2007 IEEE International Symposium
on Signal Processing and Information Technology

A Middleware Based Network Hot Swapping Solution for SCA Compliant Radio

Md. Kamrul Hasan, Lenin Mehedy, Salahuddin Muhammad Salim Zabir, Sungyoung Lee, Sang-chul Oh*
Ubiquitous Computing Lab, Department of Computer Engineering, Kyung Hee University, 449-701, Republic of Korea
*Electronics and Telecommunications Research Institute (ETRI), Daejeon, 305-700, South Korea
E-mail: kamrul@oslab.khu.ac.kr, leninmehedy@yahoo.com, szabir@yahoo.com, sylee@oslab.khu.ac.kr, scoh@etri.re.kr

***Abstract*-Software Communication Architecture (SCA) provides a framework for developing Software Defined Radios (SDR). Theoretically, SCA compliant SDRs are supposed to have the capability of self-configuring and switching from one wireless protocol to any other. However, conventional SDRs running standard upper layer protocol stack fail to switch wireless protocols on the fly retaining all the connections and incur large delay. In this paper, we propose a mechanism for wireless protocol switching that is free from these problems. We base our approach on dual or multiple antennas operable by SDR software. The main contribution of this paper is the Hot Swapping Middleware (HSM) that ensures seamless switching between wireless protocols without affecting the application performance. HSM manages IP addressing issues transparently and ensures in order delivery of data packets without loss. Our proposal is free from any requirement of infrastructure support, light weight and easily SCA component upgradeable. Experiments show that our proposal improves the total data transfer of applications in an SDR by more than 25%.**

I. Introduction

With the expansion of different wireless networking technologies, Software Defined Radio (SDR) has been gaining increasing attention in recent years. JTRS has played a pivotal role in the process through their standard Software Communication Architecture (SCA). SCA provides a framework for developing Software Defined Radios (SDR). Theoretically, SCA compliant SDRs are supposed to have the capability of self-configuring and switching from one wireless protocol to another [1-7]. However, conventional SDRs running standard upper layer protocol stack like TCP/IP, fail to switch wireless protocol on the fly without affecting the application performance.

Usually, each wireless protocol switch involves a change in the IP address of the SDR device unless both wireless access points and base stations belong to the same network. A change in the IP address results in a discontinuation of traditional connection oriented protocols like TCP. This problem may be solved by Mobile IP [12-13] or USHA [11] at the cost of infrastructure requirement and large hand off latency. Also, their adoption causes data loss due to large handoff latency.

Infrastructure requirement is expensive and difficult to deploy in practice. On the other hand, large handoff latency causes data loss which ultimately leads to degradation in Quality of Service (QoS) as experienced by the application.

In this paper, we therefore propose a mechanism of network switching for SCA compliant SDR which overcomes the above problems. That is, our proposal ensures replacement of wireless network protocol or network switching without affecting an SDR's operation. This essentially complies with the definition of hot swapping or hot plugging in general. Therefore, we name our proposal as network hot swapping. At the base of our approach lies dual or multiple antennas operable by different SDR software. The main contribution of this paper is the Hot Swapping Middleware (HSM) that ensures seamless switching between wireless protocols without affecting the application performance.

Hot swapping means plug ability of hardware or re-configurability and upgradeability of the software components. Software hot swap infrastructure provides an auto-managed mechanism to reconfigure or upgrade software, component wise, in such a way that the operation of the whole software is not interrupted. It is highly desirable that the downtime to be zero or near to zero.

Software radio brings a band of waveform from antenna and processes that in software [1-7]. Because of the software processing, it is possible to reconfigure a software radio for many different wave forms. It is easily possible to configure a handheld to be a GSM handset or a WLAN receiver. But current software radios cannot do network switching on the fly, retaining all the connections. The radio turns off network activities, configures itself for a different protocol and then starts working again. Although, some similar efforts are found in the literature in the name of vertical hand off [8-13] for non software radios, all of those demand external infrastructure support. Multi-homing techniques [8-10] require transport layer of every intermediary hosts/server to be modified and thus they incur very high deployment cost. Mobile IP based solutions [12-13] deploy home agents and foreign agents to assure zero packet loss. However, they cannot guarantee zero delay. These types of solutions also require infrastructural change that may not be possible to achieve. USHA [11] provides a middleware based solution using IP-tunneling on UDP. It provides seamless network traffic for both horizontal and vertical handoff. This solution seems to be good, but it also needs an external middleware support. Hence those vertical handoff solutions are not appropriate, if we look for a solution confined within the mobile host.

978-1-4244-1835-0/07/$25.00 ©2007 IEEE

So, in this paper, we present a middleware based hot swapping mechanism for protocol stack switching. Our proposed solution needs zero infrastructure support but ensures zero packet loss. We present the mechanism in an SCA compliant radio; because SCA provides a modularized, platform independent radio definition which is indispensable for any hot swapping architecture [5].

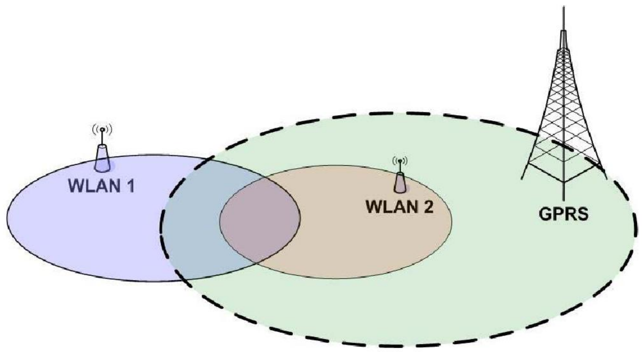

Figure 1: Spatially overlapping networks

We name the protocol stack switching as network hot swapping, characterized by zero packet loss and zero latency gained without any external infrastructure. Our mechanism involves the physical layer, LLC layer and network layer. In this paper, we will discuss only about IP in the network layer to prove our concept. By definition, our solution involves both hardware and software hot swapping. Actually, the only hardware part in a software radio is the antenna. As it is not possible to configure a single antenna to receive two waveforms simultaneously, we propose software radios with multi/dual antenna. Multiple antennas are indispensable for network hot swap capable software radio. We configure two antennas for two different network protocols at the same time, continue the parallel communication for a while, and then turn one protocol off.

We imagine a scenario where several wireless networks are operating and they overlap in places (Fig.1). A user may roam in these networks. While roaming several situations may occur. A user may change the cell of same type of network e.g. WLAN1 to WLAN2, which is called horizontal handoff. Horizontal handoff is easily achievable as the protocol itself provides the handoff mechanism. However, when the user moves to a different network e.g. WLAN2 TO GPRS or GPRS to WLAN2, technical as well as economic criteria come into play. We want to provide seamless network switching in this environment considering the contexts:

- Mobility
- Heterogeneous Networks
- Cost of network connectivity (e.g. GPRS, WLAN etc. may have different usage cost per unit time)

In The later sections we will be elaborating our idea.

II. Our Proposal

Our assumption is that user is communicating over the internet using SCA compliant radio. With the SCA complaint radio, suppose the user was transferring a file over the internet. Now, as the user moves from one network to another e.g. WLAN to GPRS in Fig.1, several situations may occur:

- IP may change (i.e. WLAN to GPRS or vice versa)
- TCP session will be interrupted due to IP change
- Source needs to be informed about destination SDR's new IP
- Loss of packets due to network switching

But we want to provide seamless network transfer. The challenges that lie here are providing *uninterrupted TCP session* and *zero packet loss* without taking support of any *external proxy* or *middleware*. The physical challenge is configuring dual network interface simultaneously which is still unconventional in the current software radios. We propose a multi/dual antenna based SCA compliant radio to solve this issue. We discuss about configuring multiple physical interfaces first and then delve into our solution, because configuring and switching interfaces is basic task of our solution.

A. Multi/Dual Antenna based Software Radio

Multi/dual antenna is indispensable for zero packet loss during Network Hot Swap. Packet loss occurs in SDR due to long delay for configuration of one antenna for network switching. We can continue full-duplex operation through one antenna while configuring the other one for another network. Thus we may achieve zero packet loss during network switching.

Above the antenna we provide *Modem Configuration Module*, which configures the antennas for different waveforms and for different protocols by associating appropriate modulators and demodulators with them. A *Hot Swap Decision Manager* works on top of the *Modem Configuration Module* to decide when to initiate the swap.

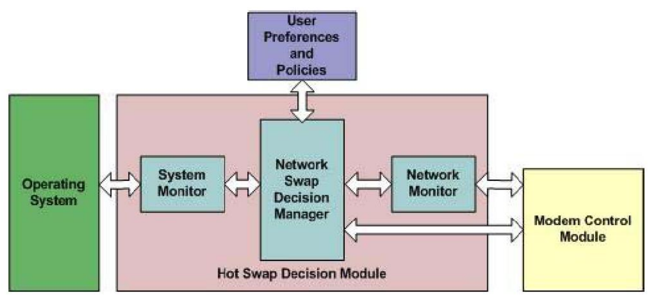

Figure 2: Network Swap Decision Manager

B. Hot Swap Decision Module

As we have discussed in the introduction that we want to provide network hot swapping based on the contexts: mobility, availability of heterogeneous networks and the cost for using the network. So, that way, seamless network switching also becomes context aware, which is a motto of ubiquitous computing. Generally, a network switching occurs when user is moving to some direction and the mobile host usually switches to the best available network. The best available network means, the network with strongest signal coverage and/or network with higher bandwidth.

The switching may be decided based on the cost also. Suppose the user was watching an online video stream on the

WLAN. There was a GPRS network present there. However, due to high cost of GPRS, the user did not use it. But suddenly the WLAN becomes congested or packet error rate increases in WLAN, so the user switches to GPRS.

All these factors can be put in an equation or as the parameters of some policies to provide some context awareness. The context change decision can be made by the user exclusively also. But we think that automated decision making and quick switching is more desirable than explicit user intervention.

Network Swap decision Manager (Fig.2) monitors the interfaces by taking signal strengths and other parameters from Modem Control Module. The manager may take some input from the operating system also. It can have some previous policies set inside it. Based on the previously set option, manager will take a decision to switch the network or may prompt to the user for explicit consent. After the decision is made, manager instructs the Modem Control Module to instantiate a new modem and initializes it.

C. *New Modem Configuration*

After the Hot Swap Decision Manager instructs the Modem Control Module to swap networks, the Modem Control Module initiates a New Modem Configuration (see Fig. 3):

- Modem Control Module instructs the antenna module to free one antenna and antenna 2 is separated
- Modem Control Module configures Modem 2 and Modem 2 in turn configures Antenna 2
- Then Modem 2 acquires an IP address.
- The IP address is in turn passed to Modem Control Module which passes the IP address to upper layer. If the host implements our IP Level Middleware, then the middleware gets the IP and stores it in its mapping table.

The same way any communicating modem can be stopped by the Modem Configuration Module, if the decision comes from the Network Swap Decision Manager. Now we discuss our solution *IP Level Middleware for Hot Swapping*:

D. *IP Level Middleware for Hot Swapping*

We assume that Mobile host (MH) can receive data through multiple wireless protocols simultaneously (i.e. WLAN, GPRS, CDMA). MH can measure channel information for different wireless protocols while receiving/transmitting data and then the user or the Hot Swap Decision Manager decides to change the network. When the MH changes the network, it also changes the IP address. We assume that there is no external infrastructure support. The base station may or may not transmit using multiple interfaces. We say that by deploying our IP Level Hot Swapping Middleware (HSM), the mobile hosts can still switch the networks seamlessly with zero packet loss.

D.1 *Hot Swapping Architecture (Fig.3)*

IP Layer gives the packet to IP Level Hot Swap Middleware (HSM), considering it as an LLC component. IP Level Middleware duplicates the packet for the second interface and gives it to Modem Control Module. So, IP Level Middleware is also an IP Level component to the Modem Control Module (UML diagram in Fig.4 clarifies the object relationships).

Modem Control Module gives the packets to the appropriate interfaces to transmit the packets. When packets come from two different interfaces to the IP Level Middleware, it just looks at the destination port of the packet to decide the recipient application.

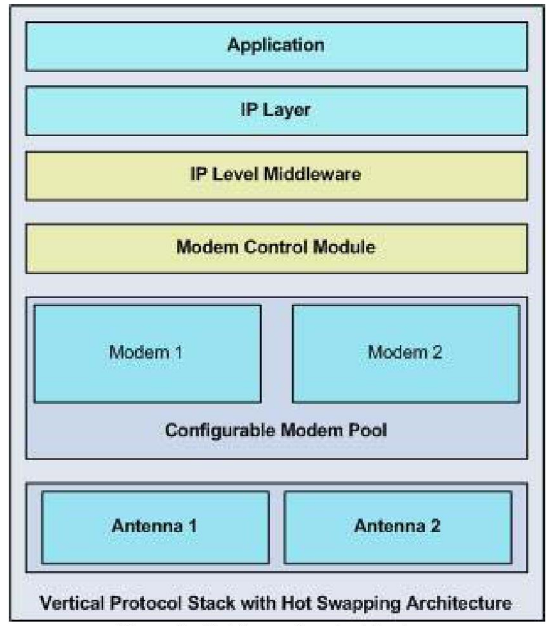

Figure 3: Hot Swapping Architecture

The two parallel interfaces are intended to continue for a while for providing smooth switching. IP Level Middleware acts as a MAC layer to the IP layer and as an IP Layer to the Modem Control Module. Modem Control Module is actually a MAC layer (LLC) specially implemented for application with hot swapping. Applications willing to have network hot swapping should make a protocol stack with usual components plus the Modem Control Module and IP Level Middleware.

In this architectural description, now we describe the flow of intercommunication within the components. The Network Swap Decision Manager instructs the Modem Control Module to instantiate a new modem and it does so. Then a type of handshaking is performed between the IP Level Middleware to make them aware of this switching. After the handshaking, parallel two interfaces are kept open for a while and then the IP Level Middleware decides to stop one modem. Another handshaking is performed between the middleware on this occasion. Then, it instructs the Modem Control Module to stop one interface. We had discussed how a new modem is configured in an earlier section. So, we start from handshaking.

D.2 *Handshaking between the Middleware*

Middleware in SCA Compliant Radio sends a special IP packet to the counterpart. The packet format is simple and can be in the format: ***<Hot swap, parallel ,ip1, ip2>.*** Middleware in the other communicating host, after getting the packet, keeps the IP pair as a hot swapping pair. It then sends an acknowledgement to SCA compliant radio by both interfaces.

Once the handshaking is performed, communication starts. This time packet is sent over both the interfaces. There are two parties in this scenario.

The rest of the IP packets come through the second modem and the IP Middleware replaces ip2 with ip1 in the destination field of the packet before giving it to upper layer.

III. EVALUATION

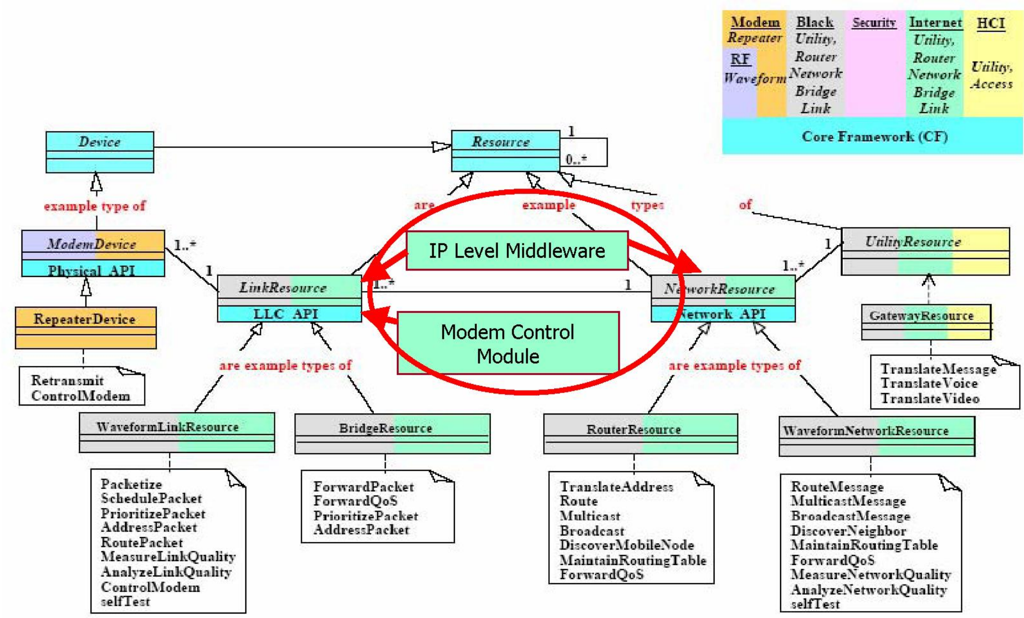

Figure 4: UML Diagram of IP Level Middleware and Modem Control Module

D.3 Packet Sending Scenario: SCA Compliant Radio End

Application gives data to IP Layer to send. IP Layer creates IP packets and gives it to IP Level Middleware. Middleware duplicates the packets (creates another packet with source address replaced by the new/second IP). The packets are then given to Modem Control Module. Modem Control Module gives the packets to appropriate Modems (mapping IP to MAC address). Modems make physical frames and transmit them in the physical media.

The IP Level Middleware in the other radio gets two packets from same source but takes only the first packet and discards the second.

D.4 Packet Sending Scenario: Counterpart

Packets in the second radio's middleware are also duplicated for two different destination IP addresses of SCA compliant radio and then given to lower level protocol stack for transmission. When the SCA compliant radio receives both the packets, it keeps only the first packet.

D.5 Stopping the First Modem

SCA compliant radio's IP Level Middleware decides to stop the first interface and sends a special IP packet: ***<Hot swap, swap, ip1, ip2>.*** The counter part's middleware then sends an acknowledgement to SCA compliant radio through the second interface and stops sending duplicate packets. SCA compliant radio's middleware asks the Modem Control Module to stop the first modem.

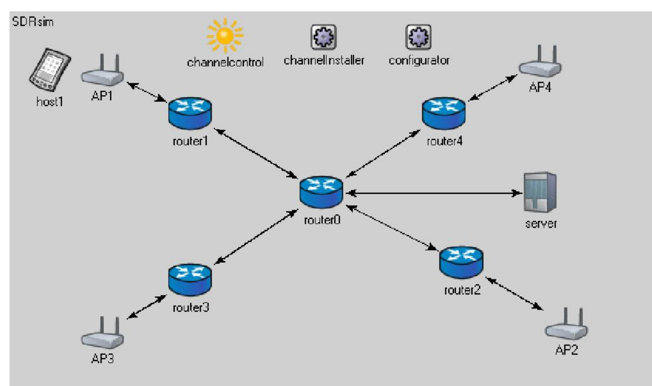

Figure 5: Simulation model for evaluation of our scheme

A. Simulation Environment

We evaluate our proposal through extensive simulations. Through comparisons with the existing conventional SDR, we establish the superiority of our proposal.

In our experiments, we assume a TCP-Reno source at a fixed host or server that transfers bulk data to a software radio enabled mobile host (Fig.5). This mobile host roams within four different wireless networks through four different Access Points or base-stations (AP1 through AP4). Each time the host changes the point of attachment from one AP to another, a handoff occurs. Network switching results in problems like change of IP address and before a handoff takes place; there may be a period of disconnection.

We use the popular OMNET++ simulator [14] for evaluations. The wireless protocols being simulated are IEEE 802.11 or some variants of it – all operating at a data rate of 2 Mbps. Also, in order to avoid simulation complexity without loss of generalization, we simulated the wireless networks and the server as connected to a common switch. Our mobile host takes a random path or direction in this simulated area. Upon reaching the boundary of simulation field, the host changes its direction and starts moving again.

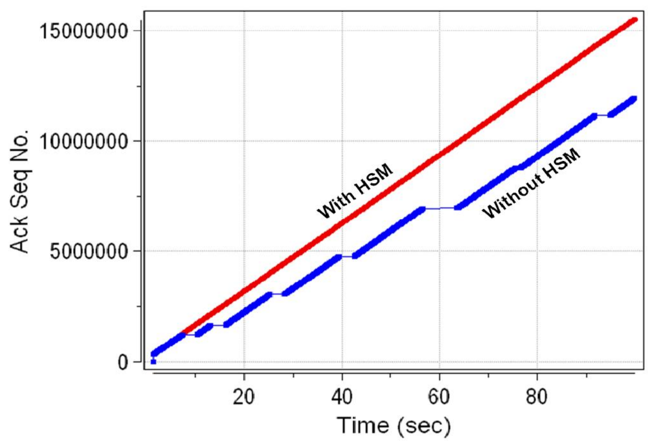

Figure 6: Sequence number of data acknowledged by the receiver

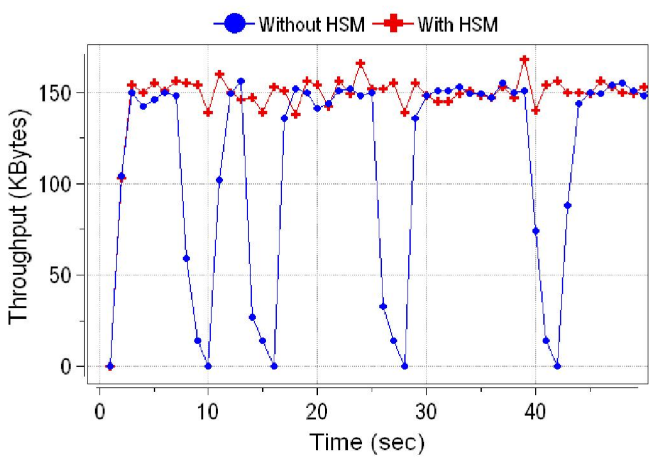

Figure 7: Throughput per each one second interval

B. Performance Metrics

We use the following performance metrics for evaluation of our proposed framework under the simulation environment of section 4.1:

B.1 Acknowledged Data Sequences

This is measured as the sequence numbers of data segments that have been acknowledged by the receiver. In case of TCP, it is simply the segment sequence number in the Acknowledgement field.

B.2 Throughput

Throughput is calculated as the amount of data transferred per unit time. Suppose we transfer m bytes of data in t seconds. Then the throughput of this t second transmission, T is:

$$T = m/t \quad (1)$$

In our results, we show the throughput values for each one second long time slice.

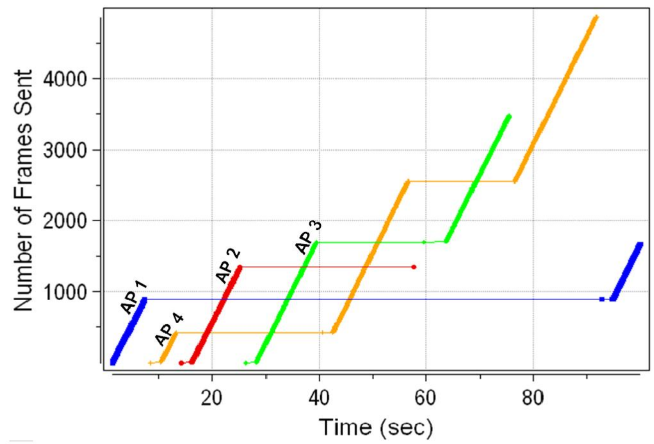

Figure 8: Access point data traffic pattern for conventional SDR

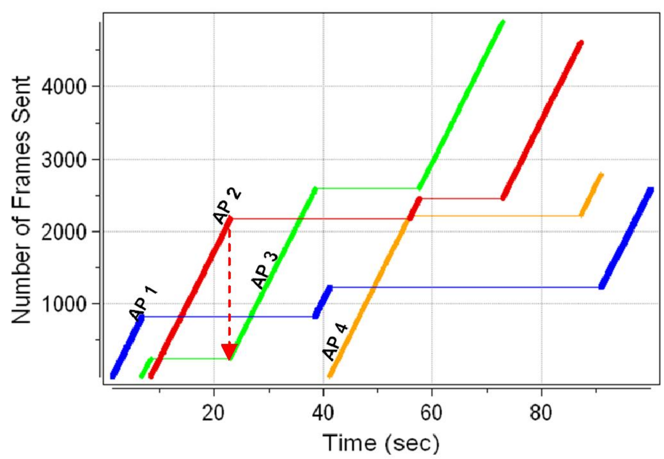

Figure 9: Access point data traffic pattern for multi antenna HSM SDR

B.3 Access Point or Base Station Traffic

This is measured as the pattern of effective data traffic that flows from each access point to the mobile host under the handoff scenarios. We measure this in no of frames for each access points.

C. Simulation Results

We carry out experiments through simulations of a large FTP session from the server to the mobile host. It has been proved in our simulations that with its Hot Swapping Middleware (HSM) based two antenna support, our mobile host SDR continues to receive data even at the occurrence of a handoff. On the other hand, the conventional SDR fails to maintain connectivity during these times that result in a degraded performance. For the sake of a fair comparison, we assume that in conventional approach, the mobile host will adopt some mechanism like mobile IP or USHA [11] in order to continue the session even after a handoff and corresponding change of IP address. We conducted sets of experiments that run for 100 seconds of simulated time.

C.1 Acknowledged Data Sequences

In Fig. 6, we show a comparison between the sequence numbers of acknowledged data during the FTP session using our dual antenna middleware assisted SDR solution and the conventional SDR. We observe that with our HSM, the sequence number curve is a straight line without any breaks. That means, with dual antenna HSM, SDR does not get affected by the handoff. To the contrary, without our HSM, conventional SDR shows breaks in transmission during each handoff due to disconnections. As we can observe from the curves that at the end of this 100 sec simulation, our HSM scheme results in more than 25%, i.e., a significant increase in acknowledged data than conventional SDR.

C.2 Throughput

Fig.7 shows variation of throughput with time. Here also, we observe that dual antenna HSM supported SDR maintains a consistent average throughput without any significant degradation at any time. To the contrary, conventional scheme, although, sometimes maintains an average throughput value, it degrades to zero throughput at times of handoff. The reason is, during handoff related disconnection, there is no transfer of data. This ultimately leads to a low yield in total or overall throughput.

C.3 Access Point or Base Station Traffic

Fig. 8 and Fig. 9 show the effective traffic pattern at different APs before and after handoffs in cases of conventional SDR and dual antenna HSM SDR respectively. Here again, we observe that in conventional case, no AP transfers data traffic to the SDR during handoff. While in the case of HSM SDR, there is no break in effective traffic. That is, whenever there is a switch of networks, the new AP starts transmitting data immediately. For example, in case of the handoff at 25 sec between AP2 and AP3, Fig. 8 shows some gap for AP3 to take over. However in Fig. 9, AP3 starts functioning immediately.

IV. CONCLUSION

In summary, we provide a solution for network hot swapping for SCA complaint radio. We also incorporate context awareness in taking the swap decision. The basis of our solution is software radio with multiple (e.g. double) antennas. Multiple antennas enable SDR to have a standby antenna that can be configured to a different protocol while the other antenna is still operating using the current communication protocol. Thus, we achieve zero configuration time and zero packet loss during network switching. We exclude the requirement for external proxy or middleware by including an IP Level Middleware for Hot Swapping. Simulation results show that our Hot Swap Middleware effectively solves the network hot swapping problem.

V. ACKNOWLEDGEMENT

This work was partially funded by Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea; MIC (Ministry of Information and Communication), Korea, under the ITFSIP (IT Foreign Specialist Inviting Program) supervised by the IITA (Institute of Information Technology Advancement) and MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2006-C1090-0602-0002).

REFERENCES

[1] J. Mitola, “The Software Radio Architecture,” IEEE Commun.Mag., vol. 33, no. 5, May 1995, pp. 26–38.

[2] M. Cummings and S. Heath, “Mode Switching and Software Download for Software Defined Radio: The SDR Forum Approach,” IEEE Commun. Mag., vol. 37, Aug.1999, pp. 104–6.

[3] M. Mehta et al., “Reconfigurable Terminals: An Overview of Architectural Solutions,” IEEE Commun.Mag., vol. 39, Aug. 2001, pp. 82–89.

[4] N. Nakajima, R. Kohno, and S. Kubota, “Research and Developments of Software-Defined Radio Technologies in Japan,” IEEE Commun. Mag., vol. 39, Aug. 2001, pp.146–155.

[5] SCA specification, JTRS-5000, SCA V3.0, Aug 27, 2004.

[6] Enrico Del Re (Ed), Software Radio Technologies and Services, Springer, ISBN 1-85233-346-4.

[7] S. Hykin, Cognitive Radio: Brain-Empowered Wireless Communications, IEEE Journal On Selected Areas In Communications, Vol. 23, No. 2, February 2005.

[8] A. Matsumoto, M. Kozuka, K. Fujikawa, and Y. Okabe, “TCP Multi-Home Options,” draft-arifumi-tcp-mh-00.txt, IETF Internet draft, Oct.2003.

[9] A. C. Snoeren, “A Session-Based Approach to Internet Mobility,” PhD Thesis, Massachusetts Institute of Technology, December 2002.

[10] R. Stewart et al. “Stream Control Transmission Protocol,” RFC 2960, IETF, Oct. 2000.

[11] L. J. Chen, T. Sun, M. Gerla ,”USHA : A Practical Vertical Handoff Solution”, Mobiquitous 2004.

[12] S. Sharma, I. Baek, Y. Dodia, T. Chiueh, “OmniCon: A Mobile IP-based Vertical Handoff System for Wireless LAN and GPRS Links”, LISA 2004.

[13] Mark Stem and Randy Katz. “Vertical Handoffs in Wireless Overlay Networks”. In ACM Mobile Networking (MONET), Special Issue on Mobile Networking in the Internet, 1997.

[14] OMNeT++ Simulator at http://www.omnetpp.org/.