

# An Improved Autonomous Cross-Layer Optimization Framework for Wireless Multimedia Communication in Heterogeneous Networks

**Yuping Dong**

*Department of Electrical and Computer Engineering  
Tufts University, Medford, MA, USA  
E-mail: yuping.dong@tufts.edu*

**C. Hwa Chang**

*Department of Electrical and Computer Engineering  
Tufts University, Medford, MA, USA  
E-mail: hchang@ece.tufts.edu*

**Zengxu Yang**

*Department of Electrical and Computer Engineering  
Tufts University, Medford, MA, USA  
E-mail: zengxu.yang@tufts.edu*

## Abstract

Various cross-layer optimization schemes have been proposed for the purpose of enhancing wireless network performances. In this paper, we present an improved autonomous cross-layer optimization to fit modern heterogeneous communication networks. In our design, different Quality of Service (QoS) requirements for all sorts of information with various communication protocols are considered when making optimal transmission decisions. The modified framework achieves better system computation efficiency and adapts to various sources of different communication networks quickly while preserves the overall system performance.

*Keywords:* Autonomous cross-layer optimization; Multimedia Communication; Quality of service (QoS); Markov Decision Process (MDP); Dynamic Programming (DP); Heterogeneous Networks.

## 1. Introduction

With latest technology, the wireless electronic devices are capable of handling a variety of multimedia applications, from texting to high definition voice/video streaming, to online gaming, all of which pose very high requirements on wireless networks. The traditional layered abstraction, the Open Systems Interconnection (OSI) model partitions a communication system into seven layers, with each lower layer provides services to the layer above it. Each abstraction layer runs an individual protocol and makes independent decisions on selecting protocol parameters. When designing lower layer protocols, the designers do not consider the characteristics of specific multimedia applications, whereas the designers of higher layers do not take into account of the status of lower layers. This design manner prevents the protocol under optimization from maximizing the system performance in real time by not adapting to the parameters of the multimedia applications and the characteristics of the time-varying wireless channels.

Multiple users share the same wireless communication medium. These users interfere with each other when transmitting at the same time, which makes transmission

medium unstable. Cross-layer optimization jointly analyzes, selects and adapts different strategies available at various OSI layers in order to deliver a required data quality at the receiver by expediting the wireless network performance by lowering transmission latency, reducing device power consumption, and increasing network throughput.<sup>1</sup>

In recent years, numerous cross-layer solutions emerge one after another for the resource restricted, error-prone wireless networks. Most intuitive solutions adapt layer parameters depending upon the internal information obtained from other layers, which invalidate the layered architecture of the OSI model. By permitting interaction between layers, the freedom of designing layer protocols independently is deprived from the designers. The autonomous cross-layer optimization proposed in Ref. 2 is a layered solution in which each layer observes its own real time environment and selects its protocol parameters accordingly by allowing only minimum information exchange between layers. This method preserves the OSI layered architecture, leaves protocol designers flexibility on designing optimal protocols for each layer.

Today's advanced technology presents us a new world with high technology electronic devices capable of wireless

multimedia communications. With multiple applications installed on one device, information with various constraints is exchanged between the user and the base station. Different applications switch from one another frequently as multitasking becomes available with these devices for increased efficiency. Since the hardware resource of the wireless device is limited, it is very important to adapt the transmission protocols at each layer to the transmission constraints of different applications. Most cross-layer optimization approaches deal with delay constraint communications. Others deal with applications that require higher delivery rate or lower transmission cost. In this paper, we present an improved autonomous cross-layer optimization that takes into consideration of real time switching among multiple applications with different QoS requirements when selecting optimal decision variables (i.e., power allocation, modulation scheme, retransmission limit, and more.) for each layer. Simulation results show that our improved design greatly reduces the processing time of the cross-layer optimization over the original design in Ref. 2, while achieving a similar system performance.

Nowadays, wireless is everywhere, in every realm of human's daily activity. A series of popular low cost, low power, short ranged wireless communication protocols working on the Industrial, Scientific, and Medical (ISM) radio bands, such as ZigBee,<sup>3</sup> Bluetooth,<sup>4</sup> and WiFi<sup>5</sup> are being deployed for wireless Body Area Networks (BAN), Personal Area Networks (PAN), and Local Area Networks (LAN). The emerging new concepts such as Ubiquitous Computing, Pervasive Computing, and Internet of Things are used to describe a world that such wireless networks are used for much wider applications than the cellular networks and the Internet today are capable of. An innovative unified platform for heterogeneous network communications is currently being developed in the Tufts Wireless Laboratory (TWL). Our cross-layer design aiming at integrating into the unified platform is conveniently designed to flexibly switch among different wireless protocols. Implemented with cross-layer optimization, the unified wireless mesh networks using a variety of low power, inexpensive wireless sensors will achieve better communication, and longer lifetime that conventional wireless sensor networks<sup>6</sup> are not comparable to.

In this paper, the original autonomous cross-layer optimization framework proposed in Ref. 2 is adopted and improved. The new framework is then applied to the unified network platform of heterogeneous network communications.

The rest of this paper is organized as follows. In section II, we briefly review previous cross-layer optimization designs.

We describe the autonomous cross-layer optimization framework and thoroughly explain our improvement in section III. In section IV, the unified platform of wireless heterogeneous networks that is under development, and the integration with our cross-layer design are explained in detail. Explicit examples and simulations with both our proposed design and the original design in Ref. 2 are presented in section V. Finally we conclude this paper with a discussion of future research on this subject.

## 2. Related Work

Cross-layer optimization is widely researched in recent years to maximize the usage and optimize the performance of resource limited wireless devices.<sup>7</sup>

The most intuitive methods select the optimal protocol parameters depending on the information directly retrieved from other layers. In this type of solutions, one specific layer is permitted to directly access related layers (usually not adjacent layers) to obtain their internal information in order to enhance its performance. Examples of this type are presented in Refs. 1 and 8-12. These solutions jointly adapt multiple layers, which in turn greatly improves the overall network performance and reliability of the wireless users. However, the inter-layer communication between non-adjacent layers violates the layered architecture of the OSI model, since direct access to internal variables of other layers is essential. The OSI layered architecture enforces strict boundaries between layers. The violation to this architecture blurring the boundaries between layers makes protocol design of certain layers dependent to other layers.

A centralized optimization solution uses a centralized optimizer (middleware or system controller) to dynamically allocate resources across network based on the observed environmental dynamics, resources and delay constraints of each node in the network. T. Holliday et al. proposed a centralized optimization system in Ref. 13. In setup phase, the centralized optimizer gathers information, such as channel status, power constraints from the network and users, and provides an optimal control policy for the users. With this design, the centralized optimizer also violates the OSI layered architecture by obtaining the complete information from all layers before making the decision on resource allocation for each layer. Modifying the protocol or algorithm of one layer without considering other layers' information gathered by the centralized optimizer is not accomplishable.

Fig. 1 presents a number of cross-layer designs built over the seven OSI layers.

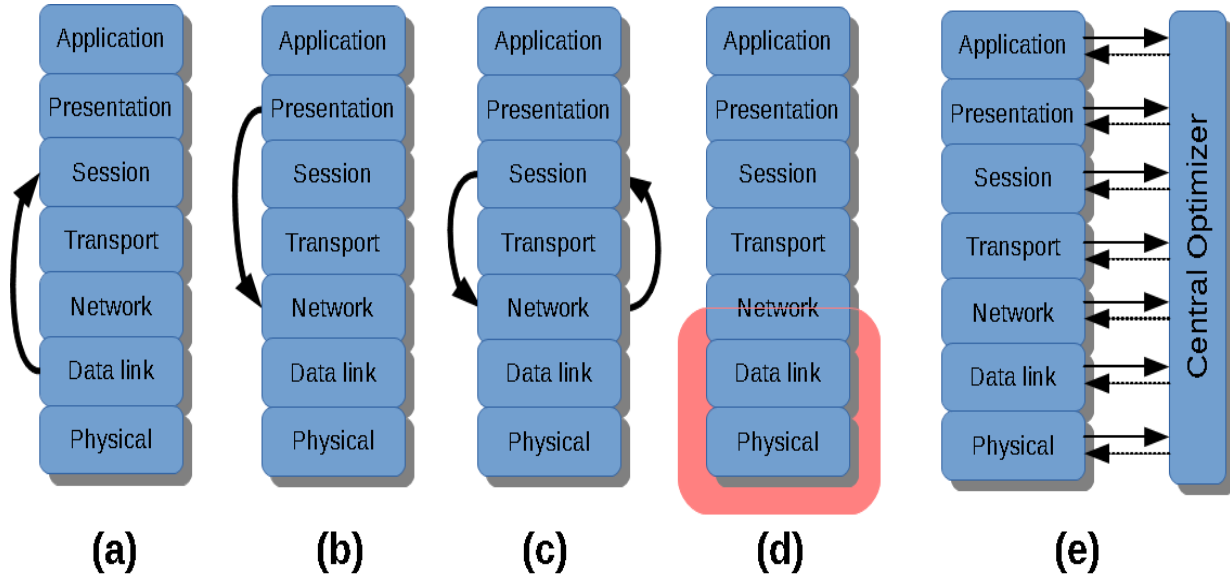


Figure 1. Examples of common cross-layer design protocols .

Fig. 1a-c create new interfaces between non-adjacent layers to allow information exchange. Certain layer can directly provide internal protocol parameters to other layers. For example, Data Link layer can provide Application (APP) layer with channel availability, while Physical (PHY) layer can provide energy level. APP layer then adjusts its source coding algorithm accordingly based on the information it has retrieved from lower layers. Examples of these cross-layer designs are studied in Refs. 8, 9 and 11. The protocol optimization of one layer affects the design of related layers. In Fig. 1d, multiple lower layers are grouped into a “black box” and provide necessary feedback information to the upper layers in order for them to correspondingly adapt the transmission strategies. This design ignores the adaptability between the grouped lower layers. The centralized cross-layer design shown in Fig. 1e requires each layer to provide the central optimizer with internal parameters. The central optimizer then determines on the action variables for each layer. F. Foukalas et al. listed a variety of existing cross-layer design methods, and discussed the challenges and problems of each design in detail.<sup>14</sup>

The autonomous cross-layer optimization framework proposed in Ref. 2 preserves the layered architecture of the protocol stack by adapting the transmission strategies autonomously in each layer according to the local environment dynamics and the messages exchanged between adjacent layers. The extra information exchanged between layers is minimized. The design is free of non-adjacent layer communications. A Markov Decision Process (MDP)<sup>15</sup> is used on selecting the optimal protocol variables for each layer and a foresighted layer adaptation, which considers future utility

rewards in addition to the current reward, was adopted to ensure more accurate optimization.

### 3. Autonomous Cross-layer Design Architecture

The autonomous cross-layer design modeled as a MDP makes foresighted strategy adaptation based on the discounted cumulative network utility as in Refs. 13, 16 and 17.

A layered MDP design allows each layer to make its own transmission decision autonomously with the environmental dynamics it experienced and the information exchanged between adjacent layers. For example, APP layer determines the optimal source scheduling, PHY layer decides on the effective power allocation independently from any other layer. This solution is fully compatible with the layered architecture in wireless network protocol implementation.

#### 3.1. Basic Concepts

##### 3.1.1. States

Each layer is indexed by  $l$  with  $l \in \{1, \dots, L\}$ . Layer 1 represents the lowest layer in the stack, the PHY layer, while layer  $L$  corresponds to the highest layer, the APP layer. The current state and the next state of each layer are denoted by  $s_l$  and  $s'_l$  separately. The state of the wireless user is then displayed as  $s = [s_1, \dots, s_L]$ .

The three participating layers are modeled as Finite State Markov Chains (FSMC) as illustrated in Refs. 18-19. The state  $s$  is defined such that the transition from current state to

next state does not depend on the history of states precede the current state as stated in the definition of FSMC.

PHY, MAC and APP layers are used to show the optimization process. The optimization of other layers can be established for various wireless networks analogously. Fig. 2 shows the

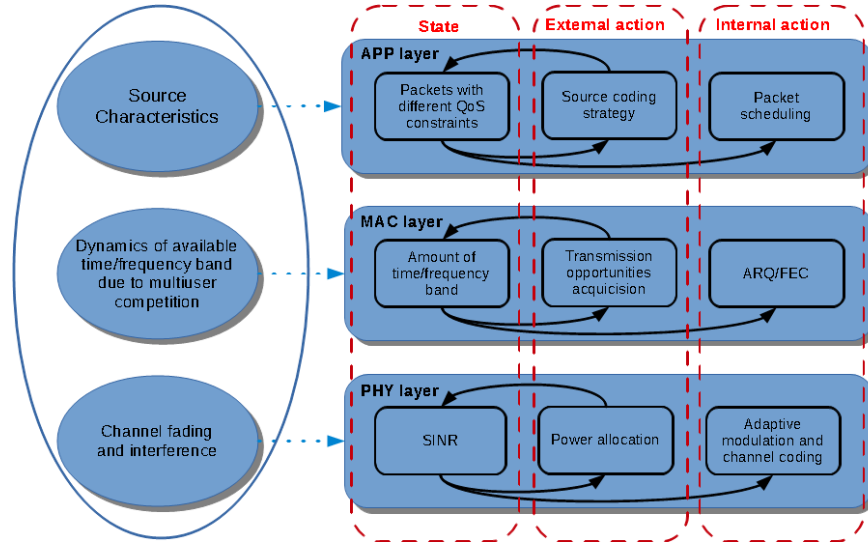


Figure 2. Layered cross-layer optimization model.

layered cross-layer design model.

### 3.1.2. Actions

Two types of transmission action, the external action  $a$ , and the internal action  $b$ , are defined to represent the transmission strategies at each layer. Depending on the experienced dynamics, and the foresighted cumulative rewards, wireless users take different transmission actions at each layer to maximize the system quality. A set of possible external actions  $\mathbf{A}$  and a set of possible internal actions  $\mathbf{B}$  for different technologies are given at each layer before the optimal decisions are made by the layered optimizer.

$\xi_l = [a_l, b_l]$  is used to represent the mixed action at layer  $l$ .

### 3.1.3. Transition Probability

Each layer is modeled as a FSMC, which is a random process whose next state depends only on the current state and not on the sequence of states that preceded it. The next state of each layer is determined by the external action performed at current state. The service provided to the upper layer is determined by the internal action taken at current state. Therefore, the transition probability from current state  $s$  to a possible next state  $s'$  of a FSMC  $p(s' | s, \xi)$  is only determined by the current state, the external action taken, and the environmental dynamics.

## 3.2. System Model

A one-hop wireless communication model between an end user and the base-layer station is adopted to illustrate the autonomous cross-layer design as in Ref .2. Three layers,

### 3.2.1. Physical layer model

At PHY layer, the Signal to Interference and Noise Ratio (SINR) is derived by comparing the received signal strength to the interference level from other users and the channel noise. We use SINR as the PHY layer state  $s_1$ , and model the time-varying Rayleigh fading channel as a FSMC as described in Ref .20. State transition is driven by the allocated transmission power  $a_1$  which is considered the external action at PHY layer. One-step transition for the PHY layer states is used in the model. Thus only the transitions between adjacent states are allowed. The possible received SINR at the receiver can be divided into  $k+1$  regions with thresholds of  $\Gamma_0, \dots, \Gamma_{k+1}$ , and  $\Gamma_0 < \Gamma_1 < \dots < \Gamma_{k+1}$ . The PHY layer is said to be in state  $s_r$  if the SINR of the received packet lies in the range of  $[\Gamma_r, \Gamma_{r+1})$ .

The channel gain of the Rayleigh fading channel has an exponential distribution with a probability density function of

$$p(\gamma) = \frac{1}{\gamma_0} \exp\left(-\frac{\gamma}{\gamma_0}\right), \quad \gamma \geq 0. \quad (1)$$

where  $\gamma_0$  is the average received SINR, which is determined by  $a_1$ . The state transition probability is then calculated as in Ref .20.

$$p(s'_1|s_1, a_1) = \begin{cases} \mathcal{N}(\tilde{\Gamma}_{i+1}) \frac{T_p}{\omega_i}, & s_1 = \tilde{\Gamma}_i, s'_1 = \tilde{\Gamma}_{i+1} \\ \mathcal{N}(\tilde{\Gamma}_i) \frac{T_p}{\omega_i}, & s_1 = \tilde{\Gamma}_i, s'_1 = \tilde{\Gamma}_{i-1} \\ 1 - \mathcal{N}(\tilde{\Gamma}_{i+1}) \frac{T_p}{\omega_i} - \mathcal{N}(\tilde{\Gamma}_i) \frac{T_p}{\omega_i}, & s_1 = \tilde{\Gamma}_i, s'_1 = \tilde{\Gamma}_i \\ 0, & o.w. \end{cases} \quad (2)$$

where  $\mathcal{N}(\gamma) = (2\pi\gamma/\gamma_0)^{1/2} \exp(-\gamma/\gamma_0)$ ,

$\omega_i = \exp(-\Gamma_i/\gamma_0) - \exp(-\Gamma_{i+1}/\gamma_0)$ ,

$f_d$  is the maximum Doppler frequency caused by the wireless user's motion,

and  $T_p$  is the transmission time for one packet.

### 3.2.2. MAC layer model

MAC layer is also modeled as a FSMC with the state  $s_2$  representing the amount of time that the wireless channel is allocated to the user.

MAC layer external action  $a_2$  is defined as user's competition bid for the acquisition of spectrum usage for TDMA.

The control mechanism, Automatic Repeat reQuest (ARQ), is used at MAC layer to enhance the QoS provided to upper layer at the receiver by requesting the transmitter to retransmit a block of data when errors are detected. The number of retransmission limit is considered as MAC layer internal action  $b_2$ .

If channel access is based on Time Division Multiple Access (TDMA), state transition is determined by the competing strategies of the user. When Code Division Multiple Access (CDMA) or Frequency Division Multiple Access (FDMA) is used, the user always has the access to the channel. In this case, the MAC layer state stays constant all the time,<sup>21</sup> which means the state transition probability

$$p(s'_2 = 1 | s_2 = 1, a_2) = 1$$

where  $a_2 = \phi$ .

### 3.2.3. APP layer model

At APP layer, different data are generated from time to time for various applications. We divide the incoming data into three categories, delay-constrained data, throughput-sensitive data, and cost-restricted data, based on QoS requirements of each application. When delay-sensitive data are generated, we assume each packet has a lifetime of  $J$  stages. After  $J$  stages, the packet will be expired. If it is not successfully transmitted, it will be dropped.  $s_3^k = [s_{3,1}^k, \dots, s_{3,J}^k]$  represents the APP layer state at stage  $k$ . The external action  $a_3^k$  determines the number of packets arriving at APP layer in the beginning of

stage  $k$ . We assume it as the average number of arriving packets, and the actual number of newly generated packets at stage  $k$  is a random variable  $Y_3^k(a_3^k)$ . Therefore, the next state of APP layer at stage  $k+1$  is computed as

$$\begin{bmatrix} s_{3,1}^{k+1} \\ \vdots \\ s_{3,j}^{k+1} \\ \vdots \\ s_{3,J}^{k+1} \end{bmatrix} = \begin{bmatrix} s_{3,2}^k - \max(n_3^k - s_{3,1}^k, 0) \\ \vdots \\ s_{3,j+1}^k - \max(n_3^k - \sum_{m=1}^j s_{3,m}^k, 0) \\ \vdots \\ Y_3^k(a_3^k) \end{bmatrix}. \quad (3)$$

where  $n_3^k$  is the number of packets transmitted in stage  $k$ . The state transition probability is then calculated as follows.

$$p(s_3^{k+1}|s_3^k, a_3^k, Z_L^k) = \begin{cases} P(Y_3^k(a_3^k) = y), & \text{if (3) holds} \\ 0, & o.w. \end{cases} \quad (4)$$

where  $Z_L^k = (\epsilon_L^k, \tau_L^k, v_L^k)$  is the optimized QoS at APP layer with a packet error rate of  $\epsilon_L^k$ , a packet transmission time of  $\tau_L^k$ , and a packet transmission cost of  $v_L^k$ .

For other applications that have transmission accuracy and transmission power as constraints, the information waiting to be transmitted may not have a strict deadline. In this case, the APP layer is modeled differently. An incoming data buffer with a capacity of *buffersize* is maintained at the wireless transmitting device. Excessive packets are dropped if the buffer is full. In this case, the next state of the APP layer becomes

$$s'_3 = \min(s_3 - n_3 + Y_3, \text{buffersize}). \quad (5)$$

## 3.3. The Improved Layered Cross-layer Optimization

### 3.3.1. Markov decision process (MDP)

Markov Decision Process is used to solve the layered cross-layer optimization. MDP is a discrete time stochastic control process. The definition is given below.

**Definition 1.** MDP is a 4-tuple (S, A, P, R) where

- S is a set of finite number of states,
- A is a set of finite number of actions available at current state,
- $P(s_{t+1}=s' | s_t=s, a)$  is the probability of the process moving to state  $s'$  when action  $a$  is taken at state  $s$ ,
- R is the immediate reward given to the decision maker after the process goes to state  $s'$ .

The state transition of MDP only depends on the current state  $s$  and the action  $a$  taken at current state. Any previous states or actions have no effect on the transition to the next state. The process gives a reward to the decision maker for

the action it selected after it moves to the next state. A discounted cumulative reward, Eq. (6) takes future rewards into consideration in addition to the immediate reward given at state transition. The immediate reward gets more credit

$$\sum_{k=0}^{\infty} \gamma^k R(s^k, \xi^k | s^0) \quad (6)$$

than the future rewards, because unpredictable situation in the future is likely to happen to wireless communications. An efficient cross-layer design implemented with MDP selects the optimal external and internal actions that maximize the discounted cumulative rewards. Dynamic programming (DP)<sup>22</sup> is used to effectively solve the problem.

### 3.3.2. Improvements for multimedia communications

Modern multimedia communications applied to vast application areas often have different QoS requirements. To ensure that the cross-layer optimization satisfies these QoS requirements and makes quick response to wireless user's application switch, we improve the layered cross-layer optimization proposed in Ref .2 with following modifications:

- In the original design, when calculating the system reward, only network throughput and packet transmission cost were considered in the application utility. Whereas, in our approach, we included the value generated by packet transmission latency to complete the system reward.
- When forming the QoS frontier set to the upper layer, the original design promotes whatever QoS that has at least one portion better than the previously selected ones regardless of the application requirements. This method inevitably selects redundant QoS'es to be included in the frontier set, which in turn increases consumed computation source of the resource limited wireless devices. To fix this problem, we modified this part such that when selecting QoS frontiers to the upper layer, we consider all three portions by giving them different weights according to different applications. The QoS with better overall credit gets promoted to the upper layer.

Each layer's DP operator is designed as in Ref .2. In Table 1 we simplify them to the one-hop wireless network, with which only three layers are considered for optimization.

The function  $V()$  in Table 1 is the state value function with the states as parameters.  $R_{in}$  is the internal reward which is computed by subtracting internal cost incurred by taking internal actions from the application quality achieved with the APP layer QoS  $Z_3$  of the wireless user.  $c_1$ ,  $c_2$  and  $c_3$  are the external costs generated by taking corresponding external actions at the three layers.  $\lambda_1^a$ ,  $\lambda_2^a$ ,  $\lambda_3^a$  are three positive parameters that trade off between the application quality and the external costs.

In Ref .2,  $R_{in}$  only takes system throughput into account. We added one portion of the value that benefits from the short transmission latency of the network. The new  $R_{in}$  including both throughput and latency, along with the internal and external costs considered in the value calculation, the whole system is more complete.

Table 1. DP operator of each layer

Layer	DP Operator
APP	$V_2(s'_1, s'_2) = \max_{a_3, Z_3} [R_{in}(s_3, Z_3) - \lambda_3^a c_3(s_3, a_3) + \rho \sum_{s'_3 \in \delta_3} p(s'_3   s_3, Z_3, a_3) V(s'_1, s'_2, s'_3)]$
MAC	$V_1(s'_1) = \max_{a_2} [-\lambda_2^a c_2(s_2, a_2) + \sum_{s'_2 \in \delta_2} p(s'_2   s_2, a_2) V_2(s'_1, s'_2)]$
PHY	$V(s_1, s_2, s_3) = \max_{a_1} [-\lambda_1^a c_1(s_1, a_1) + \sum_{s'_1 \in \delta_1} p(s'_1   s_1, a_1) V_1(s'_1)]$

### 3.4. Quality of Service Computation

As shown in Table 1, the determination of optimal APP layer QoS is essential in maximizing both the internal reward and the future reward. Each layer's QoS consists of three portions in this paper: the packet loss rate  $\varepsilon_l$ , the transmission time of one packet  $\tau_l$ , and the packet transmission cost  $v_l$ . These values are calculated based on current state, internal action taken, and the QoS provided by the lower layer.

To simplify the calculation and save computation energy, the set of possible QoS values provided from the lower layer are filtered before being used to calculate the upper layer's QoS set. Thus only the frontier of the QoS set at the lower layer is passed to the upper layer.

In Ref .2, if any one of the three portions of a QoS is smaller than all the previously selected QoS'es, this QoS is included in the QoS frontier, regardless of the application requirements.

To improve this QoS frontier computation process, we consider all three parts of the QoS, and give them different weights for different applications. Below we give the criteria of whether a QoS belongs to the QoS frontier set.

**Criteria 1.** Assume  $QoS_1$  is already included in the QoS frontier set. To determine if a new QoS  $QoS_2$  should be included in the frontier set, we calculate the overall credit of  $QoS_2$  by comparing it with  $QoS_1$ . If the overall weighted value of  $QoS_2$  is larger than  $QoS_1$ ,  $QoS_2$  will not be included in the frontier set. Otherwise, we include  $QoS_2$  in the QoS frontier set that is then provided to the upper layer. We calculate the overall credit (OC) of a QoS under consideration as in Eq. (7).

$$OC = \frac{QoS_2 - QoS_1}{QoS_1} \quad (7)$$

$$= \frac{QoS_2 \cdot \varepsilon_2 - QoS_1 \cdot \varepsilon_1}{QoS_1 \cdot \varepsilon_1} * w_e + \frac{QoS_2 \cdot \tau_2 - QoS_1 \cdot \tau_1}{QoS_1 \cdot \tau_1} * w_r + \frac{QoS_2 \cdot v_2 - QoS_1 \cdot v_1}{QoS_1 \cdot \varepsilon_1} * w_v$$

#### 4. Integration of Cross-Layer Design with the Unified Platform of Heterogeneous Network

##### 4.1. Overview of the Unified Platform

Today in the world of wireless communications, a large amount of networks using different protocols exist for various purposes. They are often used together in one system to serve different parts separately. A unified data exchange platform is required to ensure different networks run as expected for the whole system. The unified platform of heterogeneous network Communications<sup>23</sup> is a distributed and delay tolerant communication platform that enables different wireless networks to communicate with each other and serves as a control center to remotely collect information, send commands or configuration parameters to those networks. Each network is connected to the platform via a gateway that understands the wireless protocol that the network uses.

Fig. 3 shows a block diagram of the proposed platform. As illustrated in the figure, the platform serves as a control and communication center. A ZigBee transceiver module is attached to the USB port serving as a ZigBee gateway for communications with a ZigBee vehicular network that was also developed at Tufts Wireless Laboratory by T. Zeybek and Prof. H. Chang.<sup>24</sup> Each vehicular unit is able to interact with the platform via the ZigBee gateway, and further communicate with other devices of a different network through data exchange between the two gateways.

The detailed structure of the platform is shown in Fig. 4. A database is used to store and query the network information. PostgreSQL is used in the prototype for its full support of SQL standards and open source license. Tomcat is used to serve as a Java web engine to support Java web programming. Java servlets can communicate with PostgreSQL database and gateway ports. The web user interface and database modules are described in Ref. 25. OpenDDS middleware is adopted as a publish/subscribe communication mechanism to enable distributed, real-time, high performance data exchange in the platform.<sup>26</sup>

##### 4.2. Integration of Cross-Layer Design with the Platform

The unified platform enables interactions between different wireless networks, makes it possible to form a complete system consisting of multiple wireless networks using different protocols. For example, to manage chronic diseases

or monitor patients' post-hospitalization, a number of wireless devices, including glucose monitor, ECG monitor, etc. are used to collect the patient's health status data. Some of the devices may use different communication protocols than others. With the use of the unified platform, all of the wireless healthcare devices can work together and give a better overview of the patient's status. Since these wireless devices

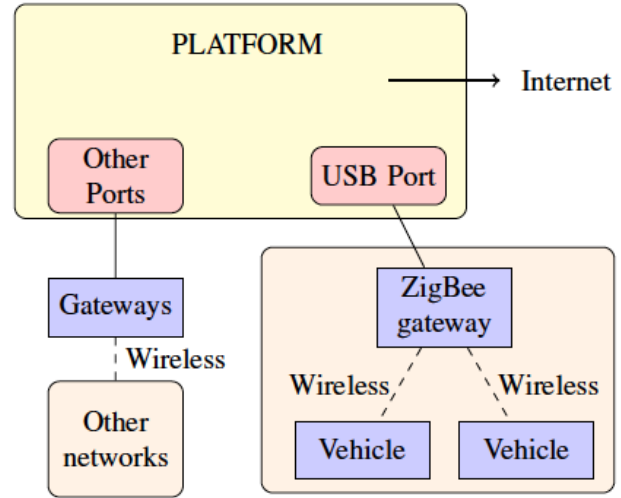


Figure 3. Block diagram of unified communication platform.

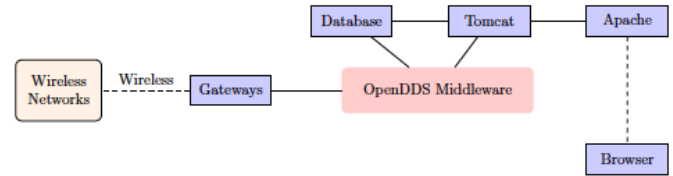


Figure 4. Detailed communication platform.

are battery powered, it is very important to keep the power consumption of transmission as low as possible. For some high-risk patients, the data collected by the monitors have to be transmitted in a timely manner. Applying cross-layer optimization to the transmission protocols of these wireless devices will solve the aforementioned problems and make the complete system more reliable and power efficient.

To apply cross-layer optimization to a device, we need to patch the wireless network card driver with our cross-layer design. We develop layered cross-layer designs for different wireless protocols and store them in the database separately at the unified platform. The platform then notifies all the gateways the availability of cross-layer optimization for the corresponding protocols. When a device tries to communicate with its gateway, the gateway transmits appropriate commands from the platform to modify the device's wireless network card driver. After this is done, the wireless device transmits

efficiently which will greatly improve the overall performance of the system.

## 5. Simulation and Results

We have developed the improved autonomous cross-layer optimization and applied it to a Cellular network<sup>27</sup> as well as a Bluetooth network.

In this section, we simulate both networks with multimedia communication in which one wireless user with autonomous cross-layer optimization transmits multimedia information to another user over a one-hop network.

### 5.1. Simulation with Cellular Network

To compare our improved design with the original design in Ref .2, for Cellular network, we used the same settings for simulation parameters provided in Table IV in Ref .2, except that the Bit Error Rate (BER) was computed as in Eq. (7) as in Ref .28 instead of using the one provided in Ref .2 which did not seem to give reasonable values.

$$BER = \frac{1}{b} \operatorname{erfc}(\sqrt{s_1} \sin \frac{\pi}{2m}) \quad (7)$$

Here  $b$  is the number of bits per symbol for specific modulation scheme, and  $m = 2^b$ .

We assume at APP layer, the packet arriving pattern follows the Poisson Distribution with a PDF of

$$\Pr(x = k) = \frac{\lambda^k e^{-\lambda}}{k!} \quad (8)$$

where  $\lambda$  is the average number of packets arriving at the application. A buffer size of 8 packets is used.

The weights we assign to each QoS portion are 0.2, 0.2, and 0.6 depending on the application.

Table 2 presents the calculation time of each MDP iteration for different applications with both designs. The improved design spends only a half of the optimization time of the original one. This in turn saves significant amount of computation energy for the wireless user.

In Fig. 5, we show the simulated state values of both the original autonomous cross-layer design and our improved design. The results of three applications follow the same pattern as SINR varies. Some representative plots with different SINR values are displayed. Fig. 5a shows the state values of a cost efficient application when SINR is in small to medium range. Fig. 5b shows the state values of a delay sensitive application when SINR is in medium to large range. We can see from the plots that when the SINR is not big, the state values of the improved design are almost the same as the original design. When SINR is high, our design achieves smaller state values than the original design. This is because at those PHY states, the possible QoS provided to upper layer can have very low packet error rates, which were included in

QoS frontier based on the overall credit. But those QoS'es usually do not have the lowest transmission delay, which in turn yields smaller state values.

Table 2. Average calculation time for Cellular network.

Application	Original Design	Improved Design
Delay Sensitive	1.516ms	0.781ms
Throughput Sensitive	0.739ms	0.357ms
Cost Constrained	0.781ms	0.374ms

### 5.2. Simulation with Bluetooth Network

At PHY layer, the same Rayleigh fading channel is used. The SINR is among the range of 0 to 18dB, available modulation schemes are Gaussian Frequency-Shift Keying (GFSK),  $\pi/4$ -Differential Quaternary Phase-Shift Keying ( $\pi/4$ -DQPSK), and 8 Differential Phase-Shift Keying(8DPSK). The transmitted power of class 1 Bluetooth device has a maximum value of 100mW and a minimum amount of 1mW. Bit Error Rate and Packet Loss Rate are calculated as in Refs. 29-30.

Table 3. Average calculation time for Bluetooth network.

Application	Original Design	Improved Design
Delay Sensitive	1.815ms	1.48ms
Throughput Sensitive	1.074ms	0.78ms
Cost Constrained	1.062ms	0.8ms

At MAC layer, ARQ scheme is used. Available channel access methods are TDMA and FDMA. During the connection, a Bluetooth device can be in one of the following four modes: Active mode, Sniff mode, Hold mode, and Park mode.<sup>31</sup> The external action is defined as to determine the mode that the Bluetooth device is in. We use the same APP layer parameters as for the Cellular network.

The calculation times of each MDP iteration for both the improved design and the original design are shown in Table 3. The improved design applied to Bluetooth device also outperforms the original design with spending less computation time.

Fig. 6 shows the plots of state values of simulated Bluetooth transmission with both the original cross-layer design and our improved design. In Fig. 6a, plots of state values of a throughput efficient application are given with



different PHY layer state values. Fig. 6b shows the plots of a delay sensitive application. Observed from the plots, our improved cross-layer design achieved very close performance to the original design.

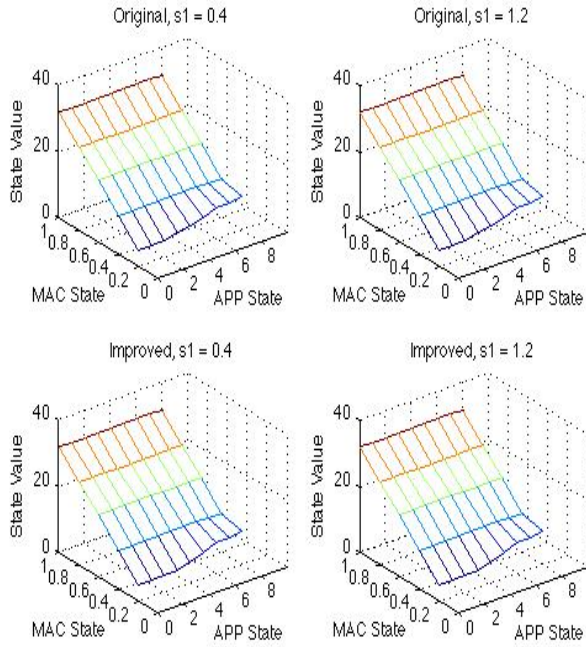


Figure 5a. Simulated state value of cost efficient application for cellular network.

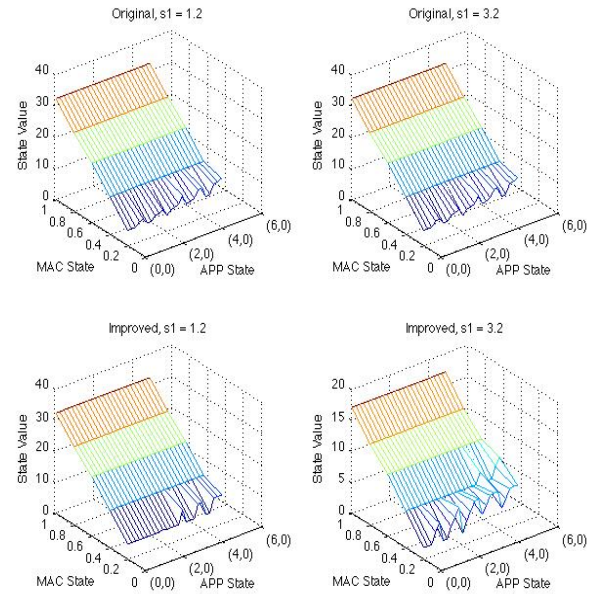


Figure 5b. Simulated state value of delay sensitive application for cellular network.

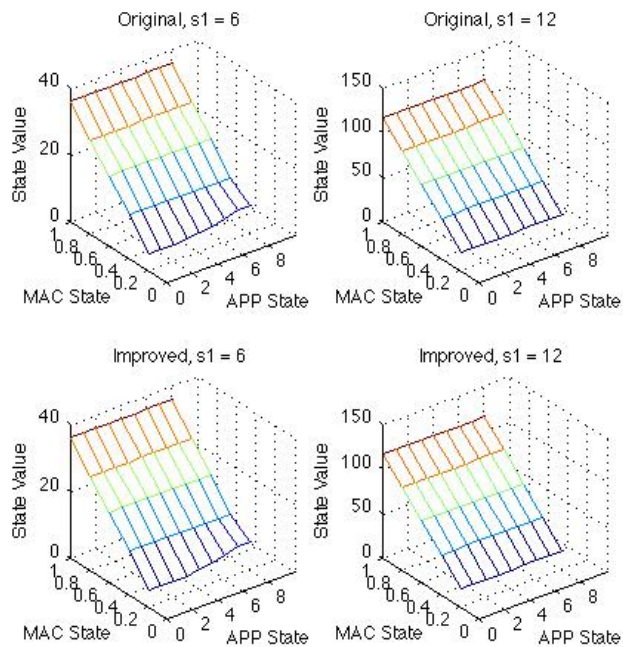


Figure 6a. Simulated state value of throughput efficient application for Bluetooth network.

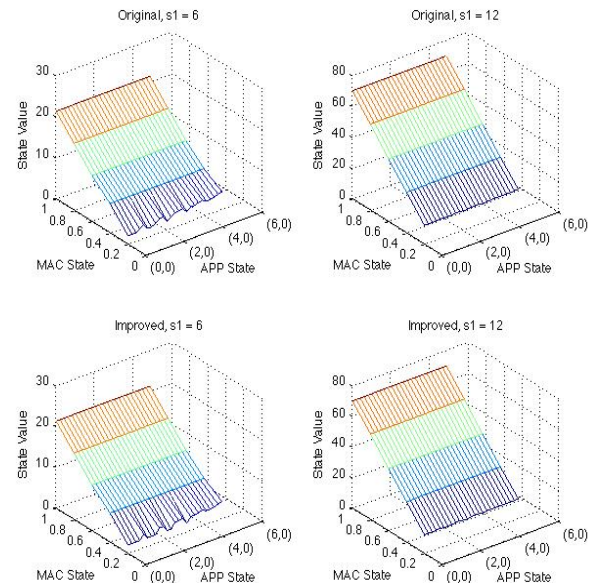


Figure 6b. Simulated state value of delay sensitive application for Bluetooth network.

## 6. Conclusion

With advanced technology, wireless users can install various applications on their mobile devices. These resource-limited devices may be transmitting information of different QoS requirements from time to time. It is critical for the device to be capable of promptly optimizing the transmission all the time, in order not to consume too much energy. This paper presented an improved autonomous cross-layer optimization which solved the aforementioned problem. Simulation results showed the proposed design saved 50% computation power of the autonomous cross-layer optimization proposed in Ref. 2, while achieved close performance for a cellular network. The integration of the proposed cross-layer design into the unified heterogeneous network platform was also proposed. Simulation of the improved cross-layer design with a Bluetooth network was done with a promising result which proves that our design is flexible and the integration with the unified platform is possible to accomplish. With the rapid development of mobile networks, 5G cellular networks are already being researched to provide even higher data rate and wider coverage than 4G LTE does<sup>32</sup>. It is possible that our work could serve to deliver data more quickly and reliably for the 5G networks. Our future research will be to implement this design on real hardware as in Ref. 33, and to complete the integration with the unified platform of heterogeneous networks in reality.

## References

1. M. van der Schaar and S. Shankar, "Cross-layer wireless multimedia transmission: Challenges, principles, and new paradigms," *IEEE Wireless Commun.*, vol. 12, no. 4, pp.50–58, August 2005.
2. F. Fu, M. van der Schaar, "A new systematic framework for autonomous cross-layer optimization," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 1887–1903, May 2009.
3. Z. Alliance, *Zigbee Specification* (ZigBee standards organization, 2008).
4. B. SIG, *Specification of the Bluetooth System* (2001).
5. M. Gast, *802.11 Wireless Networks: The Definitive Guide* (O'Reilly Media, 2002).
6. A. Davis and H. Chang, A Survey of Wireless Sensor Network Architectures, *Int. J. Comput. Sci. Eng. Surv.*, vol. 3, no. 6 (2012).
7. B. Fu, Y. Xiao, H. Deng, and H. Zeng, A Survey of Cross-Layer Designs in Wireless Networks, *IEEE Commun. Surv. and Tuto.* (2014).
8. X. Wang, Q. Liu, and G. B. Giannakis, "Analyzing and optimizing adaptive modulation coding jointly with ARQ for QoS-guaranteed traffic," *IEEE Trans. Veh. Technol.*, vol. 56, no. 2, pp. 710–720, March 2007.
9. Y. J. Chang, F. T. Chien, and C. C. Kuo, "Cross-layer QoS analysis of opportunistic OFDM-TDMA and OFDMA networks," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 4, pp.657–666, May 2007.
10. M.-T. Sun and A. R. Reibman, Eds., *Compressed video over networks*. New York: Marcel Dekker, 2000.
11. Y. Dong, H. Chang, Z. Zou, and S. Tang, "An energy conserving routing algorithm for wireless sensor networks," *Int. J. Futu. Gen. Commun. Netw.*, vol. 4, no. 1, March 2011.
12. M. A. Matin, Ed., *Wireless sensor networks – technology and protocols*, InTech, September, 2012.
13. T. Holliday, A. Goldsmith, and P. Glynn, "Optimal power control and source-channel coding for delay constrained traffic over wireless channels," in *Proc. IEEE Int. Conf. Commun.*, vol. 2, pp. 831–835, May 2002.
14. F. Foukalas, V. Gazis, and N. Alonisioti, "Cross-layer design proposals for wireless mobile networks: a survey and taxonomy," *IEEE Commun. Surv.*, vol. 10, no. 1, pp. 70–85, 2008.
15. M. L. Puterman, *Markov decision processes-discrete stochastic dynamic programming*. New York: Wiley, 1994.
16. D. Djonin and V. Krishnamurthy, MIMO Transmission Control in Fading Channels – a Constrained Markov Decision Process Formulation with Monotone Randomized Policies, *IEEE Trans. Signal Process.*, vol. 55, no. 10 (2007), pp. 5069–5083.
17. M. Goyal, A. Kumar, and V. Sharma, Power Constrained and Delay Optimal Policies for Scheduling Transmission over a Fading Channel, in *Proc. IEEE Informcom.* (2003), pp. 311–320.
18. D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov chains and mixing times*, AMS, 2009.
19. V. Bhaskar, "Finite-state Markov model for lognormal, chi-square (central), chisquare (non-central) and K-distributions," *Int. J. Wirel. Inf. Netw.*, vol. 14, pp. 237–250, October 2007.
20. Q. Zhang and S. A. Kassam, "Finite-state Markov model for Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 47, no. 11, pp. 1688–1692, November 1999.
21. D. Rajaveerappa, "Design and development of a hybrid TDMA/CDMA MAC protocol for multimedia wireless networks," Ph.D thesis, April 2004.
22. D. P. Bertsekas, *Dynamic programming and optimal control*, 3<sup>rd</sup> ed., Belmont, MA: Athena Scientific, 2005.
23. Z. Yang, "An innovative unified platform for heterogeneous network communications," Technical report, TWL, 2014.
24. T. Zeybek and H. Chang, Delay Tolerant Network for Autonomous Robotic Vehicle Charging and Hazard Detection, *IEEE Int. Conf. Comput. Info. Sci.* (2014).
25. R. Gao and H. Chang, A scalable and flexible communication protocol in a heterogeneous network, *IEEE Int. Conf. Comput. Info. Sci.* (2014).
26. G. Pardo-Castellote, Omg Data-Distribution Service: Architectural Overview, *IEEE Int. Conf. Distributed Computing Systems Workshops* (2003).
27. Y. Dong and C. H. Chang, An Improved Autonomous Cross-Layer Optimization Framework for Wireless Multimedia Communication, *IEEE Int. Conf. Comput. Info. Sci.* (2014).
28. S. C. K. Lye, M. S. Arifanto, H. T. Yew, C. F. Liau, and K. T. K. Teo, "Performance of signal-to-noise ratio estimator with adaptive modulation," AMS, May 2012.
29. A. Zanella, Analysis of the Packet Reception Statistics of Bluetooth v2+EDR in Fading Channels (2007).
30. J. S. Roh, Performance Analysis and Evaluation of Bluetooth Networks in Wireless Channel Environment, in *Proc. Int. Conf. Syst. Netw. Commun.* (2006), pp. 61–65.
31. A. K. M. Najmul Islam and J. Gojoko, Bluetooth: Overview of Architecture, PHY and MAC, presentation.
32. Huawei Technologies Co., Ltd., "5G: a technology vision," 2013.
33. S. Kumar, D. Cifuentes, S. Gollakota, and D. Katabi, "Bringing cross-layer MIMO to today's wireless LANs," in *Proc. ACM SIGCOMM*, August 2013.