linear MIMO multiplexing [9, 10, 11, 12, 13].

7. The Appliance

The proposed convergence appliance has the telecommunication exchange encapsulated in software and utilizes IPv6 SIP based P2P communication for distributed end-to-end terminations or SIP phone circuits. To implement this appliance a MIMO Linux based router is required. A Linksys WRT300N MIMO router was utilized connected to a Sipura ATA linked to a POTS phone. This models our appliance in physical form. The Linksys WRT300N MIMO router would provide the MIMO RF connection as well as host Asterisk* telephony engine with a SIP proxy exchange, allowing connection to a telephony interface or ATA. In our proposal we considered a Sipura FXO/FXS 3000 ATA due to its portable nature. The Sipura 3000 allows both a regular POTS phone via RJ11 and Telco Public-Switched-Telephone-Network (PSTN) to be re-distributed digitally through Asterisk* arbiter running on the MIMO router. Furthermore, we can connect our PBX to other PBX's using IPv6 to establish P2P branched connectivity [14] allowing PSTN collocation or forwarding to other members utilizing this appliance to ultimately converge into a Bring-Your-Own-Telephone-Service (BYOTS) approach throughout the Internet. The appliance offers the ability to make P2P SIP based calls to agents registered on the appliance through SIP as well allows the appliance itself to register with other appliances to distribute its telephony services over a network. To connect to the appliance as a client, all that is required is a SIP based ATA or IP Soft-phone capable of registering with the Asterisk* arbiter mediated over a network.

Figure 5, is an overview of the convergence appliance in physical form showing router with voice, data, and video for BYOTS inter-operability.

8. Future Vision: Appliance via Virtualization and Distributed Storage

This paper focused on a software telephony arbiter with modularized connectivity components and observes first order relationships of the appliance over a distributed IPv6 P2P network. If we abstract away from the hardware we can consider the telephony engine as a virtual telecommunications' appliance or virtual machine run through software alone. Considering para-virtualization [15] concepts and using Xen™ an open source virtual monitor [16], it is possible to configure the telephony engine as a virtual appliance stored on flash memory, which can be placed within a persons personal effects. Therefore, as we abstract away from the hardware we are only left with logic transcribed on memory storage. The memory storage as flash memory alone is functionless and inert but coupled with a vessel of computation capable of running the virtual machine turns the personal effects into a highly functional connectivity suite capable of allowing telecommunications from anywhere provided internet access is available. All that is required by the end user is to carry some curtailed peripherals to establish internet connection and the memory module itself to be strategically imposed on a PC or machine able to run the virtual environment reducing the footprint of the appliance to that of a match stick box. To further abstract we can consider the appliance stored on the internet on virtual storage to achieve un-wired un-possessed transparency for communication freedom. This is possible using virtual networkable file systems. An example is Secure-Shell-File-System (SSHFS) based on Virtual-File-System-in-User-Space (FUSE) [17] capable of being mounted on any Internet based computer or computation engine. With SSHFS and FUSE built for any computer or computational engine it is possible to ultimately have the virtual machine hosted on Cyberspace (Internet). To mount the virtual constituents of the storage containing the virtual machine, it is proposed that a universal binary is downloaded via E-mail or Web page to facilitate the execution of the program on the vessel or computational engine. Once the application is executed the user can mount the virtual appliance through software alone to generate the virtual communications appliance. Having the appliance reside on virtual storage allows no requirement to physically carry the appliance thus allowing the communication appliance to be invoked without any need to understand hardware or inner workings of the virtual appliance. Furthermore, updates to the appliance to conform with new emerging technologies or standards are possible by software updates. These include changes to the underlying HMP engine, protocols and arbiter mechanism all dynamically adaptable thus keeping the appliances logical software framework current. The user of this appliance can appreciate this technology at a high level without needing to understand the low level inner workings of the appliance, facilitating greater adoption by creating an intuitive accessibility layer separating the user abstraction from the communication logic.

According to Moore's Law as transistor counts for processors are approximately doubling every 2 years, we are presented with the notion that both ends of a P2P link, edge server side and client side will incur greater computation capacity facilitating greater transmutation of communication hard-logic into reprogrammable communication soft-logic. Virtualization offers relocation of workspace, redundancy and collocation on a single unified framework over heterogeneous systems of different physical composition all accessible through the high speed un-wired Internet. Similarly, virtualization can be applied to communications and IP-based systems. It is a goal for this appliance to build the infrastructure to bring people of all walks of life and variable technical and educational backgrounds to share their physical world into a unified virtualization framework for transparent access to resources and each other in

What-You-Virtualize-Is-What-You-Get (WYVIWYG).

Figure 6, paradigm for virtualization under distributed storage using a pyramid abstraction framework.

9. The Appliance on IPv4

In current IP networks both IPv4 and IPv6 protocols exist. In this paper we only considered IPv6 but it should be noted that it is still possible to implement the appliance under IPv4. IPv4 implementations are complicated by NAT for SIP communication. IPv4 and IPv6 user agents can talk to each other by developing SIP IPv4 to IPv6 translators which allow a SIP agent or client to register with a SIP registrar using either IPv4 or IPv6 [18]. It is also possible to have both IPv4 and IPv6 addresses setup for an edge server containing a SIP registrar to facilitate both IPv4 and IPv6 client access. In this paper we strictly consider IPv6 for implementation of the appliance to avoid complications of NAT. Although, NAT was briefly described, NAT becomes increasingly difficult in large PLANS consisting of virtual IPs as the port limit on a public NAT address is 65535 ports. It is evident that although this is a large set of NAT addressable ports which can reference 65535 machines on a public IPv4 address, this scheme fails in situations where the local area network consists of more than 65535 machines all needing to bind through NAT to a specific public IP. IP-based devices increase the number of devices accessing the Internet and IPv4 is unable to adequately sustain supply and demand for Internet access. IPv4 with NAT complications is not suitable for an evolving Internet aware device landscape. IPv6 is the choice for this appliance (at the time of writing).

IPv6 offers interesting advantages [19, 20, 21] in that stateless IPv6 addresses can allow SIP based dialing. A SIP phone number allows the appliance we have proposed to dial another party either connected to the appliance as a SIP client or the appliance itself connecting to a SIP registrar on another appliance allowing bi-directional P2P connectivity over a single heterogeneous network. This heterogeneous implementation is still possible on IPv4 but because of NAT it is preferable to use IPv6.

10. Conclusions

Communication systems are changing the way we communicate with each other. IPv4 is reaching saturation with 64-bit addressing. A band-aid fix to this problem was the NAT private virtual networks, consisting of virtual private IPs all needing to be mapped to a public IP for Internet accessibility. Unfortunately, NAT is also bottlenecked with the limit of 65535 ports. The need for address space is exhaustive with the IPv4 protocol. New standards are needed to establish P2P communication with mobility in mind. IPv6 was introduced to address the shortcomings of IPv4 and allow for backwards compatibility with existing infrastructure. The IPv6 protocol uses 128-bit addresses and allows for greater connectivity on the Internet without performance loss as compared to IPv4. This paper analyzed the link properties of 802.11b, g and n to find 802.11n as the best choice for mobile wireless connectivity offering the greatest channel capacity and most resiliency to fading due to diversity gains in MIMO architecture. Interestingly, IPv4 showed greater speeds through UDP then IPv6 but when doing actual TCP transfers IPv6 prevailed on a loopback device. If we add NAT complications to IPv4, IPv6 can be seen as a superior protocol for both UDP and TCP. In using codecs it was found that the codecs behave inadequately under low bandwidth situations and thus need to be deployed with the bandwidth considerations in mind. In Experiment 8, it was shown that G729a at 8 kbps frame had the overall best codec performance and is best suited to transverse a WAN for VoIP communication. The consideration of a virtual machine to run the software-based appliance through Xen™ decreases the footprint of the appliance into an embedded storage device.

A further abstraction, of client heavy processing where a functional computer can be packaged as a unit of virtualization stored and executed over the Internet was introduced. This technology relies heavily on the nature of computers and computational capacity growing at proportional rates. As the client and edge heavy P2P terminations evolve and use more computational resources, only the high level outputs are observed virtualized in WYVIWYG. IPv6 with this appliance concept brings us into a world of wireless accessibility at throughput rates equivalent to wired Ethernet with provisions for greater Internet device allocation. The prospect of logical encapsulation of hardware also allows lower cost for market adoption as the physical machine can be seeded on low cost memory, invoked over the Internet or virtualized utilizing recycled hardware. The appliance is therefore manifested and powered by the Internet providing a resource bridge to effectively utilize idle computer computational capacity and transmutes knowledge from our physical world into mobile Internet communication.

References

[1] P. Srisuresh *et al.*, "IP Network Address Translator (NAT) Terminology and Considerations," *RFC 2663*, 1999.

[2] S. Deering and R. Hinden, "Internet Protocol Version 6," *IETF RFC 1883*, Dec 1995.

[3] J. Rosenberg, H. Schulzrinne, G. Camarillo, A.Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: Session Initiation Protocol," *IETF RFC 3261*, Jun 2002.

[4] M. Spencer *et al.*, "Asterisk* Technical Source ***http://www.asterisk.org/***," *Digium Asterisk Code Base*, 1999-2007.

[5] IEEE 802.11b, "IEEE standards ***http://standards.ieee.org/getieee802/download/802.11b-1999.pdf***," *IEEE Standards Association*, 1999.

[6] IEEE 802.11g, "IEEE standards ***http://standards.ieee.org/getieee802/download/802.11g-2003.pdf***," *IEEE Standards Association*, 2003.

[7] R. Nee, V. K. Jones, G. Awater, A. Zelst, J. Gardner, and G. Steele, "The 802.11n MIMO-OFDM Standard for Wireless LAN and Beyond," *Wireless Personal Communications: An International Journal*, vol. 37, no. 3-4, pp. 445–453, 2006.

[8] S. Cheshire *et al.*, "Dynamic configuration of ipv4 link-local addresses," *RFC 3927*, May 2005.

[9] I. E. Telatar, "Capacity of multi-antenna gaussian channel," *European Transaction on Telecom*, vol. 10, pp. 585–595, Nov,Dec 1995.

[10] J. Guey, M. P. Fitz, M. R. Bell, and W. Kuo, "Signal design for transmitter diversity wireless communication systems over rayleigh fading channels," *in Proc. IEEE VTC*, vol. 1, pp. 136–140, April,May 1996.

[11] A. J. Paulraj, D. A. Gore, R. U. Nabar, and H. Bölcskei, "An overview of MIMO communications–a key to gigabit wireless," *Proceedings of the IEEE*, vol. 92, no. 2, pp. 198–218, 2004.

[12] A. J. Paulraj and T. Kailath, "Increasing capacity in wireless broadcast systems using distributed transmission/directional reception," *U. S. Patent no. 5,345,599*, 1994.

[13] A. Sezgin, *Space-Time Codes for MIMO Systems: Quasi-Orthogonal Design and Concatenation*. PhD thesis, University of Berlin, 2005.

[14] T. Narten *et al.*, "Neighbor Discovery for IP Version 6 (IPv6)," *RFC 2461*, Dec 1998.

[15] S. Devine, E. Bugnion, and M. Rosenblum, "Virtualization system including a virtual machine monitor for a computer with a segmented architecture," *US Patent, 6397242*, Oct 1998.

[16] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and The Art of Virtualization," in *SOSP '03: Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles*, (New York, NY, USA), pp. 164–177, ACM, 2003.

[17] Virtual File System in Userspace: FUSE, "Fuse Code Base ***http://sourceforge.net/projects/fuse***," *Sourceforge*, 2004-2007.

[18] R. Campos and M. Ricardo, "Dynamic autoconfiguration in 4g networks: problem statement and preliminary solution," *Proceedings of the 1st ACM workshop on Dynamic Interconnection of Networks*, pp. 7–11, 2005.

[19] D. Barkai, "Technologies for sharing and collaborating on the net," *Proceedings of the 1st International Conference on Peer-to-Peer Computing*, Aug. 2001.

[20] R. Droms, "Dynamic host configuration protocol," *RFC 2131*, 1997.

[21] R. Bush, A. Durand, B. Fink, O. Gudmundsson, and T. Hain, "Representing Internet Protocol version 6 (IPv6) Addresses in the Domain Name System (DNS)," *IETF RFC 3363*, Aug 2002.

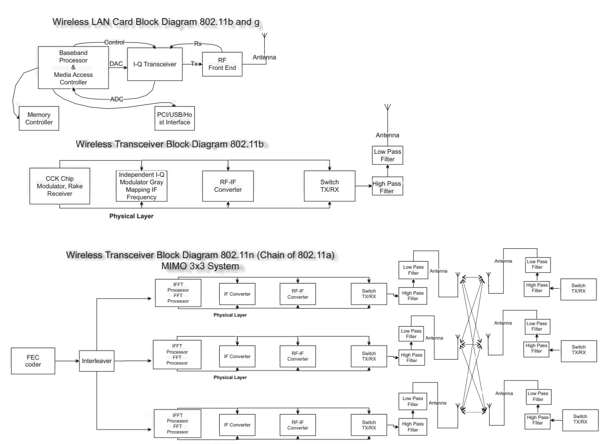


Figure 4. Block Diagrams of Wireless Card, 802.11 Hardware and MIMO Transceiver

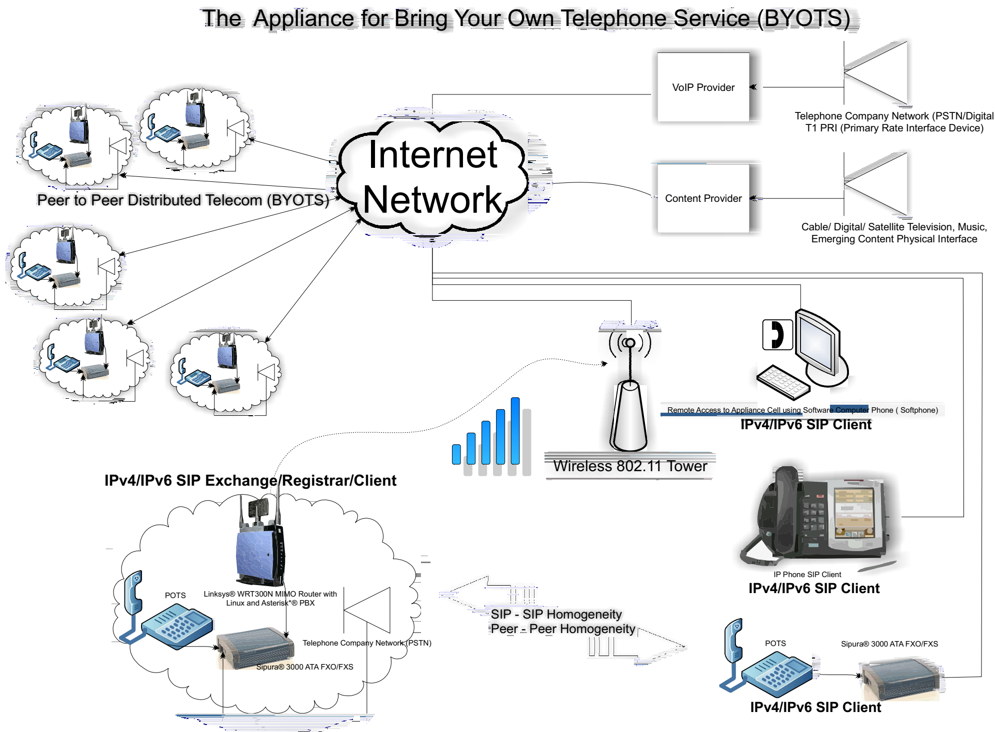


Figure 5. Overview of Convergence Appliance

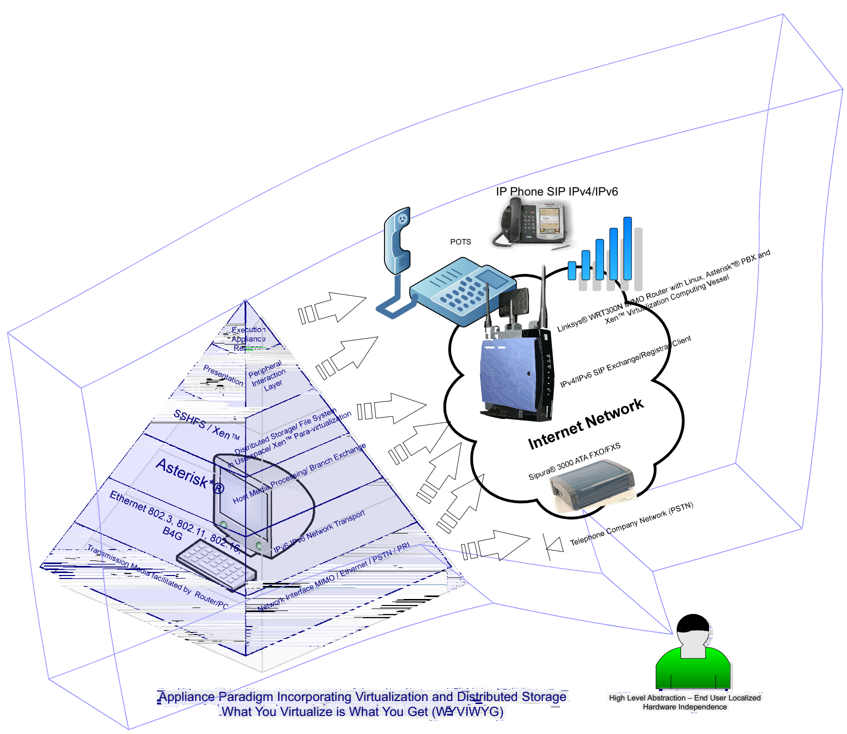
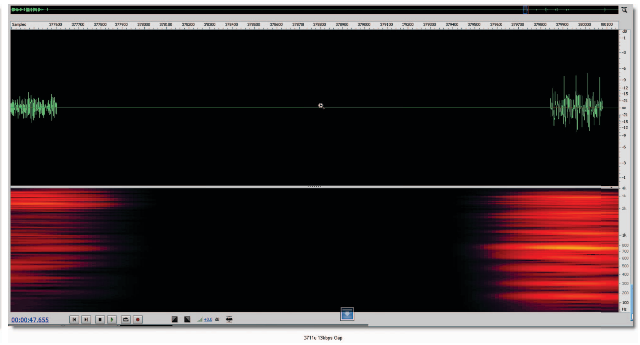
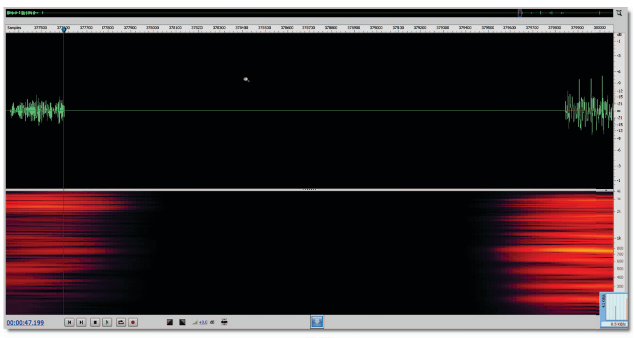


Figure 6. Paradigm for Virtualization using Pyramid Abstraction

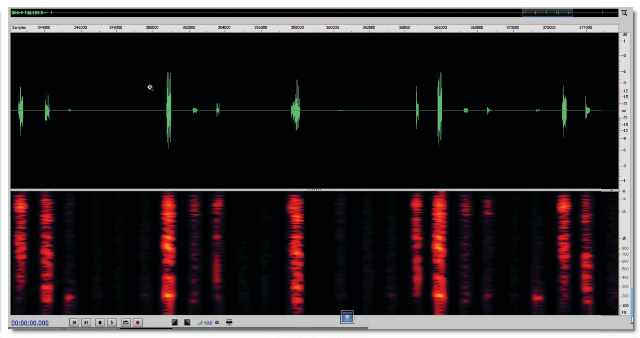
(a) G711u Analysis of 1 Period (Test.)



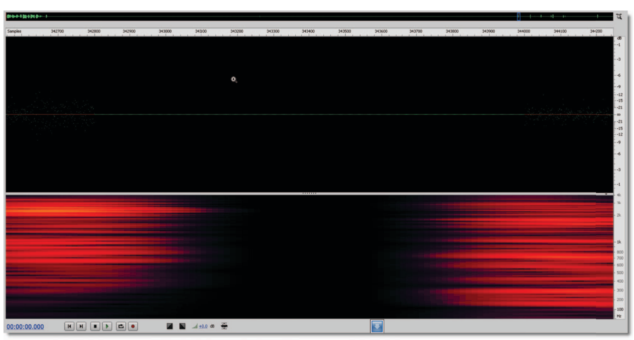
(b) G711u Analysis of 1 Period (Test.)



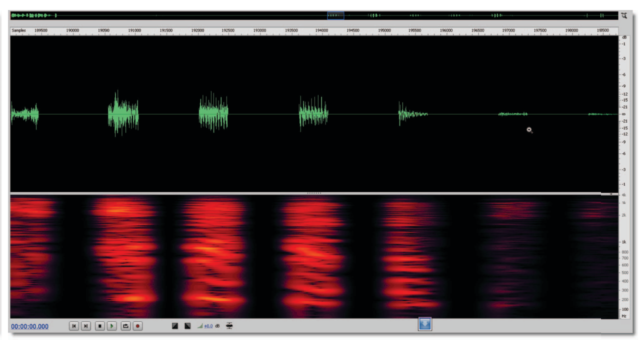
(c) G711u Analysis of 1 Period (Test.)



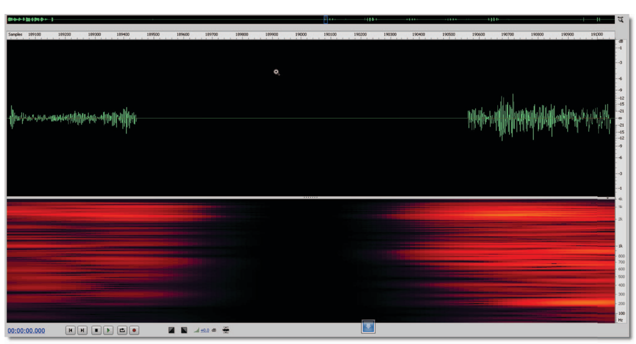
(d) G711a Analysis of 1 Period (Test.)



(e) G711u Analysis of 1 Period (Test.)

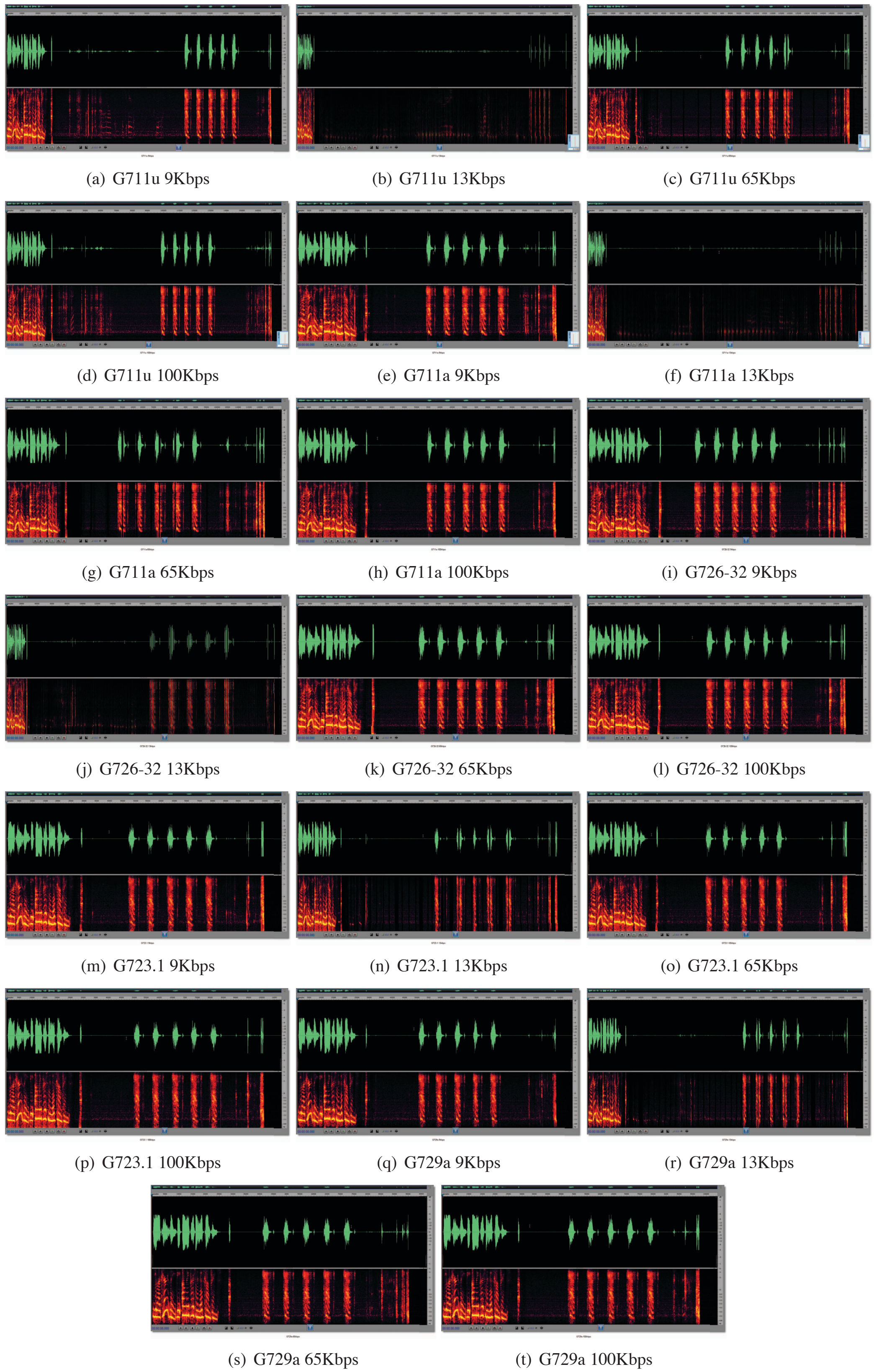


(f) G711u Analysis of 1 Period (Test.)



(g) G711a Analysis of 1 Period (Test.)

Figure 7. Analysis of 1 Period of Word (Test.)



(a) G711u 9Kbps (b) G711u 13Kbps (c) G711u 65Kbps

(d) G711u 100Kbps (e) G711a 9Kbps (f) G711a 13Kbps

(g) G711a 65Kbps (h) G711a 100Kbps (i) G726-32 9Kbps

(j) G726-32 13Kbps (k) G726-32 65Kbps (l) G726-32 100Kbps

(m) G723.1 9Kbps (n) G723.1 13Kbps (o) G723.1 65Kbps

(p) G723.1 100Kbps (q) G729a 9Kbps (r) G729a 13Kbps

(s) G729a 65Kbps (t) G729a 100Kbps

Figure 8. Experiment 8, Codecs Described in Time and Frequency Domain