International Journal of Fuzzy Logic and Intelligent Systems, vol. 8, no. 4, December 2008 pp. 239-247

A Framework for Universal Cross Layer Networks

Murad Khalid*, Ravi Sankar*, Young Hoon Joo and In Ho Ra****

*** Department of Electrical Engineering, University of South Florida**
**** School of Electronic and Information Engineering, Kunsan National University**
E-mail: mkhalid@mail.usf.edu, sankar@eng.usf.edu, yhjoo@kunsan.ac.kr, ihra@kunsan.ac.kr

Abstract

In a resource-limited wireless communication environment, various approaches to meet the ever growing application requirements in an efficient and transparent manner, are being researched and developed. Amongst many approaches, cross layer technique is by far one of the significant contributions that has undoubtedly revolutionized the way conventional layered architecture is perceived. In this paper, we propose a Universal Cross Layer Framework based on vertical layer architecture. The primary contribution of this paper is the functional architecture of the vertical layer which is primarily responsible for cross layer interaction management and optimization. The second contribution is the use of optimization cycle that comprises awareness parameters collection, mapping, classification and the analysis phases. The third contribution of the paper is the decomposition of the parameters into local and global network perspective for opportunistic optimization. Finally, we have shown through simulations how parameters' variations can represent local and global views of the network and how we can set local and global thresholds to perform opportunistic optimization.

Key Words : cross layer framework, optimization cycle awareness optimizer, system awareness, user awareness,

1. Introduction

ROSS layer approach has undoubtedly proven itself as a promising C step forward in wireless network performance optimization solutions. Recent trends in wireless networks allow users with heterogeneous service requirements to communicate effectively in a dynamic resource-limited environment. Each user has their own set of end-to-end Quality of Service (QoS) requirements that the wireless network must satisfy. To cater to multiple user service requirements every bit of available resource has to be used in an optimal manner. The quest for optimization consequently leads to establishing and harnessing the richer interactions between the OSI layers of the communication stack. Each layer of the OSI protocol stack has to perform a set of specific functions and depending upon the user service requirement each function needs to adapt based on the information from the other layers. For instance, if the mobile user in a centralized wireless network has a stringent throughput requirement; the Medium Access control (MAC) layer can dynamically adapt the modulation and coding based on the channel feedback from the Physical (PHY) layer to optimize user's throughput. The Network (NET) layer, in turn, may assign a channel so that the interference is minimized for the user on that channel. Another example that has been cited in numerous papers ([1]-[4]) is that of the TCP window reset in Transport (TRAN) layer. It is shown that the TCP window size resets to unity when the signal fading results in packet errors. The TCP misinterprets ACK delay from retransmissions due to signal fading as the sign of congestion in the network and consequently resets its transmission rate window. It is shown that if the TCP gets insight into congestion by using the Explicit Congestion Notification (ECN) bit or the Explicit Loss Notification (ELN) bit, then it can easily distinguish the actual cause of delay and, therefore, avoid resetting its transmission window. This clearly depicts that many layers can interact concurrently and exchange their protocol variables to squeeze out better performance in every possible way to achieve the optimal performance goal.

Many wireless networks like Ad hoc networks, Sensor networks and Third-Generation (3G) cellular networks require real time adaptations for dynamic network conditions and changing user requirements. Furthermore, recent development in multi-antenna and multi-packet capture technologies requires the OSI layered architecture to be flexible enough to fully utilize the potential of the above technologies.

Indeed the temptation of using the cross layer interaction for performance optimization is irresistible. However, the long term consequences of unbridled cross layer interaction schemes in a heterogeneous wireless environment, can lead to a chaotic collection of disparate cross layer techniques that may not interoperate flexibly. This obviously requires us to think how far we want to go beyond the layered architecture or how much of a trade-off is acceptable so that the performance and

---

Manuscript received Sep. 1, 2008; revised Dec. 05, 2008.
This work was financially supported by the Kunsan National University's Long-term Overseas Research Program for Faculty Member in the year 2006.

interoperability objectives are achieved. So far, the research community has mainly focused on interactions between various combinations of non-adjacent layers to meet specific performance goals. Thus, to date there are still a number of unresolved questions in this realm of cross layer optimization and many issues that remain unexplored. Some significant open questions are related to the monitoring of the parameters that trigger interactions; type of trigger event; statistical significance of the trigger; trigger relationship to performance goals; generation and control of interactions; standardization of interactions; and the type of interactions.

In this paper we propose a universal cross layer framework for distributed wireless networks that addresses the aforementioned questions and at the same time upholds the sanctity of modular architecture.

2. Snapshot of Related Work

Generally, cross layer research contributions can be categorized into cross layer design surveys ([1], [2], [4], [5]) design coupling approaches ([3], [6]-[13]) cross layer architecture for information sharing ([14]-[15]), and cross layer interaction characterization and modeling ([16]-[18]). A broad level definition and classification of cross layer designs, type of cross layer couplings and challenges related to cross layer design were presented in [1]. However, it only provides high level glance into cross layer architecture and framework. The authors in [2] highlighted the importance of reference architecture with respect to long term performance goals, ease of modification, stability and reliability. It convincingly points out that cross layer approach to optimization can lead to unintended conflicts between various performance requirements of different layers. Furthermore, cross layer modifications are not easy and require understanding of interactions between parameters, as such, tracing and debugging code becomes difficult. A survey of cross layer optimization solutions related to third-generation (3G) wireless mobile networks was presented in [4]. The paper adopts a vertical layer approach with security, QoS, mobility and link adaptation layers as the vertical layers that are visible to the standard OSI layers. It characterizes various types of interactions and signaling into inter-layer, intra-layer control messages within the host and between different hosts. However, it does not discuss existing cross layer frameworks that control and manage all the interactions and signaling in a stable manner. An exhaustive survey of cross layer approaches related to sensor networks can be found in [5]. Papers reviewed are based on design coupling approaches, where pairs of PHY, MAC, NET and TRAN layers interact to optimize certain performance goals. However, it lacks any discussion on generic cross layer framework architecture.

A small set of contributions related to cross layer design coupling approach is presented in [6]-[13]. In [6], PHY layer capture and multi-packet reception capability in conjunction with MAC layer to improve throughput and delay performance of the wireless network is presented. In [7], a PHY-MAC cross layer interaction is exploited to optimize throughput and delay. It introduces a new multiple-input-multiple-output (MIMO) based MAC protocol extension to 802.11a that enables multi antenna terminals to communicate with each other in multiplexing mode and/or diversity mode in a low correlation wireless propagation environment. In [8], cross layer interaction between MAC and NET network layers is exploited. It proposes throughput increase via interference based routing protocol that chooses routes with minimal interference using spatial proximity of transmitters. In [9], a design coupling approach using PHY, MAC and NET layer interactions for a centralized multi-hop wireless network is developed to minimize end-to-end latency. In [10], a PHY-MAC cross layer interaction is used to optimize end-to-end throughput and energy efficiency for a set of pre-defined routes in a multi-hop wireless network. The authors in [11] propose the MAC, NET and TRAN layer interactions to optimize the throughput based on congestion price input. The algorithm is based on the congestion metric to reduce congestion which impacts the throughput of a multi-hop route. A cross layer approach to optimize network lifetime, delay and energy consumption by jointly using routing and scheduling in a TDMA wireless ad hoc network is presented in [12]. The scheme is based on utility function, optimization, which is defined as a weighted function of queue length, transmission power and node's utilization. The authors in [13] propose a cross layer approach to minimize energy consumption for transmission by joint scheduling, routing and using power control in a fixed TDMA wireless network.

In [14], [15] and [19] a cross layer information sharing approach, in which modular architecture is preserved, is presented. A cross layer framework for multimedia application in Ad Hoc networks can be found in [14]. The Framework is based on interaction between the middleware and the routing layers only. Middleware layer provides the QoS requirement to the routing layer for optimal route selection and the routing layer shares node location information with the middleware layer to provide data availability information. However, it lacks generic mechanism for controlling and managing interactions. In [15] cross layer framework is generic and very close to our proposed framework. It is based on local and neighbor state information which are then collectively used in optimization based on performance criteria. In [19], another information sharing architecture in which a Network Status Layer is used as a repository to share parameters between layers is proposed. In contrast to the aforementioned contributions, our paper defines a vertical layer architecture which is much more intelligent than just being a repository, with general functionalities as awareness parameters monitoring, mapping, classification and stochastic optimization using local and global network perspectives.

Characterization of interactions and their inherent stochastic behavior provide invaluable insight into protocol design and cross layer adaptation. But, unfortunately other than the contributions made available in [16]-[18], there is very little to be found. In [16], cross layer interaction models and the cross layer interaction arrays for three broad classes of cross layer atomic actions are defined. It provides a comprehensive list of interactions for various types of events and triggers. However, it does not address interaction management and control to achieve the required QoS. In [17], the authors interestingly point out that the stochastic behavior of the network is a combination of protocol dynamics and statistical behavior. Using methods insensitive to the correlation between dynamical and stochastic behavior, network performance characteristics can be accurately predicted to effectively use cross layer optimization. An exhaustive analysis of interactions between the MAC and the NET layers under varying packet injection rate, node's speed and mobility conditions was given in [18]. Results clearly show that under varying conditions different combination of protocols must be invoked at MAC and NET layer for optimal performance.

Despite many years of contributions in cross layer realm, not much consolidation is seen in terms of cross layer framework standardization and interoperability. This paper is an attempt in that direction to develop a universal cross layer framework that can co-exist with the legacy systems and maintains modularity and stability.

3. Proposed Universal Cross Layer Framework

In this section, we propose a vertical layer cross layer framework as shown in Figure 1. The vertical layer in the context of our paper is not just a repository for the standard OSI layers to share parameters, but provides a complete parameter exchange between layers in an intelligent and controlled manner. This further does not require major modifications to the existing protocols or the need for a new protocol design. A minor modification to the existing protocols is required to create control and data information path between the horizontal and the vertical layers. The main idea is to design the vertical layer independent of the horizontal OSI layers and control the interactions through this intelligent vertical layer. Thus, the vertical layer can be thought of as the control engine (brain) behind the cross layer interactions which makes this framework modular, intelligent and adaptive. Some salient functions of the vertical layer framework are described as under.

1. Vertical layer is aware of the outside network(s) state, aware of the user, and aware of the internal state
2. Vertical layer communicates with the horizontal layers through V-SAP (Vertical Service Access Points) primitives
3. Vertical layer monitors interactions between horizontal layers
4. Vertical layer activates or deactivates any combinations of protocols in horizontal layers
5. Vertical layer monitors and changes protocol parameters depending upon outside network state, user state, and system state
6. Vertical layer performs stochastic optimization to make accurate adaptations
7. Vertical layer uses Awareness Knowledge and Policy Database to determine network type, policies, and performance goals; select appropriate protocols, and optimize performance
8. Vertical layer learns and updates Awareness Knowledge and Policy Database

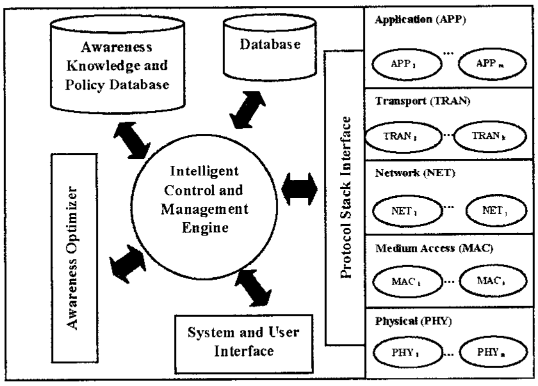

Fig. 1. Universal Cross Layer Framework

The aforementioned vertical layer functions provide for a generic framework to adapt if the network environment changes or if a new service network becomes available. The following subsections further expound on the proposed vertical cross layer framework.

A. Awareness Descriptors and Parameters

In this section we introduce awareness parameters related to vertical cross layer framework as shown in Table 1. Awareness parameters are like state variables that comprise of many parameters as defined in the summary column of Table 1. The awareness parameters contain information about the local and global view of the network dynamics, channel and environment information, user requirements and behavior information, and internal system information. This information has to be extracted from the awareness parameters and mapped to awareness descriptors for proper information classification and database storage. Awareness descriptors are used by the awareness optimizer for decision making. The idea of awareness is mainly derived from cognitive radio domain [20]. However, awareness introduced in [20] is primarily focused on location awareness related information. We know from wireless network environment that location awareness is just one piece of useful information in addition to other awareness descriptors as introduced below.

1. System Awareness
2. User Awareness
3. RF Awareness
4. Protocol Stack Awareness
5. Network Awareness
6. Application Awareness
7. Topographical Awareness
8. Meteorological Awareness

Systems awareness used in the framework provides useful information about the node's self awareness aspect. It may include battery life, bugs in the system, looping in the system, RF capabilities and limitations. In many network applications battery life is extremely important, since it has to be considered along with other performance goals and awareness monitoring. The system awareness descriptor encompasses energy and system interrupt awareness parameters as shown in Table 1. User awareness represents information regarding user behavior in time and location, and user generated commands. RF awareness provides information gathered from the PHY layer of the protocol stack. It contains RF state and Antenna mode awareness parameters. Protocol stack awareness defines monitoring of MAC, NET and TRAN layers' parameters and messages for performance optimization. Network awareness represents the type of network service available in the area, mode of service and capabilities, and security and encryption policies of the network. Application awareness contains information about user application requirements and its security and encryption constraints. Topographical awareness and meteorological awareness provide information about the terrain environment and climatic condition relative to the user location.

Awareness parameters contained within the awareness descriptors can be compared with the information stored in the knowledge and policy database to generate awareness descriptors which are used for optimization decisions. For example, RF and topographical states are compared to a pre-stored thresholds in the knowledge and policy database to determine if the RF environment is urban with heavy vegetation, dense urban with sparse vegetation, suburban with dense vegetation, and rural. Furthermore, antenna mode awareness parameters define single antenna or multi-antenna configurations. The resultant RF awareness descriptor defines the channel characteristics and the antenna configuration, but does not provide any information about the climatic condition which can have detrimental propagation effects. Thus, in order to make an intelligent decision multiple awareness descriptors have to be jointly analyzed for performance enhancement.

B. Functional Architecture

The proposed cross layer framework is based on intelligent vertical layer architecture for interactions control and management. The vertical layer consists of six functional blocks.

- Intelligent Control and Management Engine (ICME)
- Protocol Stack Interface
- System and User Interface
- Database
- Awareness Knowledge and Policy Database
- Awareness Optimizer

The ICME is the brain of interactions control. It is primarily responsible for monitoring all awareness parameters, monitoring interactions between OSI layers, selecting protocols and tuning protocol parameters based on awareness optimizer decisions. The ICME communicates with the system, user and the OSI layers via the Protocol Stack Interface and the System and User Interface, respectively. Protocol Stack Interface has the capability to receive and send any type of information from and to any protocol in a particular layer within the OSI protocol stack, respectively. Since, the communication can be with multiple layers at a time, therefore, V-SAP packet format needs to be defined with proper address, control and data sections. However, the interface definition is not within the scope of this paper. The vertical cross layer framework contains two logical databases. The Awareness Knowledge and Policy Database contain pre-stored thresholds and cases to be used for decision making. For example, for Network Awareness it may contain information about different types of networks, available services and modes of operation, security policies, and relevant protocols. As another example, User Awareness may contain information about user's time and location trend and user's application services history. This can prove very useful in pre-emptive adaptations where network dynamics change rapidly. The last functional block is the Awareness Optimizer which uses awareness parameters for optimization decisions. The Awareness Optimizer will be explained later in this paper.

C. Optimization Cycle

As explained before, the main purpose of the vertical cross layer framework is performance optimization. The modularity and standardized interaction between the OSI layers are maintained in this framework (see Fig. 1). The optimization performed by the vertical cross layer framework is based on three phases as depicted in Figure 2. The first phase is the monitoring phase, where all the internal and external events are constantly monitored. The rate of monitoring will actually depend on the application. Interested readers are referred to [4] and [17] for a non-standardized list of interactions and events. In the monitoring phase, all awareness parameters are gathered in real time and stored in the database. The events trigger the optimization phase, where all the awareness parameters and awareness descriptors are classified and jointly analyzed. The comparison with the pre-stored cases and thresholds and optimization approach leads to decision output. The decision output triggers adaptation phase. In adaptation phase, the parameters belonging to the respective protocols and layers are tuned for performance enhancement. After the adaptation phase the optimization cycle repeats the monitoring phase. As seen in Figure 2, awareness learning phase represents updating of the Awareness Knowledge and Policy Database if new cases or thresholds are learned during the optimization cycle. It must be kept in mind that awareness learning phase is not where the optimization cycle ends. Rather, it is invoked in parallel with the adaptation phase for learning purpose only.

D. Awareness Optimizer

In this section we introduce the functional block architecture of the Awareness Optimizer as shown in Figure 3. We propose three functional blocks within the Awareness Optimizer. The block shown in dotted line merely illustrates interaction with the Awareness Knowledge and Policy Database and is not part of the Awareness Optimizer. The function of Awareness Mapper is to map awareness parameters into awareness descriptors. For instance, Awareness Mapper may extract information about the antenna operating in a multiplexing mode and the RF environment as the dense urban and utilize it as the RF awareness descriptor. This RF awareness descriptor is used by the classifier to generate a local view of the network. The Awareness Classifier uses the awareness descriptors to extract performance goals, constraints, local and global view of the network. The local view comprises awareness parameters that provide information about the single hop neighbors where as global view comprises awareness parameters that provide information over multiple hops in the network. It must be understood that the local and global views make sense in a distributed network environment. The Stochastic Awareness Analyzer block analyzes stochastic behavior of the local and global views of the network. The purpose is to determine the time scale of local and global perturbations. Then based on the local and global views; user behavior, awareness parameters, goals and constraints; the Stochastic Awareness Analyzer uses some form of optimization technique (e.g., heuristic optimization) to make decisions for the choice of protocols and parameters.

4. Universal Cross Layer Framework Optimization

As emphasized before that vertical cross layer framework relies on Awareness Optimizer and Awareness Knowledge and Policy Database to make intelligent cross layer decisions. The computational complexity of the Awareness Optimizer can be reduced by pre-storing thresholds, cases and precedence. To clarify further, the pre-stored cases should address some of the following questions.

- Is the RF environment urban, suburban or rural?
- What should be the antenna mode?
- What type of network service is available and what are its policies?
- What are the security requirements for the network service?
- Which protocols to invoke for available network services?
- What are the user's next location and service need?
- What are the goals and constraints for the required applications?
- What should be preferred modulation and coding for the required goals and constraints?
- Is it a local or global perturbation?
- Is the local or global perturbation critical or significant?
- What should be adapted in case of particular local or global perturbations?
- What should be adapted in case of medium or low battery energy?
- What applications type to be switched in case of certain type of local or global perturbations?

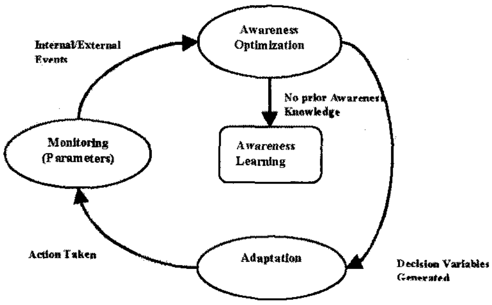

Fig. 2. Optimization Cycle

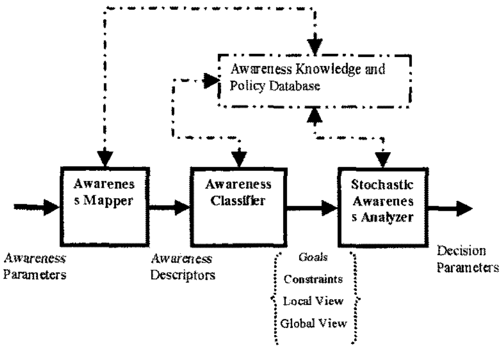

Fig. 3. Awareness Optimizer

It is worth mentioning that multi-objective optimization techniques can be used to enhance performance, but at the expense of increased computational complexity. Although, discussion on optimization methods is beyond the scope of this paper; yet, we believe that pre-stored thresholds, cases and scenarios can reduce computations and conflicts in complex scenarios.

A. Simulation Scenario for Ad Hoc Network

Through simulation of an ad hoc network scenario, we illustrate how the Awareness Optimizer (described in section III, see Fig. 3) can make a decision about the criticality of local and global perturbations. The local and global perturbation information can be obtained through implicit or explicit messages as briefly discussed in [4]. Subsequently, the Awareness Optimizer can isolate the stochastic perturbations into local or global perturbations and make appropriate decisions based on the goals and constraints.

The simulation was performed in QualNet 4.0 environment. In the simulation 100 nodes were placed in a grid format in an area of 1500x1500m2. A single File Transfer Protocol (FTP) multi-hop session was established between the source node and the destination node as designated by 'S' and 'D' letters in Figure 4. The routing protocol used was Dynamic Source Routing (DSR) and the MAC protocol was the standard IEEE 802.11b. The simulations were conducted in four separate steps. In the first step, only FTP sessions were activated with no mobility and without any local or global interference. In the second step, only local interference was created by creating independent sessions as designated by 'IL.' The effects on parameters were observed at the source node 'S.' In the third step, only global interference was created by creating independent sessions as designated by 'IG.' The global interfering nodes were placed in such a manner so that they interfere with intermediate hops of the FTP session. This global interference has no direct effect on the link at the source (S) or the destination (D) node. The resulting effects on parameters were observed at the source (S) and the destination (D) nodes. In the fourth step, mobility was introduced along with the local and global interferences. Random waypoint mobility model was used with a pause time of 30 seconds and a random speed of 0-10 meters/second. The observed parameters were averaged over ten independent runs for each step, where each run is for the duration of 600 seconds.

The parameters were observed at the MAC, NET and TRAN layers. As shown in Figures 5(a)-(d), a select number of parameters for the above four steps are picked to illustrate the global and local perturbations. It is clear from Figures 5(a)-(d), when local interference (Local INTF) is introduced, RTS transmissions and packet retransmissions at MAC layer show a significant change compared to DuPACK (DuPlicate ACK), RERR (Route ERRor) and Time in Queue parameters at TRAN and NET layers. This means that a node can sense its local neighborhood environment based on MAC parameters. On the other hand, when only global interference (Global INTF) is introduced, DuPACK, and RERR parameters do not show any significant change compared to no interference (No INTF) situation. However, Average Time in Queue parameter shows a significant change. This means that a node can get good indication of global interference through this parameter. This information can be explicitly communicated through piggy backing, or through control protocol like Internet Control and Message Protocol (ICMP). However, explicit communication through control protocols increase network load as observed during the simulation (due to scope irrelevance results are not shown). It must be understood that if local or global interference is increased by adding more interfering nodes the absolute values shown in Figures 5(a)-(d) will change, but the general behavior will remain the same. Moreover, if local interference increases the source node will get an indication of link contention at the MAC level, but at the source node the Average Time in Queue at NET level will also be affected. An explicit means (i.e., in-band or out-of-band control information) can be used to get the Average Time in Queue and other parameters' information regarding the global situation. It is also obvious from Figures 5(a)-(d) that mobility and interference combined (Mobility + INTF), affects all the parameters appreciably. This implies that correlating changes in the parameters can give a good indication of mobility at the local or global level. In essence, the point we are trying to make is that the local perturbations that affect parameters at the NET and TRAN layers can be separated from the global perturbations at the NET and TRAN layers. As stated before, the Awareness Optimizer can use this separated local and global views to make intelligent decisions.

B. High Level Optimization Strategy

At this juncture, we propose a high level strategy for the Awareness Optimizer to optimize system performance. We define global adaptation as parameters analysis and steps taken at layers above MAC layer that effect the entire session of the source node. Further, we define local adaptation as parameters analysis and steps taken at MAC and PHY layers that affect the local neighborhood of the source node. For single hop session, both local and global adaptations affect the entire session. Let P and PL represent total and local perturbations; THl and THg represent local and global thresholds; and Al and Ag represent local and global adaptations, respectively.

The strategy for optimization is as follows:

IF $\{P_L < T_{Hl}\}$ and $\{(P\text{-}P_L) < T_{Hg}\} \rightarrow$ {Initiate A_l};

ELSE {Initiate A_g}.

The validation and proof of the actual effectiveness of the proposed strategy has to researched in depth.

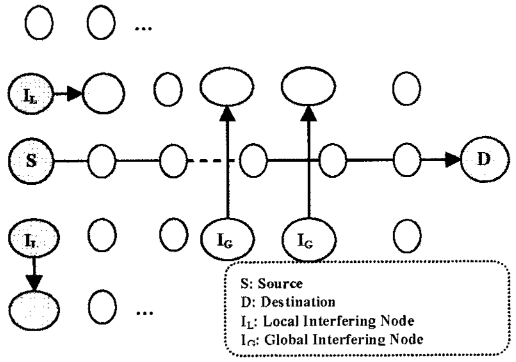

Fig. 4. Ad Hoc Network Scenario

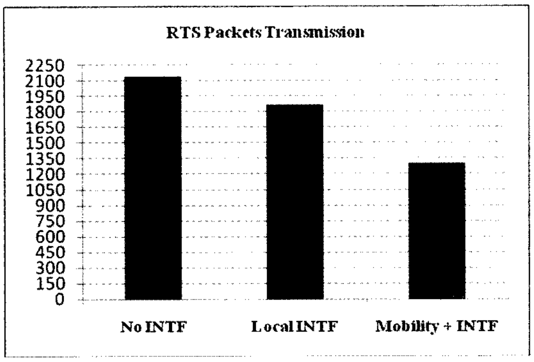

Fig. 5 (a). Average RTS Packets Transmission

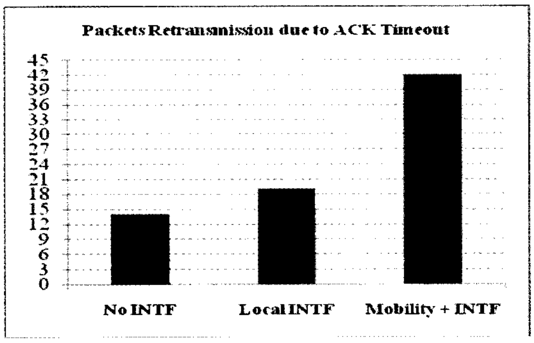

Fig. 5 (b). Average Packets Retransmission due to ACK timeout

5. Conclusion

In this paper, we propose a universal vertical cross layer framework. The key contributions are functional vertical cross layer architecture, concept and identification of awareness parameters for true performance optimization, functional architecture and behavior of Awareness Optimizer, optimization cycle for the cross layer optimization, and the concept of local and global optimization approach. Simulation scenarios are studied for parameter behavior with respect to local and global dynamics. The detailed simulation implementation and validation of the vertical cross layer framework behavior in various scenarios will be considered in the future papers.

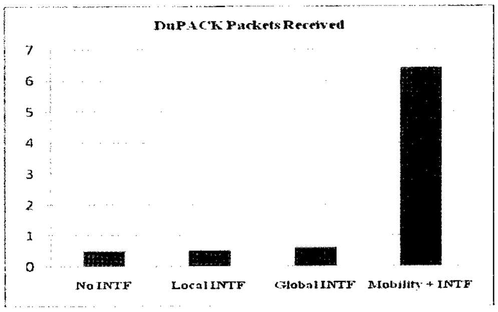

Fig. 5(c). Average DuPACK Packets Received

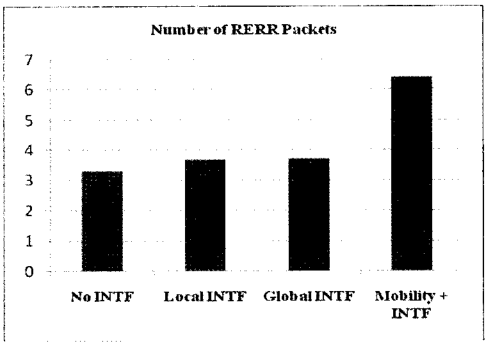

Fig. 5(d). Average Number of RERR Packets

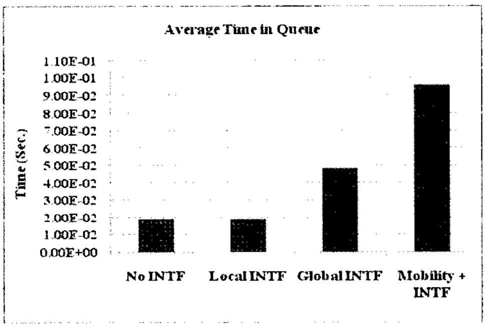

Fig. 5(e). Average Time Spent in Queue

References

[1] V. Srivastava and M. Motani, "*Cross-Layer Design: A Survey and the Road Ahead*," IEEE Communication Magazine, Dec. 2005, pp. 112-119.

[2] V. Kawadia and P.R. Kumar, "*A Cautionary Perspective On Cross-Layer Design*," IEEE Wireless Communications Magazine, February 2005, pp. 3-11.

[3] R. Jurdak, "*Wireless Ad Hoc and Sensor Networks: A Cross-layer Design Perspective*," Springer, USA 2007.

][4] F. Foukalas, V. Gazis, and N. Alonistioti, "*Cross-Layer Design Proposals for Wireless Mobile Networks: A Survey and Taxonomy*," IEEE Communication Surveys, Vol. 10, No. 1, First Quarter 2008.

[5] T. Melodia, M.C. Vuran, and D. Pompili, "*The state of the art in cross layer design for Wireless Sensor Networks," in Wireless Systems and Network Architectures in Next*

Generation Internet, Vol. 3883, Springer-Verlag, Berlin Heidelberg, 2006, pp. 78–92.

[6] G. Dimic, N. D. Sidiropoulos, and R. Zhang, "*Medium Access Control — Physical Cross-Layer Design*," IEEE Sig. Proc., Vol. 21, No. 5, Sept. 2004, pp. 40–50.

[7] J. Mirkovic, G. Orfanos, H. Reumerman, and D. Denteneer, "*A MAC Protocol for MIMO Based IEEE 802.11 Wireless Local Area Networks*," IEEE WCNC, 2007, pp. 2133-2138.

[8] T. Elbatt and T. Andersen, "*Cross-layer Interference-aware Routing for Wireless Multi-hop Networks*,“ IWCMC 2006.

[9] D. Chafekar, V. Kumar, M. Marathe, S. Parthasarathy, and A. Srinivasan, "*Cross-Layer Latency Minimization in Wireless Networks with SINR Constraints*," ACM. Mobihoc, 2007, pp. 110-119.

[10] G. Kulkarni, V. Raghunathan, and M Srivastava, "*Joint End-to-End Scheduling, Power Control and Rate Control in Multi-hop Wireless Networks*," IEEE GLOBECOM , Dec. 2004, pp. 3357-3362,.

[11] L. Chen, S. Low, M. Chiang, and J. Doyle, "*Cross-layer Congestion Control, Routing and Scheduling Design in Ad Hoc Wireless Networks*," in Proceedings of INFOCOM, 2006.

[12] T. Girici and A. Ephremides, "*Joint Routing and Scheduling Metrics in Wireless Ad Hoc Networks*," Proc. 36th Asilomar Conference on Signals Systems and Computers, Nov. 2002.

[13] R. Cruz and A. Santhanam, "*Optimal routing, link scheduling and power control in multi-hop wireless networks*," in Proceedings of INFOCOM, 2003, San Francisco, USA, Mar. 2003, pp. 702–711.

[14] K. Chen, S.H. Shah, and K. Nahrstedt, "*Cross-layer Design for Data Accessibility in Mobile Ad Hoc Networks*," Kluwer Wireless Personal Communications, 21:4976, 2002.

[15] R. Jurdak, "*Modeling and Optimization of Ad Hoc and Sensor Networks*," Bren School of Information and Computer Science, University of California Irvine, Ph.D. Dissertation, Sept. 2005.

[16] M.I. Tiado, R. Dhaou, and A.L. Beylot, "*RCL: A new Method for Cross–Layer Network Modelling and Simulation*," June 2005, ttp://whitepapers.techrepublic.com.com/abstract.aspx?docid=307826.

[17] V. Zaborovsky, A. Gorodetsky, and A. Lapin, "*Network Complexity: Cross Layer Models and Characteristics*," in Advanced International Conference on Telecommunications, May 2007.

[18] C. Barrett, M. Drozda, A. Marathe, and M.V. Marathe, "*Characterizing the Interaction Between Routing and MAC Protocols in Ad-hoc Networks*," *ACM Mobihoc*, June 2002, pp. 92-103.

[19] M. Conti, G. Maselli, G. Turi, and S. Giordano, "*Cross-Layering in Mobile Ad Hoc Network Design*," *IEEE Computer Society*, Vol. 37, Issue 2, Feb. 2004, pp. 48-51.

[20] H. Celebi and H. Arslan, "*Enabling Location and Environment Awareness in Cognitive Radios*," *Elsevier Computer Communications-Special Issue on Advanced Location-Based Services*, Vol. 31, Issue 6, Apr. 2008, pp. 1114-1125.

Murad Khalid received B.S. and M.S. degrees in Electrical Engineering from the City College of the City University of NewYork, USA, in 1993 and 1995, respectively. He is currently a research assistant working towards a Ph.D. degree in Wireless Communication and Networking area in the Department of Electrical Engineering at the University of South Florida. His research interests include mobile wireless communications and networks,, resource allocation and performance optimization of wireless Ad Hoc networks. He was with Bahria University as the Asst. Professor in Computer Engineering Department. Before that he served as senior systems engineer for Motorola, Bosch Telecom, and Ericsson Radio Systems.

Ravi Sankar received the B. E. (Honors) degree in Electronics and Communication Engineering from the University of Madras, India, in 1978, the M. Eng. degree in Electrical Engineering from Concordia University, in 1980, and the Ph.D. degree in Electrical Engineering from the Pennsylvania State University, in 1985. Since then, he has been with the Department of Electrical Engineering at the University of South Florida, Tampa. He is currently a Professor of Electrical Engineering, Supervisor of the Communications and Signal Processing graduate program track, Member of the Biomedical Engineering program and Director of the Interdisciplinary Communications, Networking and Signal Processing (*iCONS*) Research group and Interdisciplinary Center of Excellence in Telemedicine (*ICE-T*). His main research interests are in the areas of wireless communications, networking, and signal processing and its applications.

Young Hoon Joo received the B.S.,M.S., and Ph.D. degrees in Electrical Engineering from Yonsei University, Seoul, Korea, in 1982, 1984, and 1995, respectively. He worked with Samsung Electronics Company, Seoul, Korea, from 1986 to 1995, as a Project Manager. He was with the University of Houston, Houston, TX, from 1998 to 1999, as a Visiting Professor in the Department of Electrical and Computer Engineering. He is currently a Professor in the School of Electronic and Information Engineering, Kunsan National University, Korea. His major interest is mainly in the field of intelligent control, intelligent control, and nonlinear systems control. He is serving as Vice-President for the Journal of Fuzzy Logic and Intelligent Systems (KFIS) (2007-2008) and Director for the Transactions of the Korean Institute of Electrical Engineers (KIEE) (2006-2008) and Editor for International Journal of Control, Automation and Systems (IJCAS)(2008).

In-ho Ra received M.S. and Ph.D. degrees in Computer Engineering from the Chung-Ang University, South Korea, in 1991 and 1995, respectively. He is currently a Professor in the School of Electronic and Information Engineering, at the Kunsan National University. He is a visiting scholar at the University of South Florida from 2007 to 2008. His research interests include mobile wireless communication networks, sensor networks, middleware design, cross-layer design, and QoS management.

Appendix

Table 1. Awareness Parameters

Awareness Parameters	Interface	Awareness Descriptors	Summary
Energy	System and User Interface	System Awareness	Represents energy state of the node (local view)
System Interrupt	System and User Interface	System Awareness	Represents system bugs, loops, failures and interrupts (local view)
Meteorological State	Protocol Stack Interface (source: PHY Layer)	Meteorological Awareness	Represents weather state for adaptation (local view)
Topographical State	Protocol Stack Interface (source: PHY Layer)	Topographical Awareness	Represents detailed terrain, vegetation, buildings, Google map information for adaptation (local view)
Modulation and Coding	Protocol Stack Interface (source: PHY and MAC Layers)	Protocol Stack Awareness	Represents modulation and coding used for transmission (local view)
Antenna Mode	Protocol Stack Interface (source: PHY and MAC Layers)	RF Awareness	Represents multi-antenna multiplexing, diversity or beam-forming modes (local view)
Protocol Suite	Protocol Stack Interface (source: All Layers)	Protocol Stack Awareness	Represents protocol combinations used through all the layers (local view)
RF State	Protocol Stack Interface (source: PHY Layer)	RF Awareness	Represents RF parameters: Angular Spread, Delay Spread, Doppler Spread, Signal-to-Noise Plus Interference, BER, Signal Strength, Power, Frequency Band (local view)
MAC State	Protocol Stack Interface (source: MAC Layer)	Protocol Stack Awareness, Network Awareness	Represents MAC layer parameters: No. of stations, No. of retransmissions, Messages to/from PHY layer, No. of ACKs, ARQ techniques, Back-off techniques, Average Back-off window size (local view)
NET Sate	Protocol Stack Interface (source: NET Layer)	Protocol Stack Awareness, Network Awareness	Represents Network layer parameters: ICMP control parameters (if enabled), No. of Route Requests, No. of Route Updates, No. of Route Errors, Average data and control queue length, Average data and control queue delay, No. of fragments to MAC layer, No. of fragment retransmissions, No. of hops, TTL, Queue policy, Scheduling policy
TRAN State	Protocol Stack Interface (source: TRAN Layer)	Protocol Stack Awareness, Network Awareness	Represents Transport layer parameters: Control messages, RTT, No. of DuPACK, No. of ACKs, Window Size, No. of packets exchanged with adjacent layers, No. of packets retransmitted, MTU size, Control messages
User Time and Location	System and User Interface	User Awareness	Represents user behavior in time and space using GPS based Location Awareness (local view)
User Interrupt	System and User Interface	User Awareness	Represents user command (local view)
Network Type	Protocol Stack Interface (source: All Layers)	Network Awareness	Represents network service type, security policy, network mode, (global view)
Application Type	Protocol Layer Interface (source: App Layer)	Application Awareness	Represents real time, non real time performance constraints, security and encryption constraints