Wireless Network Aware Cloud Scheduler for Scalable Cloud Mobile Gaming

Shaoxuan Wang, Yao Liu, Sujit Dey
Mobile Systems Design Lab, Dept. of Electrical and Computer Engineering
University of California, San Diego
shaoxuan@ece.ucsd.edu, yal019@ucsd.edu, dey@ece.ucsd.edu

***Abstract* — Cloud Mobile Gaming (CMG) [1][10][11] has been proposed as an approach to enable rich Internet games on mobile devices, where the rendering of the games is performed on cloud servers, as opposed to on mobile devices. Though promising, the CMG approach may require significant cloud computing resources for the concurrent gaming sessions, and even more critically, significant bandwidth for delivering the rendered videos back to mobile devices, leading to high cloud costs, and questions regarding resource constrained wireless networks. This paper addresses the problem of making the CMG approach scalable and economically feasible by proposing a novel Wireless Cloud Scheduler (WCS), which can increase the number of simultaneous CMG sessions that can be supported while ensuring Mobile Gaming User Experience (MGUE)[1] with the available wireless network resources, while minimizing the cloud service cost incurred by the CMG provider. Unlike conventional network schedulers, WCS considers simultaneously the constraints of the wireless networks that may be available to each CMG user, including cellular and WiFi, as well as the cost of available cloud resources, while scheduling the most optimal wireless link and cloud server for each CMG session. To further enhance the performance of WCS, we also propose a joint scheduling-adaptation algorithm, that can systematically leverage adaptation techniques introduced in [10][11] to adapt the communication needs of in-service users if the available wireless network bandwidth is not sufficient for a new CMG user. Our simulation results demonstrate that the use of WCS, and the joint scheduling-adaptation algorithm, can significantly improve the performance of the CMG approach, increasing the number of simultaneous CMG sessions that can be supported, while maximizing aggregate MGUE and minimizing the average cloud service cost.**

I. INTRODUCTION

With the tremendous growth in the market of mobile devices, there is a growing desire to enable 3D, multiplayer, Internet video games on smart phones. However, it is difficult for thin mobile devices to use the traditional gaming approach for rich 3D games, where the gaming clients execute the data, computation and energy intensive tasks such as 3D graphic rendering. Instead, a new Cloud Mobile Gaming (CMG) approach [1][10][11], where the responsibility of executing the gaming engines is put on CMG servers instead of the mobile devices, has the potential for enabling mobile users to play the same rich Internet games available to PC users.

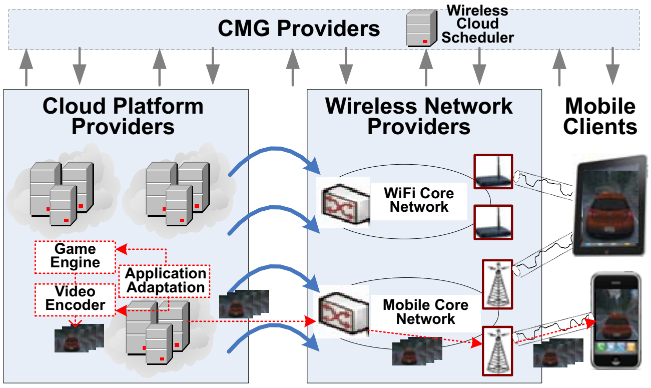

Fig. 1. Overall eco-system and architecture of cloud mobile gaming

While a CMG service can be launched in multiple ways, we assume the architecture and eco-system shown in Figure 1. The CMG provider "rents" cloud servers from cloud platform providers to host the CMG engines that need to be executed for each CMG session. We assume the mobile users have WiFi and/or cellular access provided by their wireless network provider, and paid by the mobile users according to the data rate plans. When a mobile user starts a CMG session, the CMG provider will first assign an appropriate wireless network and a proper CMG server, from amongst multiple choices available, taking into account availabilities in the cloud, and the CMG session requirements. Subsequently, the assigned CMG server will execute the CMG engine, starting with processing the game logic and user data to render raw game video, which is encoded and streamed to the mobile client over the wireless networks. On the other hand, the mobile user's gaming control commands are issued on his mobile device and delivered to the CMG server. In the meanwhile, as appropriately guided by the CMG provider, the application adaptation techniques, proposed in [10][11], can adapt the communication need of each CMG session to the dynamic conditions of the wireless network.

Though promising, there are two challenges we have to cope with to make CMG approach feasible. The first challenge is to ensure Mobile Gaming User Experience (MGUE) when using the CMG approach. In [1], we have developed a technique to quantitatively measure MGUE based on the results of subjective assessment studies. With this MGUE measurement technique, we have assessed the MGUE that can be achieved with current mobile networks. Our assessments show significant challenges to ensure MGUE in the mobile networks, considering (a) gaming video will have to be streamed from the CMG servers to the mobile devices through error-prone wireless networks, and (b) CMG is a highly interactive application, where round-trip latency will have to satisfy stringent response time requirements of gaming. Besides ensuring MGUE, scalability is another challenge for the success of the CMG approach, given that (a) compute intensive 3D rendering tasks for each concurrent CMG user may require high computing need and high video data rate, leading to a high cloud service cost, $\text{Cost}_{\text{CLOUD}}$ (the price paid by the CMG provider to the cloud platform provider for CMG sessions), albeit elastic cloud computing resources; and (b) CMG videos from millions of concurrent gaming sessions will have to be transmitted to mobile user devices over bandwidth

978-1-4577-2053-6/12/$31.00 ©2012 IEEE

constrained wireless networks, which may lead to non-availability of adequate bandwidth for some CMG users. To address the above challenges, in this paper we will design a Wireless Cloud Scheduler (WCS) for CMG provider and study several scheduling techniques, to investigate how to minimize average $Cost_{CLOUD}$ and how to support a large number of concurrent CMG sessions while ensuring acceptable MGUE for each user such that the aggregate MGUE is maximized.

The problems of cloud computation scheduling and network communication scheduling have been studied extensively but mostly considered in isolation. In [2][3][4], problems of assigning computing resources to a cloud user over a set of cloud computing servers in the cloud data centers are studied. The approach presented in [2] is mainly aimed at efficiently scheduling cloud tasks on multiple cloud servers, while [3][4] are targeted to alleviate the impacts of cloud resource fluctuation on service quality a cloud user perceives. On the other hand, scheduling and allocation of wireless network resources has been widely studied and incorporated in current wireless networks. For example, [5][6][7][8] propose various types of wireless network schedulers to allocate network bandwidth to users for different goals, including maximizing the aggregate throughput [6][7][8], achieving network fairness [5][7] and maintaining quality of service [8]. However, the above techniques do not consider the problem of scheduling considering simultaneously both cloud computing and wireless network resources, which is the goal of this paper. There has been one prior research which has considered scheduling computing tasks on mobile devices to remote processors, considering both the availability of the computing resources as well as the bandwidth availability of the mobile access networks [9]. However, the above technique needs knowledge of all the tasks that need to be scheduled and all the available resources to make a scheduling decision, and hence is not suitable for the CMG scheduling problem we address in this paper, where the set of CMG users can change dynamically, and rescheduling the entire set of users every time a new user requests service would be too computationally expensive. Moreover, the approach in [9] considers fixed computing resources, while CMG scheduling needs to consider elastic cloud computing resources, where the prime consideration is cloud cost and not the constraint on availability of computing resources.

We start by introducing the challenges and objectives of WCS in section II, where we have also formulated cloud service cost and introduced the different cloud instance needs of different CMG sessions. In section III, we describe a scheduling algorithm with three different utility functions: a MGUE-based utility function, a Cost-based utility function, and a MGUE/Cost-based utility function. We perform a set of simulation experiments to compare and characterize the performance of WCS with the three utility functions, and the performance of the original CMG approach where WCS is not applied. The simulation results show that WCS can help the CMG approach to achieve a higher aggregate MGUE and a lower $Cost_{CLOUD}$. We further enhance the performance of WCS by proposing a joint scheduling-adaptation algorithm in section IV, which can leverage the application adaptation techniques, adapting the communication and computation needs of in-service users, such that the aggregate MGUE is dramatically increased (by about 2.5 times) while $Cost_{CLOUD}$ is significantly reduced (by about 3 times). Section V summaries our findings in this paper.

TABLE I. CLOUD PRICING STRUCTURES OF RESERVING DIFFERENT AMAZON CLOUD EC2 INSTANCES FOR A YEAR

Cloud Instance Type	Computing Price (P_C)	Storage Price (P_S)	Network Price (P_N)	Has GPU
High-CPU Instance	$0.022 /EC2 CU per hour	$3.85e-8 /GB per second	$1.5e-8 /Kb	No
Cluster Compute Instance	$0.031 /EC2 CU per hour	$3.85e-8 /GB per second	$1.5e-8 /Kb	No
Cluster GPU Instance	$0.040 /EC2 CU per hour	$3.85e-8 /GB per second	$1.5e-8 /Kb	Yes

II. CHALLENGES AND OBJECTIVES OF WIRELESS CLOUD SCHEDULER FOR CMG PROVIDER

In this section, we explain the challenges and summarize the objectives of wireless cloud scheduler. The proposed WCS should firstly be able to schedule each requesting CMG user to a wireless network and a CMG server, so as to satisfy Quality of Service (QoS) requirements for an acceptable MGUE. According to our previous studies in [1], the major factors affecting MGUE are game response time, and video data rate associated with the quality of gaming video. For each game, there is a certain Response Time Acceptable threshold RT_A, above which user cannot accept the gaming quality. And for each CMG session, there is an associated computing resource requirement CMG_{Comp}, a storage space requirement $CMG_{Storage}$, and a communication video data rate requirement $CMG_{DataRate}$, which are determined by the type of game played and the game resolution used. Therefore, the QoS requirements of each CMG session consist of RT_A, CMG_{Comp}, $CMG_{Storage}$, and $CMG_{DataRate}$.

Second, though cloud computing resource is elastic and unlimited, the bandwidth of wireless mobile network is limited. It is possible that a user request cannot be scheduled due to the network bandwidth constraint. Thus for CMG service, there is a schedule rate, a term that denotes the percentage of the users who are actually being served in CMG system (while satisfying QoS requirements for each served user). For example, if the number of requesting customers is 100 and we can only schedule 90 of them, then the schedule rate is 90%. The WCS should achieve as high schedule rate as possible in a given network bandwidth constraint.

Third, unlike enterprise applications, CMG is an extremely compute intensive and network bandwidth demanding application, leading to a high $Cost_{CLOUD}$. Besides expensive, the cloud service prices for executing a certain computing need are different across different cloud instances. Table I presents the cloud pricing structures of reserving different Elastic Computing Cloud (EC2) instances provided by Amazon Web Services [12] for a year, including computing price (P_C), storage price (P_S), and network bandwidth price (P_N). Having known these cloud pricing structures, we can calculate the cloud service cost ($Cost_{CLOUD}$) by equation 1:

$$Cost_{CLOUD} = P_C \times CMG_{Comp} + P_S \times CMG_{Storage} + P_N \times CMG_{DataRate} \quad (1).$$

As we introduced before, CMG provider is charged by cloud platform provider based upon resource usage or reservation. Different types of game may have different requirements of cloud instance. For example, most games (e.g. flash games) can be executed in high CPU cloud instance, but for some games (e.g. racing games) fast response time is more crucial and hence they need a cloud instance with high I/O performance (such as cluster compute instance), while for some other games (e.g. MMORPG) rendering effect is more crucial and therefore they need a cloud instance with GPU (such as cluster GPU instance). To support different types of game, CMG provider may want to reserve different kinds of cloud instances in the cloud data center. Given this diversity, it will be important to consider the availability and cost of the diverse cloud computing resources, besides the wireless network resources, while scheduling the CMG sessions so that the average $Cost_{CLOUD}$ of millions of concurrent CMG sessions can be reduced.

More specifically, the proposed wireless cloud scheduler in this paper will achieve the following three objectives: 1) satisfying QoS requirements to meet acceptable MGUE for each scheduled user, 2) provisioning a high CMG schedule rate such that alleviating the obstacle of wireless network bandwidth constraints as much as possible, 3) reducing the average $Cost_{CLOUD}$. Motivated by these objectives, in the next section, we will propose a scheduling algorithm and study three different scheduling "utility functions" to investigate how to achieve the above three objectives.

III. Scheduling Algorithm for WCS

Traditional network and processor scheduler usually calculates the optimal solution by looking into the entire system resources and considering all the users/tasks, including new requesting users/tasks and in-service users/tasks. This optimal solution will very likely have to reschedule in-service users/tasks to the new resources. However, such scheduling approach may not be appropriate for CMG application. Firstly, because the CMG scheduler will need to cater to a very large number of CMG sessions using very distributed wireless network and cloud computing resources (as opposed to a base station scheduler caring about a few sectors, or a processor scheduler caring about a few cores or processors), it may be computation and time consuming to recalculate the optimal schedule every time a new CMG request comes. And more importantly, once a CMG session starts, it is very diffcult to reschedule the network access method and cloud server for this CMG session. Therefore, in this paper we will propose an in-service dynamic scheduling algorithm, which looks into the available resources to make the scheduling decision for the new users only, without affecting the in-service users. Based on this dynamic wireless cloud scheduling approach, we propose three different utility functions. We then conduct simulation experiments, with which we can characterize and analyze the benefits and deficiencies of each of the proposed utility function.

```
Input:   CMG_DataRate(i) for user i, available network bandwidth for any
         network B(m), utility function F(m,n), and game Type.
Output:  Selected network X and server Y for user i, if i can be scheduled.

Initial set  S = Φ;
For each network m
    For each CMG server n
        If B(m) >CMG_DataRate(i), and cloud instance in server n can support
            use requested game Type, and MGUE(m,n)>MGUE_MIN.
        Then S = S ∪ {θ(m,n)}; Endif
    Endfor
Endfor
If S equal to Φ
    Then user i cannot be scheduled; Exit;
Else
    Select X, Y, where
    F(X,Y) >= F(m,n)  ∀  θ(m,n) ∈ S; Exit;
Endif
```

Fig. 2. Scheduling algorithm

A. Scheduling Algorithm

Before we describe scheduling algorithm, we first define a scheduling utility function F(*m,n*), where *m* is the network selected, and *n* is the CMG server selected. Utility function F(*m,n*) is a key factor in deciding the optimal choice in scheduling algorithm. Different utility functions, described later, use the same scheduling algorithm, will lead to different objectives.

We next present the scheduling algorithm, shown in figure 2. The WCS will simultaneously monitor the resource being used by each CMG session. Then given any wireless network link *m* and CMG server *n*, we will know the available network bandwidth B(*m*) and cloud service cost $Cost_{CLOUD}(n)$. We can also measure network delay and server delay, thereby calculating the user perceived gaming quality MGUE(*m,n*) (MGUE for network *m* and cloud server *n*) using the model in [1]. Then given QoS requirements ($RT_A(i)$, $CMG_{Comp}(i)$, $CMG_{Storage}(i)$ and $CMG_{DataRate}(i)$) of user *i*, we first decide the possible solution set S. Each element θ(*m,n*) in S indicates the choice of network *m* and server *n*, which satisfies the following three requirements: 1) the network bandwidth B(*m*) is greater than $CMG_{DataRate}(i)$; 2) the cloud instance in cloud server *n* can support the game type requested by the CMG session; 3) mobile gaming user experience MGUE(*m,n*) is greater than minimum acceptable MGUE, $MGUE_{MIN}$, which is 3.0 according to [1]. Then we select the optimal choice of network *X* and server *Y* in S, which can maximize the value of the scheduling utility function F(*m,n*). Note that the requesting user *i* may not be able to be scheduled, if we cannot find any qualified network and server to satisfy its QoS requirements. In such case, his/her MGUE is 0, and the schedule rate will drop accordingly.

We next describe three different scheduling utility functions. The first scheduling utility function is targeted to maximize the MGUE of each assigned user. It is termed **MGUE-based utility function** ($F_1(m,n)$), and its value is equal to MGUE achieved:

$$F_1(m,n) = MGUE(m,n) \tag{2}.$$

The second scheduling utility function is called **Cost-based utility function** ($F_2(m,n)$), which takes into account both network bandwidth utilization Util(*m*) and cloud service cost $Cost_{CLOUD}(n)$. Before giving the Cost-based utility function, we first define network cost ($Cost_{Network}$), which indicates the impact of choosing a wireless network. As the wireless network bandwidth is limited, the WCS should not choose the network if its utilization is already very high, because increasing the utilization of the highly utilized network link may hurt the schedule rate in the long run. Due to this property, we create the relationship function between $Cost_{Network}$ and network utilization as shown in equation 3:

$$Cost_{Network}(m) = \frac{1}{1-Util(m)} \times CMG_{DataRate} \tag{3}.$$

Given a certain CMG communication video data rate $CMG_{DataRate}$, an increase of network utilization will result in the increase of network cost. And the increasing slop of network cost will be extremely high if the network utilization is close to 100%. Having defined Cost$_{\text{Network}}$, we next define a cost function Cost(m,n) (equation 4) which includes both Cost$_{\text{Network}}$ and Cost$_{\text{CLOUD}}$, and let Cost-based utility function F$_2$(m,n) equal to the reciprocal of Cost(m,n) (equation 5) as we want to minimize Cost(m,n) by using utility function F$_2$(m,n).

$$Cost(m,n) = Cost_{Network}(m) + \lambda \times Cost_{CLOUD}(n) \quad (4).$$

$$F_2(m,n) = 1 / Cost(m,n) \quad (5).$$

It should be also noted that the coefficient λ in equation 4 indicates the relative weight of Cost$_{\text{Network}}$ and Cost$_{\text{CLOUD}}$ in affecting the utility function F$_2$(m,n).

The third scheduling utility function considers both MGUE and costs (including wireless network and cloud service costs) simultaneously, maximizing the MGUE/Cost ratio. We term it as **MGUE/Cost-based utility function**:

$$F_3(m,n) = MGUE(m,n) / Cost(m,n) \quad (6).$$

B. Experimental Setup

To characterize and compare the performance of the above three utility functions, we developed a Matlab based simulation framework, which consists of several geographical regions with different coverage for either WiFi network, or a Cellular network, or both, with different bandwidth and delay characteristics, and a set of CMG servers each with a certain Cost$_{\text{CLOUD}}$. The framework allows the generation of CMG user requests with different network bandwidth and computing need (depending on the game session the user wants to play), different network coverage (access to WiFi only, Cellular only, or both WiFi and Cellular), and different Cost$_{\text{CLOUD}}$. While we have tested the proposed scheduling algorithm with multiple scenarios, figure 3 shows one such scenario based on which we present and discuss the experimental results in this paper. Note that our observations of the experimental results are valid for other test scenarios we have simulated.

In the test scenario shown in figure 3, we simulate 9 geographic regions, with 40% of the area in each region having both Cellular and WiFi network coverage, while 10% of the area having only WiFi network (areas with insufficient cellular coverage), and the rest 50% area has only Cellular network. We configure both WiFi bandwidth and Cellular bandwidth of each region between 8~10Mbps. We use three CMG servers with different cloud server instances as described in Table I. Also indicated are the network delays used in the simulation.

When the simulation starts, we randomly add a user into one of the 9 regions, within any of the three communication (WiFi only, Cellular only, and WiFi/Cellular) areas. (We assume the user's mobile device can support both WiFi and cellular, but is bound by one of the three communication modes above depending on which area he/she is in and the network availability in that area.) While we keep increasing the number of users in the simulated CMG system, we measure and calculate the schedule rate, average Cost$_{\text{CLOUD}}$, and aggregate MGUE. With those data, we have calculated the Average aggregate MGUE (MGUE$_{\text{A}}$) for all three utility functions by applying a Gaussian probability distribution for the number of user requests. We also compare the results of WCS (with the three utility functions) with the performance of the original CMG approach (without WCS), where we assume the wireless link and cloud server used is selected randomly.

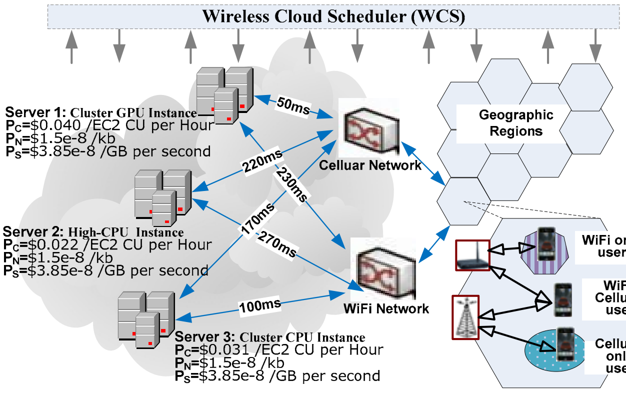
Wireless Cloud Scheduler (WCS)
Server 1: Cluster GPU Instance
P_C=$0.040 /EC2 CU per Hour
P_N=$1.5e-8 /kb
P_S=$3.85e-8 /GB per second
50ms
220ms
230ms
170ms
270ms
100ms
Cellular Network
WiFi Network
Server 2: High-CPU Instance
P_C=$0.022 /EC2 CU per Hour
P_N=$1.5e-8 /kb
P_S=$3.85e-8 /GB per second
Server 3: Cluster CPU Instance
P_C=$0.031 /EC2 CU per Hour
P_N=$1.5e-8 /kb
P_S=$3.85e-8 /GB per second
Geographic Regions
WiFi only user
WiFi/ Cellular user
Cellular only user

Fig. 3. Simulation experiment setup

C. Results and Conclusions

The results of the simulation experiments are shown in figure 4. Figure 4(a) presents the relationship between the schedule rate (the percentage of simultaneous CMG sessions that can be scheduled) and the number of users that has requested CMG service. From the results in figure 4(a), we note that WCS may not be able to schedule some of the incoming users when we keep randomly adding users into the CMG system. This is mainly because some requesting users have entered into the regions where network bandwidth is fully utilized. It can be also observed from figure 4(a) that the schedule rate if using F$_2$(m,n) or F$_3$(m,n) is always better than using F$_1$(m,n). This is because F$_1$(m,n) (MGUE-based utility function) has not considered the utilization of wireless networks. For example in figure 3, with utility function F$_1$(m,n), every user who has access to both WiFi and Cellular network will be scheduled to use Cellular network since it offers smaller network delay and therefore better gaming experience. Cellular networks will be quickly fully utilized, so that the next coming users, who only have access to Cellular networks, will not be able to be scheduled. This problem will be

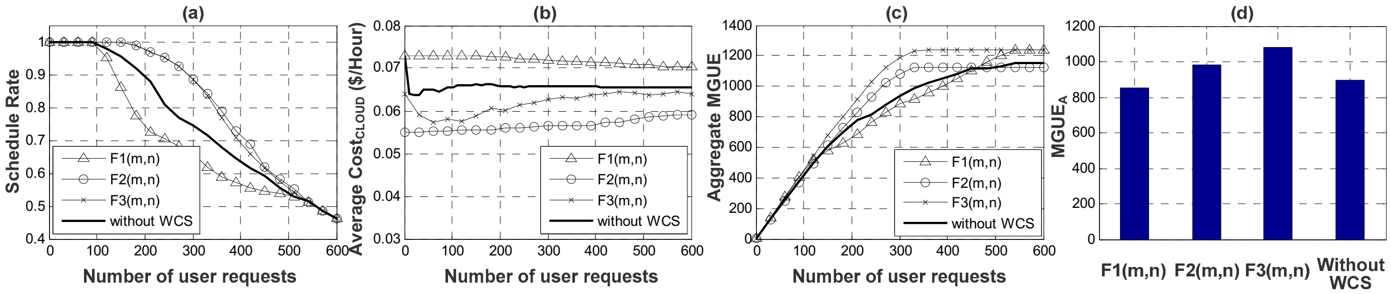

Fig. 4. Results of simulation experiment, including (a) schedule rate, (b) average Cost$_{\text{CLOUD}}$, (c) aggregate MGUE, and (d) MGUE$_{\text{A}}$, for CMG without WCS and CMG with applying scheduling algorithm with three different utility functions.

much alleviated by using $F_2(m,n)$ or $F_3(m,n)$, as they will assign some WiFi/Cellular users to WiFi networks when the utilization on Cellular network becomes high. In the test scenario of CMG without WCS, because the network link and cloud server are selected randomly without considering bandwidth utilization of wireless links, its schedule rate is worse than $F_2(m,n)$ and $F_3(m,n)$, but better than $F_1(m,n)$.

Figure 4(b) shows the results of average $Cost_{CLOUD}$ for scheduled users. As we expected, utility function $F_2(m,n)$ has the best performance in reducing $Cost_{CLOUD}$ among the three utility functions, while using $F_3(m,n)$ always has less $Cost_{CLOUD}$ than using $F_1(m,n)$. Both $F_2(m,n)$ and $F_3(m,n)$ produce lower $Cost_{CLOUD}$ than CMG approach without WCS.

We have also measured the MGUE of each scheduled user, and calculated the aggregate MGUE during the simulation experiment. As shown in figure 4(c), the aggregate MGUE increases when the number of user requests increases. However, the slope of the aggregate MGUE will keep decreasing to 0, where we cannot schedule any mobile user more. We tabulate the values in figure 4(c) into a function MGUE(x). Therefore, for any given number of user requests, x, the corresponding value of aggregate MGUE, *i.e.* MGUE(x), is shown in figure 4(c). Then we use an Average aggregate MGUE ($MGUE_A$), the weighted average of all possible values that aggregate MGUE can take on, to evaluate the performance of WCS. As similar to most service simulations, we use Gaussian distribution as the probability distribution of the number of user requests. It is presented in equation 7, where its mean is 300 and its standard deviation is 100, considering the maximum MGUE is achieved when the number of user requests is around 600 as shown in figure 4(c). Then the $MGUE_A$ can be calculated by equation 8, and the results of $MGUE_A$ are presented in figure 4(d).

$$P(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-u)^2}{2\sigma^2}} \quad (u = 300, \sigma = 100) \qquad (7).$$

$$MGUE_A = \sum_{x\in[1,600]} P(x) \times MGUE(x) \qquad (8).$$

From the above results and discussions, we can conclude that $F_3(m,n)$ is an ideal utility function for scheduling algorithm, as it can achieve better performance in all criteria including schedule rate, $MGUE_A$ and $Cost_{CLOUD}$, than the original CMG approach (without WCS). However there may be scenarios when either $F_1(m,n)$ or $F_2(m,n)$ may be preferred over $F_3(m,n)$. For example, if the goal is to maximize the MGUE for each scheduled user, $F_1(m,n)$ may be preferred. Similarly, if minimizing $Cost_{CLOUD}$ is the main objective, then $F_2(m,n)$ may be preferred. It should be also noted that for other simulation scenarios the results in figure 4 might change, however the relative performances and the conclusions of the scheduling algorithm in regards to schedule rate, average $Cost_{CLOUD}$, and $MGUE_A$, as described above, remain the same.

Though using scheduling algorithm (with MGUE/Cost-based utility function) is promising to help WCS to improve the aggregate MGUE and reduce cloud service cost, the number of users that can be assigned in CMG system is still limited due to the limited bandwidth of mobile wireless network. This leads to a limited $MGUE_A$. To further improve $MGUE_A$, in the next section we will propose a joint scheduling-adaptation algorithm which can significantly increase the number of users that can be assigned in CMG system while maintaining acceptable MGUE for each user. In the meanwhile, because adaptation techniques can reduce the CMG_{Comp} and $CMG_{DataRate}$ needed, it will also help CMG provider to achieve a very low $Cost_{CLOUD}$.

```
Input:  CMG_DataRate(i) for user i, available network bandwidth for any
network B(m), and utility function F(m,n).
Output: Selected network X and server Y for user i, if i can be scheduled.

   Initial set  S = Φ;
   For each network m
      For each CMG server n
         If  cloud instance in server n can support user requested game
            Type, and MGUE(m,n)>MGUE_MIN.
            Then S = S ∪ {θ(m,n)}; Endif
      Endfor
   Endfor
Loop:
   If  S Equal to Φ
      Then user i cannot be scheduled; Exit; Endif
   Select X, Y, where
      F(X, Y) >= F(m, n)  ∀ θ(m,n) ∈ S.
   If  B(X) > CMG_DataRate (i)
      Then schedule X, Y to user i;
   Else
      If  B(X)< CMG_DataRate (i)
         If  adaptation level on X is at its lowest level
            Then delete θ(X,Y) in S; Goto Loop;
         Else  reduce adaptation level of user on network X for one level;
         Endif
      Endif
      schedule X, Y to user i; Exit;
   Endif
```

Fig. 5. Joint scheduling-adaptation algorithm

IV. Joint Scheduling-Adaptation Algorithm for WCS

We next introduce a joint scheduling-adaptation algorithm, which can be applied for all the three utility functions described in section III. Our approach is to increase the number of users that can be assigned in CMG system by leveraging video and rendering adaptation techniques [10][11] to reduce QoS requirements of the CMG users while performing scheduling. Gaming aware video encoding adaptation introduced in [11] can adapt the video bit rate needed by a gaming session, thereby reducing the communication requirement of the session. Similarly, gaming rendering adaptation was introduced in [10] to enable adapting graphic rendering complexity thereby affecting both the computing and communication needs of a CMG session. For these adaptation techniques, we have an associated adaptation level from k to 1, where the level 1 has the lowest CMG_{Comp} and $CMG_{DataRate}$ requirement, and level k has the highest CMG_{Comp} and $CMG_{DataRate}$ requirement.

The mechanism of joint scheduling-adaptation algorithm is shown in figure 5, and is briefly described below. For a requesting user i, the joint scheduling-adaptation algorithm will first find the possible solution set S. Each element $\theta(m,n)$ in S indicates the network m and server n which can satisfy the minimum acceptable MGUE ($MGUE_{MIN}$) [1] of user i. Having known S, WCS will try to assign user i to the optimal choice, network X and server Y, which can achieve the maximum value of utility function $F(X,Y)$. Application adaptation technique will be involved when the resource of network X cannot meet QoS requirements of user i. However, it might be possible that the adaptation level is already in the lowest level, level 1, below which user cannot accept CMG quality. If this happens, we will delete element $\theta(X,Y)$ in S, and try to schedule the next optimal choice in S for user i.

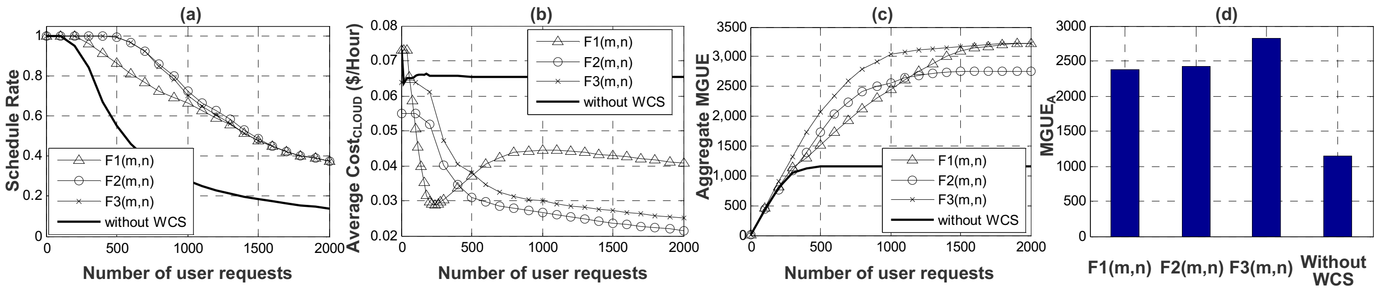

Fig. 6. Results of simulation experiments, including (a) schedule rate, (b) average Cost$_{CLOUD}$, (c) aggregate MGUE, and (d) MGUE$_A$, for CMG without WCS, and CMG with applying joint scheduling-adaptation algorithm with three different utility functions.

To demonstrate the efficiency of the proposed joint scheduling-adaptation algorithm, we use a similar simulation framework as described in section III-B. Figure 6 presents the experimental results of CMG using joint scheduling-adaptation algorithm with three utility functions and CMG without using WCS. From figure 6(a) we can observe that giving the same number of user requests the achieved schedule rate with applying joint scheduling-adaptation algorithm is much better than the schedule rate of CMG without WCS. For example, in figure 6(a), without the use of WCS, the schedule rate starts to reduce at about 100 requests, while with the use of $F_2(m,n)$ and $F_3(m,n)$, the schedule rate starts to reduce at about 500 user requests, . Figure 6(b) presents the average Cost$_{CLOUD}$ for each user. While the number of user request is increasing, the utilization of network bandwidth increases. When the available network bandwidth cannot meet a user request, we lower the CMG$_{Comp}$ and CMG$_{DataRate}$ on the scheduled users by adaptation techniques. This leads to a significant reduction (up to 3 times) of the average Cost$_{CLOUD}$ compared to CMG without WCS. Similarly, based on the results in figure 6(c)(d), we can observe that when applying joint scheduling-adaptation algorithm, the MGUE$_A$ can increase more than 2.5 times. The above results demonstrate that the proposed joint scheduling-adaptation algorithm can successfully reduce the average cloud service cost, and increase the number of users that can be scheduled in CMG system while ensuring the MGUE of assigned users such that the MGUE$_A$ is significantly enhanced. Besides these improvements, the other test results and conclusions are the same as what we have presented in section III-C, that using $F_2(m,n)$ and $F_3(m,n)$ can achieve a better schedule rate than using $F_1(m,n)$; using $F_2(m,n)$ is the best in terms of achieving the minimum Cost$_{CLOUD}$; and using $F_3(m,n)$ is still the ideal solution among three utility functions as it achieves the maximum MGUE$_A$ while having a relatively low (though not lowest) Cost$_{CLOUD}$.

V. Conclusions and Future Work

In this paper, we proposed a Wireless Cloud Scheduler (WCS), which can dynamically and intelligently schedule the wireless network and cloud server resources to requesting CMG users, in a dynamically changing and heterogeneous CMG environment. We first give a scheduling algorithm with three different utility functions, including a MGUE-based utility function, a Cost-based utility function, and a MGUE/Cost-based utility function, with which WCS can properly allocate resources to meet user QoS requirements. We have conducted a set of simulation experiments to compare the performance between CMG approach without WCS and CMG approach with WCS. Our simulation results demonstrate that WCS proposed in this paper can help the CMG approach to achieve a higher Average aggregated Mobile Gaming User Experience (MGUE$_A$) and a lower cloud service cost (Cost$_{CLOUD}$) than the original CMG approach without WCS. In order to further reduce Cost$_{CLOUD}$ and maximize MGUE$_A$, we introduce a joint scheduling-adaptation algorithm where user's communication and computation requirements can be adaptively adjusted according to the dynamic conditions of the wireless network at any time. Simulation results show that, the proposed joint scheduling-adaptation algorithm can significantly increase the number of concurrent cloud mobile gaming users under the same communication and QoS constraints thereby maximizing MGUE$_A$, and dramatically reducing the Cost$_{CLOUD}$ to the CMG provider. In the future, we plan to enhance our scheduling techniques to take into consideration the dynamic changes in wireless network conditions during already scheduled CMG sessions.

Acknowledgements

This research was partially supported by a research gift from Cisco, Inc.

References

[1] S. Wang, S. Dey, "Modeling and Characterizing User Experience in a Cloud Server Based Mobile Gaming Approach," in *Proc. of IEEE GLOBECOM,* Honolulu, Dec. 2009.

[2] P. Armstrong, et al, "Cloud Scheduler: a resource manager for distributed compute clouds," arXiv:1007.0050v1 [cs.DC], 2010.

[3] R. Nathuji, et al, "Q-clouds: managing performance interference effects for QoS-aware clouds", *in Proc. 5th European conference on computer systems,* pp. 237–250. ACM, New York, 2010.

[4] T. Cucinotta, et al. "Providing performance guarantees to virtual machines using realtime scheduling", in *in Proceedings of the 5th Workshop on Virtualization and High-Performance Cloud Computing,* Italy, Aug. 2010.

[5] N. Vaidya, et al, "Distributed Fair Scheduling in a wireless LAN," in *IEEE Transaction on Mobile Computing*, vol. 4, 2005.

[6] A. Shieh, et al, "Sharing the data center network", in *Proceedings of the 8th USENIX conference on Networked systems design and implementation*, 2011.

[7] P. Chaporkar, et al, "Throughput and fairness guarantees through maximal scheduling in wireless networks," in *IEEE Transactions on Information Theory*, 2008.

[8] J. Elias, et al, "A new approach to dynamic bandwidth allocation in Quality of Service networks: Performance and bounds"**,** in *International Journal of Computer and Telecommunications Networking*, 2007.

[9] S. Mukhopadhyay, C. Schurgers, S. Dey, "Joint Computation and Communication Scheduling to Enable Rich Mobile Applications," in *Proc. of IEEE GLOBECOM,* Washington D.C., Nov. 2007.

[10] S. Wang, S. Dey, "Rendering Adaptation to Address Communication and Computation Constraints in Cloud Mobile Gaming," in *Proc. of IEEE GLOBECOM,* Miami, Dec. 2010.

[11] S. Wang, S. Dey, "Addressing Response Time and Video Quality in Remote Server Based Internet Mobile Gaming," in *Proc. of IEEE WCNC,*Sydney, Mar. 2010.

[12] Amazon EC2 Pricing, http://aws.amazon.com/ec2/pricing/.