Cross-Layer Optimization for State Update in Mobile Gaming

Yang Yu, *Member, IEEE*, Zhu Li, *Senior Member, IEEE*, Larry Shi, *Member, IEEE*, Ethan Yi-Chiun Chen, and Hua Xu

***Abstract*—In a large-scale mobile gaming environment with limited wireless network bandwidth, efficient mechanisms for state update are crucial to allow graceful real-time interaction for a large number of players. By using the state updating threshold as a key parameter that bridges the resulting state distortion and the network traffic, we are able to study the fundamental traffic-distortion tradeoffs via both theoretical modeling and numerical analysis using real game traces. We consider a WiMAX link model, where the bandwidth allocation is driven by the underlying physical layer link quality as well as application layer gaming behaviors. Such a cross-layer optimization problem can be solved using standard convex programming techniques. By exploring the temporal locality of gaming behavior, we also propose a prediction method for on-line bandwidth adaptation. Using real data traces from a multiplayer driving game, TORCS, the proposed network-aware bandwidth allocation method (NABA) is able to achieve significant reduction in state distortion compared to two baselines: uniform and proportional policies.**

***Index Terms*—Bandwidth allocation, lagrangian relaxation, mobile gaming, state update, traffic-distortion tradeoffs, WiMAX.**

I. Introduction

MASSIVELY multiplayer online games (MMOG) are projected to be one of the rapidly growing entertainment services that will gain large popularity in near future. With the proliferation of feature rich and graphics capable mobile handsets, MMOG targeted for mobile clients will soon become a killer application for both mobile carriers and game service providers. In a mobile setting, a mobile client can connect to an MMOG server via wireless access points (e.g., WiFi or WiMAX). Each client remotely controls one or several in-game characters called *avatars* (e.g., a vehicle or a humanoid). States of the avatars are simulated by the hosting servers, with updates of avatar states broadcasted to other hosting servers and clients, so that all avatars can be properly rendered on the client side. To ensure the quality of interactivity of an MMOG requires a timely delivery of such state updates. Otherwise, players may be annoyed with various types of strange gaming experience, e.g., "dead-man shooting" in a shooting game [2]. This issue is also called "state-consistency" problem.

In contrast to wired communication, mobile communication is more constrained in network resources. Although 3G and the emerging WiMAX techniques promise to deliver higher data bandwidth than the existing mobile data services, MMOG in the mobile space still faces a number of technical challenges. In particular, a state update practice that is suitable for wired communication may not be feasible or desired in a wireless environment. For instance, if the servers blindly send updates based on simple quality metrics without considering the underlying network constraints, the large volume of state update can overstretch the buffers and queues in the wireless access points. This results in decreased consistency, when state update packets are delayed or dropped by the overloaded access points.

Therefore, the objective of this paper is to devise a network-aware state update mechanism that minimizes overall avatar state inconsistency between the servers and clients, subject to a bandwidth constraint specified by the underlying wireless access points. We describe our approach through an example MMOG driving game, named TORCS [3]. We capture the gaming inconsistency (or distortion) as the Euclid distance between the vehicles' actual location and the location interpreted by peer players. Via both theoretical modeling and numerical analysis of real game traces, we discover that distortion in vehicle location behaves as an decreasing and convex function of the updating frequency, which in turn translates linearly to network traffic. Such a function captures the fundamental traffic-distortion tradeoffs, which is the basis of our study.

We consider a WiMAX link model, which employs orthogonal frequency-division multiple access (OFDMA) for improved spectral efficiency. We treat each subchannel by an OFDMA symbol as a basic bandwidth allocation unit, referred to as a cluster. We consider adaptive modulation and coding (AMC) that adapts data rate according to physical layer link quality. To abstract the policy of WiMAX scheduler and interaction among multiple applications, we assume the total number of clusters available to the MMOG application is upper bounded by a pre-specified constraint, which translates to a bandwidth constraint at the peak data rate.

Thus, we model the target problem as given the physical layer link quality and application layer traffic-distortion tradeoffs of all clients, to derive an allocation of clusters to clients, so as to minimize the sum of distortion over all clients subject to the

Manuscript received July 31, 2007; revised November 2, 2007. Published July 9, 2008 (projected). This paper was presented in part at the International conference on Computer Communications and Networks (ICCCN), Honolulu, HI, August, 2007. This version contains substantially new results including the incorporation of WiMAX link model, a theoretical model for the traffic-distortion tradeoffs inherent to gaming behavior, and simulation results based on realistic WiMAX link quality traces. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Eckehard Steinbach.

Y. Yu and Z. Li are with the Application Research Center, Motorola Labs, Schaumburg, IL 60196 USA (e-mail: yang@motorola.com; zhu.li@motorola.com).

L. Shi was with Georgia Institute of Technology, Atlanta, GA 30332 USA. He can be reached by e-mail (e-mail: ccmod1937@yahoo.com).

E. Y.-C. Chen and H. Xu are with the Motorola Home and Networks Mobility, Arlington Heights, IL 60004 USA (e-mail: y.chen@motorola.com; hua.xu@motorola.com).

Digital Object Identifier 10.1109/TMM.2008.922777

constraint on available clusters. Such a cross-layer optimization problem can be solved using standard convex programming technique. We also propose a prediction-based real-time adaptation mechanism, referred to as the network-aware bandwidth allocation (NABA) policy, that exploits the temporal locality of gaming behavior.

The key advantage of our approach is to fully explore the user diversities in terms of both the link quality and traffic-distortion tradeoffs, so that the limited network bandwidth can be effectively utilized for better overall gaming experience. Compared to two naive solutions that employ either a uniform updating frequency or a fixed state changing threshold for all vehicles, our approach exhibits significant reduction in overall distortion, according to simulations using real game traces. Our real trace-based evaluation also demonstrate NABA's adaptability to variations in system parameters, including bandwidth constraint and network delay.

The contributions of this paper are multifold. We investigate a novel mechanism to characterize and predict the traffic-distortion tradeoffs of MMOG players. We also propose an efficient real-time adaptation mechanism for network-aware state update that minimizes overall gaming distortion, subject to a bandwidth constraint. Finally, we present performance study of NABA against two baselines using data traces from a real driving game.

The rest of the paper is organized as follows. We describe the system models in next section. In Section III, we study the traffic-distortion tradeoffs of a driving game. This is followed by a formal problem definition in Section IV. In Section V, we describe our network-aware bandwidth allocation approach in detail. In Section VI, we evaluate our approach using real game traces. We discuss related works in Section VII. Finally, we conclude the paper in Section VIII.

II. System Models

A. Network Infrastructure

We consider the scenario where an MMOG server provides gaming service to mobile clients via wireless access points. Local mobile players send avatar inputs to the server through wireless links. The server computes the next state of the avatars, sends the updated state back to a set of pre-specified clients (to be explained later) so that the avatars can be displayed at clients with correct state. Hereafter, we refer to updates from the clients to the server as *client updates*, and updates from the server to the client as *server updates*.

Since the computation performed at the server requires all client updates, it is difficult to reduce the up-link traffic without interfering with the basic processing logic of the game server. Thus, we focus on the downlink aspect of the problem, i.e., the server updates.

To improve scalability and reduce network traffic, concepts such as area of influences and area of interests are introduced [4]. Based on these concepts, a client only needs to receive state update of avatars that are within the area of influence of the client. These works are orthogonal to our work. Therefore, without loss of generality, we assume that the server needs to update the avatar state from a set of n clients to the same client set.

Also, to simplify the presentation, we assume these clients share the same wireless access point, thus experiencing the same bandwidth constraint and network delay. For games that are played by groups of localized people (e.g., audiences in a stadium and travelers on a subway), the players are likely to be connected to the same access point. For the case when multiple access points are involved to connect people in a wide area, we need to perform the proposed adaptation mechanism for every access point.

B. Wireless Link Model

We assume WiMAX (IEEE 802.16e) [5] as the underlying communication technique and establish our optimization framework based on the downlink model of WiMAX. WiMAX utilizes OFDMA, where a scheduler is employed at each access point to dynamically adapt the allocation of subchannels to users for improved system spectral efficiency. Specifically, the scheduler uses AMC technique to determine the modulation and coding scheme for each user based on their individual link quality, indicated by an effective carrier to interference-plus-noise ratio (CINR) measurement. A table-driven approach is often used to choose higher constellation size for better links. The scheduler then allocates OFDMA subchannels and symbols to users based on their application priorities.

An allocation generated by the scheduler determines the structure of the OFDMA frames. Fig. 1 demonstrates an example OFDMA frame structure, with N_s subchannels and M OFDMA symbols, out of which N is used for downlink. Each frame usually consists of 30 subchannels and 48 symbols, with a 5-ms frame duration. In Fig. 1, each square corresponding to a subchannel and a symbol is referred to as a cluster. For ease of analysis, we assume that a cluster is a basic unit of allocation.[1] In this example, 18 cluster are allocated to serve data burst #1.

It is challenging to comprehensively model the operation of a scheduler, taking into account the complex interaction between multiple applications. For tractable analysis, we abstract the impact of application priority and queue length using a prespecified upper bound, B, of the total number of clusters available to the gaming application. At the peak data rate, such a upper bound translates to a bandwidth constraint allocated to the gaming application, which is reasonable from inter-application QoS perspective.

We assume frequency-diverse subchannels, where each subchannel further consists of 24 data subcarriers that are pseudorandomly distributed across the bandwidth. Thus, at any particular time, the link quality of all subchannels allocated to a particular user are similar to each other. In other words, for each user, no quality differentiation between subchannels are considered.

C. Dead-Reckoning Algorithm

For the purpose of this paper, we limit avatar state to spacial location, which is one of the most common avatar states requiring frequent updates. For illustrative purpose, we use a multiplayer vehicle driving game, TORCS [3]. Since TORCS is open source-based, we modified the game source code to support up to 32 vehicles in a free driving mode, where each client

[1] Although two or three clusters are treated as a basic allocation unit in WiMAX specification, this variation does not affect the principle of our study.

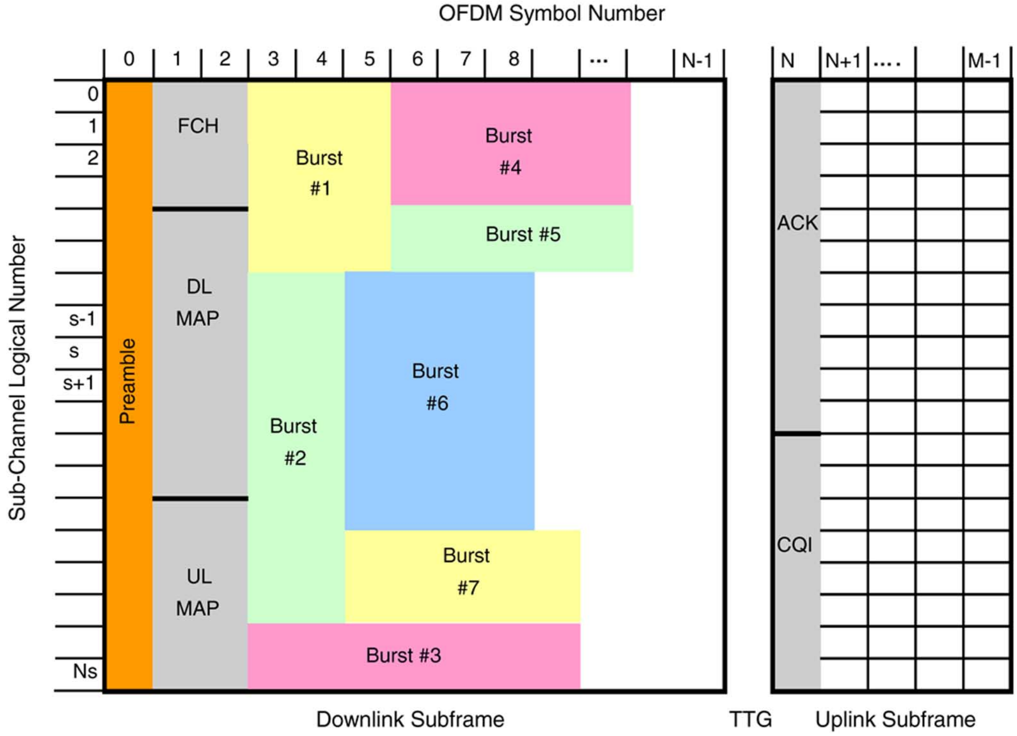


Fig. 1. Example OFDMA frame structure.

can drive a car in a 3-D virtual metropolitan [2] (the terms client and vehicle are thus used interchangeably hereafter). We also modified the source code to dump the location traces of cars at 500 samples per second. We then simulated the state updates between server and clients that are connected via a simulated wireless link by applying different updating thresholds to the location traces. The resulting state distortion and network traffic were recorded to be used for both our analysis and simulation results.

Both the server and the clients use the standard first order dead-reckoning approach to estimate the location of vehicles based on the position and velocity data in the latest server updates [6]. Consider state update for client A. Let $t_{\rm old}$ denote the time of the last server update for client A, with $l_{\rm old}$ and $v_{\rm old}$ being the location and velocity of A in the update. Let d denote the network delay of the simulated wireless links between server and the clients. We assume that d varies slowly over time and is fed back from the access point to the server in real-time. Based on [6], client A keeps updating the server about its vehicle's location at a fixed frequency f_s. The server tracks the location difference between vehicle A's server state and its estimated state at clients. A new server update is triggered at time t when the server receives a new client update on A with location $l_{\rm new}$ and velocity $v_{\rm new}$, such that

$$\|l_{\rm new} + v_{\rm new} \times d, l_{\rm old} + v_{\rm old} \times (t - t_{\rm old} + d)\| > \delta, \quad (1)$$

where δ is a pre-defined updating threshold and $\|x_1, x_2\|$ is a norm function that returns the Euclid distance between locations x_1 and x_2.

We use a temporally averaged location difference to evaluate the visual defects of the dead-reckoning algorithm. Specifically, every time the server receives a client update for A, it calculates the location difference for vehicle A. These location differences are accumulated over every second. The sum divided by the number of client updates within the second is regarded as the state distortion of client A. Moreover, the sum of distortion over all vehicles is regarded as the overall distortion of the game.

III. Traffic-Distortion Tradeoffs

Because of the difficulty in modeling vehicle mobility, it is challenging to directly characterize the tradeoffs between the updating traffic and the resulting state distortion. We, however, observe that both the updating frequency and the distortion depend on the threshold δ. Intuitively, a larger δ leads to less number of server updates, and very likely, resulting in a larger distortion; a smaller δ incurs larger number of server updates and hopefully less distortion.

A. Theoretical Intuition

We elaborate the above intuition using a first-order theoretical model. Specifically, suppose at time t_s and t_e, two consecutive server updates for vehicle A are sent. We consider the actual location of A as a function of time, denoted as $f(t)$. We approximate $f(t_e)$ using the first three items from the Taylor series at t_e[3]:

$$f(t_e) = f(t_s) + f'(t_s)(t_e - t_s) + \frac{f\check{Z}(t_s)}{2}(t_e - t_s)^2 + o((t_e - t_s)^2) \quad (2)$$

[2] There are open source, multiplayer first person shooting games that require both location and shooting actions as part of the key state information. To focus on the impact of location, we chose driving games in this study.

[3] While Taylor series is a simple example to illustrate the theoretical intuition, we expect more thorough study of the mobility model in the future.

where $f'(t_s)$ and $f''(t_s)$ are the first and second derivatives of $f(\cdot)$ at t_s, corresponding to the velocity and acceleration of vehicle A at t_s.

We now consider the estimated location of vehicle A at other clients. For the ease of presentation, we assume network delay $d = 0$ (our analysis easily extends to $d > 0$). From the server update at t_s, the clients have the knowledge of both $f(t_s)$ and $f'(t_s)$. Thus, the estimated location at t_e is $f(t_s) + f'(t_s)(t_e - t_s)$. For any time $t \in [t_s, t_e)$, the difference between the actual and estimated locations of A, $\kappa(t)$, is approximated as

$$\kappa(t) = \frac{f''(t_s)}{2}(t - t_s)^2. \quad (3)$$

Let a denote $f''(t_s)$. Let $\tau = (1/f_s)$ denote the period of client updates. Let l denote the number of client updates for vehicle A sent during $(t_s, t_e]$. Based on dead-reckoning, the server update at time t_e is triggered because $\kappa(t_e) \geq \delta$. We assume the equality case, which is reasonable for a sufficiently high f_s. We have $l = (1/\tau)\sqrt{(2\delta/a)}$. For every client update, the server records the location difference, denoted as $\kappa(t_i)$, for $i = 1, \ldots, l$. We derive the temporally averaged distortion, D, of vehicle A at other clients during $[t_s, t_e)$, as

$$D = \frac{\sum_{i=1}^{l} \kappa(t_i)}{l} = \frac{4\delta + 3\tau\sqrt{2a\delta} + a\tau^2}{12}. \quad (4)$$

Since τ is usually small, $a\tau^2$ becomes negligible.

Moreover, considering a long gaming interval, the expected value of $(1/t_e - t_s)$ can be treated as the expected frequency, F, of server updates. Recall that the condition to trigger a server update at t_e is $\kappa(t_e) = \delta$. We derive

$$F = E\left(\sqrt{\frac{a}{2\delta}}\right). \quad (5)$$

Overall, we observe that the state distortion, D, is close to a linear function of δ, while the frequency of server updates, F, behaves as a decreasing and concave function of δ.

B. Real Game Trace Verification

To verify our intuition, we take a 40-s car location trace from the TORCS game. We vary δ from 0.2 meter (m) to 10 m, in an increment of 0.2 m. For each δ, based on (1) with $d = 10$ ms, we numerically track the number of updates and the resulting distortion at a per second basis. We plot the data for the 18th second in Fig. 2. We observe that the number of updates decreases with δ, while the distortion increases almost linearly with δ. This confirms our theoretical intuition in Section III-A. Similar trends are observed for other time periods as well.

Using δ as a bridging parameter, we can now reveal the traffic-distortion tradeoffs. By assuming an average packet size of 200 bytes and network delay of 10 and 20 ms, we depict the tradeoffs for three cars in Fig. 3. It is clearly observed that in all six cases, the distortion behaves very closely as a decreasing and convex function of the updating traffic. This establishes the fundamental tradeoffs for the proposed bandwidth allocation mechanism. From information theory perspective, by treating state update as a sampling process, the traffic-distortion tradeoffs bears similarity to the well-known rate-distortion tradeoffs.

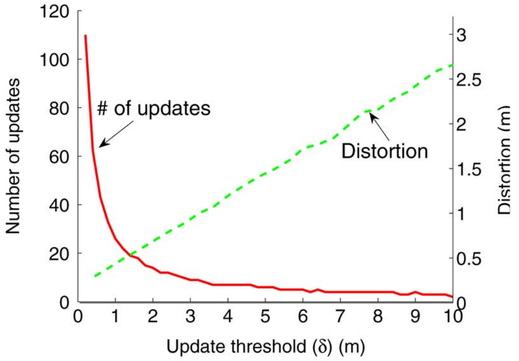


Fig. 2. Updating frequency and state distortion vs updating threshold.

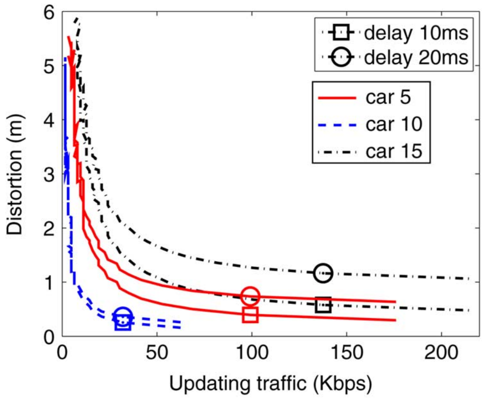


Fig. 3. Traffic-distortion tradeoffs from real driving game trace.

We also observe that for a given network delay, the curves differ for various cars, implying the necessity of exploiting the client diversity to minimize the overall state distortion. Also, for a given client, the curve changes with varied network delay. This indicates that a real-time adaptation is required in light of varying network status.

Note that throughout this section, we are concerned about the server update frequency, which may not accurately reflect the actual traffic due to variations in packet size resulting from data aggregation and compression. However, since we are interested in network traffic per second basis, we address this issue by assuming that the average packet size per server update converges over time.

IV. Problem Definition

Due to the real-time nature of gaming, our problem inherently requires an on-line adaptation solution that is invoked, for example, once every second to schedule the bandwidth allocation. However, this requires the knowledge of future traffic-distortion tradeoffs of gaming behavior. To handle this requirement, we use a historical data-based prediction that exploits the temporal locality of gaming behavior (to be detailed in Section IV). However, for a baseline analysis, we assume that the traffic-distortion tradeoffs for the entire gaming duration are known *a priori*. Thus, in this section, we focus on an off-line problem formulation.

Let n denote the number of clients and t denote the current gaming time. We consider bandwidth allocation for the tth second. For each client $i, i = 1, \ldots, n$, the updating threshold δ_i can be chosen from a given range Δ. By analyzing the data trace at the tth second, we derive the corresponding distortion function, $D_i(\delta_i)$, and traffic function, $R_i(\delta_i)$ for each client i.

As described in Section II-B, for each client, its link quality in terms of the effective CINR is fed back to the WiMAX server in real-time, upon which the modulation and coding scheme for the client is chosen accordingly. A key question in formulating the problem is that given the $R_i(\delta_i)$ bits to be transmitted in the next second, how shall these bits be distributed into all frames (e.g., 200 frames per second). This is particularly challenge when the link quality of future frames cannot be accurately predicted. In this paper, we take an initiative step by evenly distributing the traffic among all frames in one second.

Let α_i denote the expected constellation size (bits per symbol) of client i in the tth second. Let h denote the number of data subcarriers per subchannel. Also, let b_i denote the number of clusters allocated to client i by the scheduler at each frame. The sum of b_i's is upper bounded by a pre-specified B (while we assume a constant B during the tth second, B in general can vary over time due to variations in background traffic or preference to the gaming application). Moreover, let Q denote the OFDMA frame rate per second.

The bandwidth allocation problem for the tth second can be modeled as a convex programming problem as follows.

Given:

a. the distortion function, $D_i(\delta_i)$, and traffic function, $R_i(\delta_i)$, for n clients

b. the constellation size chosen for client i, α_i, and OFDMA parameters, Q and h

the bandwidth constraint, B in terms of the number of available clusters per frame,

find *vectors $\vec{\delta} = \{\delta_1, \delta_2, \ldots, \delta_n\}$ and $\vec{b} = \{b_1, b_2, \ldots, b_n\}$, so as to minimize*

$$\sum_{i=1}^{n} D_i(\delta_i) \tag{6}$$

subject to

$$\frac{R_i(\delta_i)}{Q} \le \alpha_i b_i h \tag{7}$$

$$\sum_{i=1}^{n} b_i \le B \tag{8}$$

$$b_i = 1, 2, \ldots \tag{9}$$

$$\delta_i \in \Delta, \quad i = 1, \ldots, n \tag{10}$$

A. Discussion

Although the above optimization problem is formulated using a WiMAX link model, the key tradeoffs between bandwidth usage and gaming distortion are not tied to any particular link models. Our previous publication [1] studies a generalized problem definition, which is omitted in this paper due to space limitations. Motivated by contemporary trends in both technology and business requirements, this paper studies the interesting case of WiMAX technique, which is expected to be widely deployed in near future.

Our problem formulation does not apply to users who sign contract with network service providers for guaranteed network performance, including dedicated bandwidth allocation policy. While such a contract is usually designed for critical business and civilian applications, we believe that in the context of providing leisure contents (including gaming), most users will adopt the normal residual contract that fits our bandwidth sharing model.

It is worth mentioning another practical policy for game access control, which specifies a minimal bandwidth requirement for the game players, and enforces the server to accept only qualified users. We believe that the method proposed in this paper can be employed as an optimization stage after first applying such an elimination-based policy. The interaction between the two methods is of future research interest.

V. Our Approach

We gear our analysis for a sufficiently large B, so that we do not concern ourselves with the integrality restriction of b_i. In practice, an integral but suboptimal solution can be derived by a proper rounding of the fractional b_i's.

Directly solving this problem in its primal form is difficult. Instead, we solve its dual problem via Lagrangian relaxation. Similar optimization decomposition techniques have been previously applied in solving communication and networking problems [7], [8].

We first combine the two constraints (7) and (8), so that

$$\sum_{i=1}^{n} \frac{R_i(\delta_i)}{\alpha_i} \le QBh. \tag{11}$$

The Lagrangian relaxation of the original problem is then

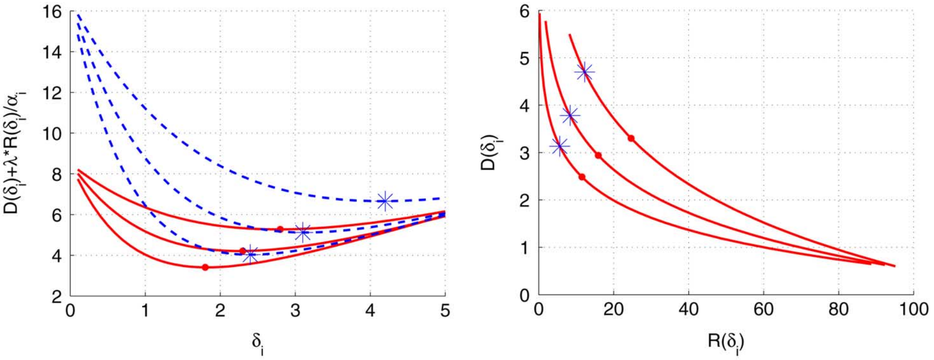
$$J(\lambda) = \min_{\delta_1, \delta_2, \ldots, \delta_n} \left\{ \sum_{i=1}^{n} D_i(\delta_i) + \lambda \sum_{i=1}^{n} \frac{R_i(\delta_i)}{\alpha_i} \right\} \tag{12}$$

where $\lambda > 0$ is the Lagrangian multiplier. The relaxed problem in (12) is separable. For each i, the relaxed problem is

$$J_i^{\lambda}(\delta_i) = \min_{\delta_i} \left\{ D_i(\delta_i) + \lambda \frac{R_i(\delta_i)}{\alpha_i} \right\}. \tag{13}$$

The Lagrangian multiplier λ controls the tradeoffs between the resulting state distortion and network traffic. For example, when the resulting network traffic is below the constraint B, it indicates that too much penalty is associated with network traffic in (13), and λ needs to be reduced to increase the bandwidth utilization. A fast binary search can quickly locate the optimal value of λ that results in the smallest distortion and the tightest bound on the constraint B. The process is illustrated in Fig. 4.

With a given λ, for each player, we solve the optimization problem that has unique solution on δ_i. The relaxed problems, $J_i^{\lambda}(\delta_i)$, for a 3-player scenario are shown in Fig. 4(a), for $\lambda = 0.08$ and 0.16 in solid and dotted curves, respectively. In Fig. 4(a), the dot and asterisk markers signify the optimal δ_i's for the relaxed problems, with the corresponding traffic and



(a)

(b)

Fig. 4. Illustration of Lagrangian decomposition solution. (a) Relaxed problem for each user. (b) Solutions on the traffic-distortion curves.

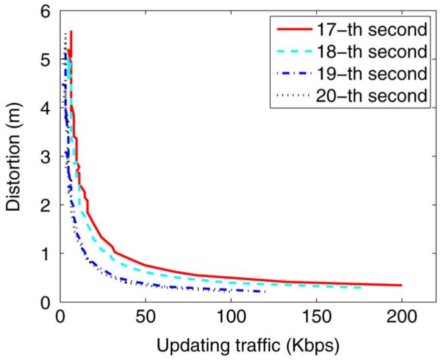


Fig. 5. Traffic-distortion tradeoffs of one vehicle in four consecutive seconds.

distortion demonstrated in Fig. 4(b). Notice that the relaxed problem for each player changes according to λ. When λ increases, more penalty is associated with network traffic, the optimal δ_i for each user moves towards larger update threshold, resulting in decreased traffic and increased distortion. This is illustrated in Fig. 4(b).

The above algorithm explores the space of λ using a binary search. For each specific λ, we need to minimize $J_i^\lambda(\delta_i)$ for all users by a binary search in the space of δ_i. Thus, the overall time complexity is $n \log \Lambda \log \Delta$, where Λ is the domain of λ and Δ is the domain of δ.

Given the above off-line analysis and optimization of the bandwidth allocation, we are further interested in an on-line adaptation mechanism for real-time game experience. To handle this challenge, we use a history-based prediction method that explores the temporal locality in gaming behavior, i.e., we expect the traffic-distortion tradeoffs for the coming second to be similar to those in the past seconds (refer to Fig. 5 for tradeoff curves for the same vehicle in four consecutive seconds of real game traces). Thus, the bandwidth allocation for the next second is based on the gaming behavior in the past seconds, and adapts with respect to variations in network status. We refer to this adaptive approach as the NABA policy.

While the prediction method is game-specific, we study various techniques for using the historical information via simulation (Section V). Our results indicated that for the studied driving game, simply using the tradeoffs in the previous second as history information delivered the best performance on average. This is also confirmed by Fig. 5, where the curves for each pair of consecutive seconds are closer to each other than other pairs.

Although our prediction technique is a best effort-based method when the temporal locality is weak, via simulation, we observed satisfactory performance on average. A more thorough study of the sensitivity of our technique to prediction accuracy is of our future interest.

A. Practical Concerns

When deriving the distortion and traffic functions for each client i, it is practically feasible to examine a finite set of values for δ_i in the domain of Δ. The same set of values are used to minimize $J_i^\lambda(\delta_i)$. The discretization of δ_i's affects the optimality of the above approach. However, by carefully choosing Δ to provide a sufficient resolution, it is practically reasonable to treat the resulting solution to be close to optimal. Thus, we ignore this issue in the following discussion.

Another concern is the rounding of b_i's. We adopt a randomized rounding technique. Specifically, for every fractional b_i, either the floor or ceiling of b_i is considered as the final solution, each with half probability. Accordingly, we need to tune δ_i so that the adjusted b_i can still accommodate the required bandwidth. Although details are not presented in this paper due to space limitations, we studied the impact of rounding via simulation. Compared to the case without rounding, we observed a slight increase ($<2\%$ averaged over time) in distortion for the case with rounding.

VI. Evaluation

A. Simulation Setup

We used data traces from the TORCS game to validate the usefulness of the NABA policy. We gathered data traces for 32 vehicles over a 40-s time period. The distortion and traffic functions, $D_i(\delta_i)$ and $R_i(\delta_i)$, of all vehicles are extracted by analyzing the data traces using a set of discrete values of δ, from 0.2 to 10 m, in an increment of 0.2 m.

We assumed an average packet size per server update of 200 bytes. To simulate link quality variations, we devised an experiment with three mobile devices placed at difference locations to communicate with the same WiMAX access point. The first

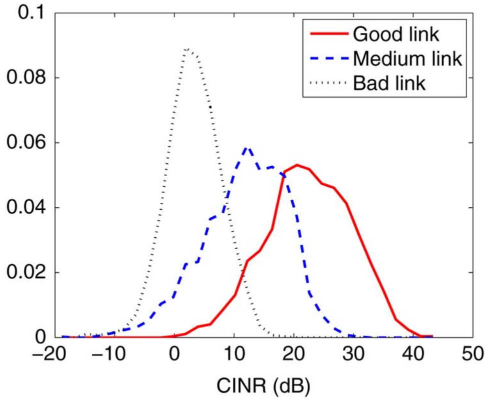


Fig. 6. CINR measurement distribution.

TABLE I
MAPPING FROM EFFECTIVE CINR TO CODING AND MODULATION SCHEMES

CINR (dB)	Modulation and coding scheme	Bits/symbol
≤ -2	QPSK, 1/2, repetition 6	0.1667
$(-2, 2]$	QPSK, 1/2, repetition 4	0.25
$(2, 8]$	QPSK, 1/2, repetition 2	0.5
$(8, 10]$	QPSK, 1/2	1
$(10, 14]$	QPSK, 3/4	1.5
$(14, 16]$	16-QAM, 1/2	2
$(16, 20]$	16-QAM, 3/4	3
$(20, 22]$	64-QAM, 2/3	4
> 22	64-QAM, 3/4	4.5

device was placed at the center of a sector with 0.1 km from the access point, which was expected to exhibit the best link quality. The second device was placed at the boundary of the sector with 0.5 km from the access point, expected to exhibit the worst link quality. The third device was placed at the center of the sector with 0.4 km from the access point, expected to exhibit a medium link quality. We gathered the CINR measurements at these devices at a rate of 200 samples per second, over a time duration of 15 s. In Fig. 6, we plot the distribution of the CINR measurements.

We then randomly determined the link quality of each client to be either good, medium, or bad. The actual CINR values were then generated based on the sample CINR measurements in Fig. 6. Specifically, consider a feedback period of 8 frames. We required in total $200/8 * 40 = 1000$ CINR measurements. For demonstration purpose, these measurements were randomly picked from the 3000 sampled measurements. The mapping of the CINR measurements to the modulation and coding scheme was performed based a pre-specified table [9] listed in Table I. We notice that the CINR measurements in Fig. 6 nicely covers the range of CINR in Table I. Thus, we believe the above measuring and sampling methods suffice to demonstrate the performance of NABA with respect to variations in CINR.

B. Baselines

Assuming the complete data traces known *a priori*, we were able to derive the optimal allocation in an off-line fashion using the techniques presented in Section IV. For a meaningful study, we also implemented two baseline policies. The first baseline assumed a uniform bandwidth allocation among all vehicles, regardless of their individual traffic-distortion tradeoffs. The second baseline assumed an identical updating threshold, δ, for all users. In this case, the bandwidth allocated to each vehicle was roughly proportional to its extent of state changes. Hereafter, we refer to these two baselines as the Uniform and Proportional policies, respectively.

Both baselines were performed in an off-line fashion. For the Uniform policy, in each second, we determined b_i's such that $b_i\alpha_i = b_j\alpha_j$, for all $i, j = 1, \ldots, n$, and $\sum b_i = B$. Then, for each vehicle i, we chose a corresponding δ_i so as to minimize the distortion of the vehicle, without violating the bandwidth constraint imposed by b_i. For the Proportional policy, in each second, iterating over every value in Δ, we calculate the number of clusters required for all vehicles with respect to their individual α_i's. We then chose the δ that utilized the maximal number of clusters within B. In both policies, the distortion of all vehicles corresponding to the chosen δ_i was then summed up as the overall distortion.

We consider the Uniform policy to be fair to all uses in terms of resource usage, and the Proportional policy to be fair in terms of gaming experience. The proposed technique achieves fairness in terms of resource efficiency, or marginal utility fairness, i.e., at the optimal allocation of resources so that all players have the same distortion gradient. This operation point is a good balance between resource and gaming experience fairness, and is widely adopted in network optimization [7], [8].

C. Main Results

In this set of simulations, we assumed a network delay of 10 ms and $B = 200$ available clusters per frame. With 24 subcarriers per subchannel and 200 frames per second, this constraint translated to a bandwidth constraint of 4.3 Mbps at peak data rate. We applied NABA, the optimal allocation, and the Uniform and Proportional polices to the game traces. For NABA, we used a 1-s history for predicting the traffic-distortion tradeoffs, with the first second for NABA implemented using the Proportional policy. The resulting state distortion and channel utilization (in terms of the ratio of allocated clusters over B) are illustrated in Fig. 7.

From Fig. 7(a), we observed that the distortion of NABA was within 18% off the optimal solution, throughout the entire time period except for the first second. Compared to the Uniform policy, NABA achieved 32% reduction in distortion averaged over time, with a maximum reduction of 46% at the 25th second. Compared to the Proportional policy, NABA achieved 30% reduction in distortion averaged over time, with a maximum reduction of 47% at the second second.

We also observed from Fig. 7(b) that both the optimal allocation and NABA achieved a channel utilization close to 1 (>0.99 averaged over time). The Proportional policy achieved an averaged channel utilization of 0.96, while the Uniform policy achieved 0.85. The under-utilization of channel resources was the main reason of increased distortion of the Uniform policies, due to its incapability of exploring user diversity in bandwidth requirements. For the Proportional policy, it failed to exploit the user diversity in terms of traffic-distortion tradeoffs. Thus, although it achieved high channel utilization, the resulting distortion was still unsatisfactory.

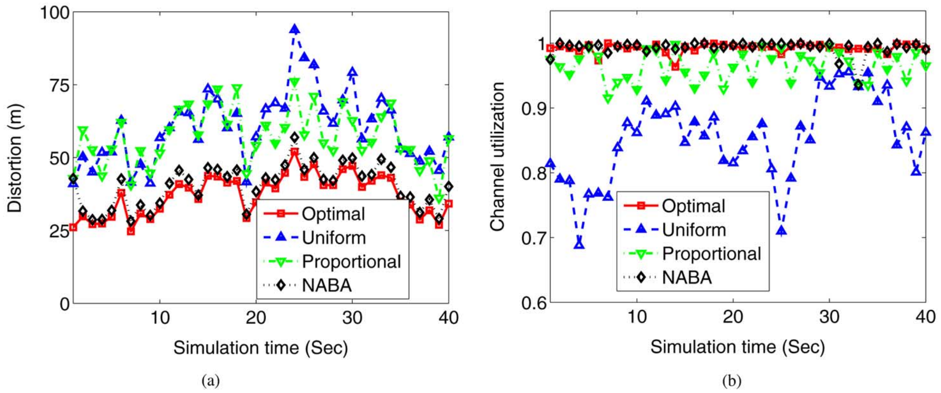


Fig. 7. Distortion and channel utilization comparison. (a) Distortion. (b) Channel utilization.

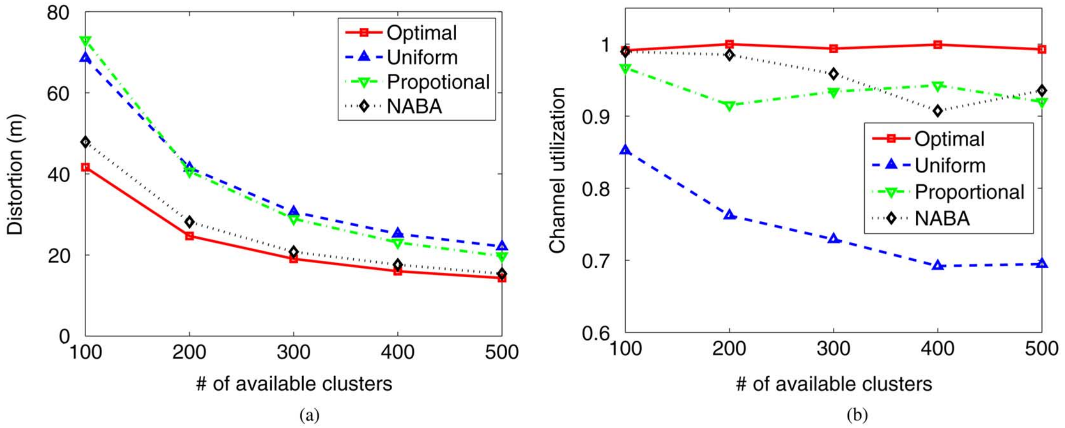


Fig. 8. Impact of available clusters. (a) Distortion. (b) Channel utilization.

Moreover, we noticed that for some time instants (e.g., the 31st second), the channel utilization of NABA was lower than that of the Proportional policy, while NABA still achieved less distortion. This highlighted the effective bandwidth utilization of NABA by exploiting the traffic-distortion tradeoffs of various clients. Similar reason explained why the channel utilization of the optimal solution was lower than that of NABA and the Proportional policy for some time instants.

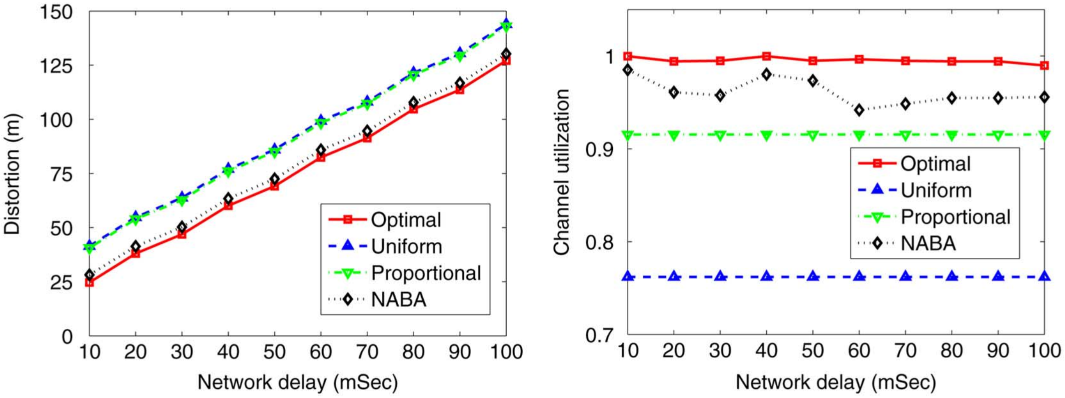
We further examined other techniques for history-based tradeoff prediction, including simple moving average and exponential moving average methods over multiple seconds. However, we discovered that although the impact of various techniques were not obvious, the 1-s history-based prediction achieved the best performance on average. This is understandable as in driving games, the state changes are usually limited in a short time window, indicating similar traffic-distortion tradeoffs in consecutive seconds. Nevertheless, the prediction methods are intuitively game-specific, and will be explored for other games in future.

We also performed the above simulations with other set of OFDMA parameters, including the number of subcarriers and the mapping between CINR to modulation schemes. Similar results were observed and thus omitted here.

D. *Impact of System Parameters*

For a wireless environment, the bandwidth constraint (in terms of available clusters) and the network delay usually vary over time. Thus, we also examined the impact of variations in bandwidth constraint and network delay on NABA. We varied the number of available clusters from 100 to 500, translating to a variation in peak bandwidth constraint from 2.2 to 10.8 Mbps. We fixed the network delay at 10 ms. In Fig. 8, we show the results for the four policies, with respect to the data trace of the 7th second. We observed that the distortion of all four policies decreased with bandwidth constraint. The distortion of NABA was consistently lower than the distortion of the Uniform and Proportional policies, approaching to the optimal solution as the number of available clusters increased.

For the same data traces, we also varied the network delay from 10 ms to 100 ms in an increment of 10 ms, while keeping a fixed $B = 200$. The results are shown in Fig. 9. From Fig. 9(a), the distortion for all four policies increased almost linearly with network delay. This was understandable based on (1). NABA achieved a distortion very close to the optimal throughout the variations in network delay. Also, NABA was able to keep a channel utilization closer 1 than the Uniform and Proportional policies.



(a)

(b)

Fig. 9. Impact of network delay. (a) Distortion. (b) Channel utilization.

VII. Related Work

There have been both empirical and theoretical studies in the distributed virtual environment and networked game communities on understanding the impact of network delay and packet loss on state consistency of simulated entities [10], [11]. Dead-reckoning is one of the most popular techniques so far and *de facto* standard for addressing the latency issue in distributed virtual and gaming environments [12], [13]. However, most research efforts in the past have focused on evaluating specific dead-reckoning approach [13], proposing new prediction technique [12], [14], or striving to solve the problem purely from the simulation domain. Our paper takes a holistic approach that brings state consistency and network resource management into the same picture. Our promising preliminary results show great potential of applying this holistic approach in solving the elusive quality of service problem between network resource management and state consistency.

There are also studies in characterizing and modeling game traffic in various context [15]–[18]. In particular, it is discovered that for the “Counter Strike” online game, while the in-bound packets to a game server have an narrow distribution centered around the mean size of 40 bytes, outgoing packets from the server have a much wider distribution around a significantly larger mean [17], [18]. Network traffic for the same game has also been modeled using the Extreme Value distribution [16]. It will be interesting to investigate the impact of such more realistic traffic characteristics and models on the proposed optimization method, and vice versa.

VIII. Conclusion

We have studied techniques for network-aware state update in massive player gaming environments. By using the updating threshold as a bridging parameter, we have theoretically revealed the fundamental traffic-distortion tradeoffs involved in state update. Our problem is based on a WiMAX link model, where user diversity in terms of link quality is incorporated into the optimization problem of bandwidth allocation to minimize the overall gaming distortion. An off-line optimal solution can be obtained via Lagrangian relaxation on a convex programming problem. For on-line adaptation, we have also proposed a prediction technique that exploits the temporal locality of gaming behavior. Using real data traces from a multiplayer driving game, we have verified the traffic-distortion tradeoffs, and also applied the proposed NABA policy, together with two baselines, the Uniform and Proportional policies. Compared to the two baselines, our results indicated a significant reduction in state distortion by using NABA, which is capable of efficiently exploring user diversity of both link quality and traffic-distortion tradeoffs.

In future, we plan to apply the proposed techniques in environments such as ad hoc networks, where distributed adaptation mechanism is required. We will also investigate the applicability of the techniques to other types of games.

Acknowledgment

The authors would like to thank the anonymous reviewers and Prof. W. Tsang Ooi for their valuable and constructive comments.

References

[1] Y. Yu, Z. Li, L. Shi, Y.-C. Chen, and H. Xu, “Network-aware state update for large scale mobile games,” in *Proc. Int. Conf. Computer Communications and Networks (ICCCN)*, Aug. 2007.

[2] M. Mauve, “How to keep a dead man from shooting,” in *Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services*, Oct. 2000.

[3] TORCS: The Open Racing Car Simulator 2007 [Online]. Available: http://torcs.sourceforge.net/

[4] S. Sinhal and M. Zyda, *Networked Virtual Environments*. New York: ACM Press, 1999.

[5] Mobile WiMAX—Part I: A Technical Overview and Performance Evaluation [Online]. Available: http://www.wimaxforum.org WiMAX Forum white paper.

[6] S. Goel and K. D. Morris, “Dead-reckoning for aircraft in distributed interactive simulation,” in *Proc. AIAA Flight Simulation Technology Conf.*, Aug. 1992.

[7] Z. Li, J. Huang, and A. K. Katsaggelos, “Pricing based collaborative multi-user video streaming over power constrained wireless down link,” in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, May 2006.

[8] M. Chiang, S. H. Low, A. R. Claderbank, and J. C. Doyle, “Layering as optimization decomposition: A mathematical theory of network architectures,” *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.

[9] CQICH channel clarification [Online]. Available: http://www.ieee802.org/16/maint/contrib/C80216maint-05_133.pdf

[10] S. Zhou, W. Cai, B.-S. Lee, and S. J. Turner, “Time-space consistency in large-scale distributed virtual environments,” *ACM Trans. Model. Comput. Simul.*, vol. 14, no. 1, pp. 31–47, 2004.

[11] T. Yasui, Y. Ishibashi, and T. Ikedo, "Influences of network latency and packet loss on consistency in networked racing games," in *ACM SIGCOMM Workshop on Network and System Support for Games (NetGames)*, Oct. 2005, pp. 1–8.

[12] L. Pantel and L. C. Wolf, "On the suitability of dead reckoning schemes for games," in *ACM SIGCOMM Workshop on Network and System Support for Games (NetGames)*, Apr. 2002, pp. 79–84.

[13] S. Aggarwal, H. Banavar, A. Khandelwal, S. Mukherjee, and S. Rangarajan, "Accuracy in dead-reckoning based distributed multi-player games," in *ACM SIGCOMM Workshop on Network and System Support for Games (NetGames)*, Aug. 2004, pp. 161–165.

[14] S. K. Singhal and D. R. Cheriton, Using a Position History-Based Protocol for Distributed Object Visualization Stanford Univ., Stanford, CA, Tech. Rep., 1994.

[15] M. S. Borella, "Source models of network game traffic," *Comput. Commun.*, vol. 23, no. 4, pp. 403–410, Feb. 2000.

[16] J. Faïrber, "Network game traffic modelling," in *ACM SIGCOMM Workshop on Network and System Support for Games (NetGames)*, Apr. 2002.

[17] K.-T. Chen, P. Huang, C.-Y. Huang, and C.-L. Lei, "Game traffic analysis: An MMORPG perspective," in *ACM Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV)*, Jun. 2005.

[18] W.-C. Feng, F. Chang, W.-C. Feng, and J. Walpole, "A traffic characterization of popular on-line games," *IEEE/ACM Trans. Netw.*, vol. 13, no. 3, pp. 588–500, Jun. 2005.

Yang Yu (M'01) received the B.S. and M.S. degrees in computer science from Shanghai Jiao Tong University, Shanghai, China, in 1996 and 1999, respectively, and the Ph.D. degree in electrical engineering from the University of Southern California (USC), Los Angeles, in 2005.

He is a Senior Staff Research Engineer at the Application Research Center of Motorola Labs, Schaumburg, IL. His research focus includes system modeling, algorithm design, and performance analysis for energy efficient information processing and routing in wireless sensor networks. He is co-author of *Energy-Efficient Information Processing and Routing in Wireless Sensor Networks* (Singapore: World Scientific, 2006).

Dr. Yu is a member of ACM.

Zhu Li (M'01–SM'07) received the Ph.D. degree in electrical and computer engineering from Northwestern University, Evanston, IL, in 2004.

He has been with the Multimedia Research Lab (MRL), Motorola Labs, Schaumburg, IL, since 2000, where he is a Principal Staff Research Engineer. His research interests include video coding and communication, game theory and optimization decomposition techniques in multimedia streaming and networking, manifold modeling and machine learning in biometrics, multimedia analysis, retrieval, and mining. He has ten issued or pending patents and more than 30 publications in book chapters, journals, and conference proceedings in his areas of research.

Dr. Li received the Best Poster Paper Award at the IEEE International Conference on Multimedia and Expo (ICME), Toronto, ON, Canada, in 2006, and the DoCoMo Labs Innovative Paper Award (Best Paper) at the IEEE International Conference on Image Processing (ICIP), San Antonio, TX, in 2007.

Larry Shi (M'01) received the B.Ed. degree from Beijing Normal University, Beijing, China, the Master's degree in psychology from Vanderbilt University, Nashville, TN, in 1999, and the Ph.D. degree in computer engineering from Georgia Institute of Technology, Atlanta, in 2006.

He is a frequent contributor to conferences and books related to computational entertainment and networked games. His research interests include high-performance computer architecture for large-scale multimedia applications, applied security, real-time computer graphics, networked games, and computational entertainment.

Ethan Yi-Chiun Chen received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, R.O.C., in 1988, and the M.S. and Ph.D. degrees from Northwestern University, Evanston, IL, in 1994 and 2000, respectively.

Since 1995, he has been with the Home and Networks Mobility Business Sector, Motorola, Arlington Heights, IL, where he is currently a Distinguished Member of Technical Staff in the Network Advanced Technologies R&D Group working on system architecture, standards, and performance analysis for the 4G wireless broadband systems, including WiMAX and LTE. His research experience and interests include performance evaluation, traffic engineering, simulation modeling, capacity planning, system architecture, and algorithm design.

Hua Xu received the B.S. degree in mathematics from Beijing University, Beijing, China, in 1985, and the M.S. and Ph.D. degrees in industrial and system engineering from Georgia Institute of Technology, Atlanta, in 1992.

She joined the Home and Networks Mobility Business Sector, Motorola, Arlington Heights, IL, in 1992, where she is a Distinguished Member of Technical Staff. Her current research interests are on radio resource management and system performance analysis for wireless networks.