Innovative Schemes for Resource Allocation in the Cloud for Media Streaming Applications

Amr Alasaad, *Member, IEEE*, Kaveh Shafiee, *Member, IEEE*, Hatim M. Behairy, *Member, IEEE*, and Victor C.M. Leung, *Fellow, IEEE*

Abstract—Media streaming applications have recently attracted a large number of users in the Internet. With the advent of these bandwidth-intensive applications, it is economically inefficient to provide streaming distribution with guaranteed QoS relying only on central resources at a media content provider. Cloud computing offers an elastic infrastructure that media content providers (e.g., Video on Demand (VoD) providers) can use to obtain streaming resources that match the demand. Media content providers are charged for the amount of resources allocated (reserved) in the cloud. Most of the existing cloud providers employ a pricing model for the reserved resources that is based on non-linear time-discount tariffs (e.g., Amazon CloudFront and Amazon EC2). Such a pricing scheme offers discount rates depending non-linearly on the period of time during which the resources are reserved in the cloud. In this case, an open problem is to decide on both the right amount of resources reserved in the cloud, and their reservation time such that the financial cost on the media content provider is minimized. We propose a simple—easy to implement—algorithm for resource reservation that maximally exploits discounted rates offered in the tariffs, while ensuring that sufficient resources are reserved in the cloud. Based on the prediction of demand for streaming capacity, our algorithm is carefully designed to reduce the risk of making wrong resource allocation decisions. The results of our numerical evaluations and simulations show that the proposed algorithm significantly reduces the monetary cost of resource allocations in the cloud as compared to other conventional schemes.

Index Terms—Media streaming, cloud computing, non-linear pricing models, network economics

1 Introduction

Media streaming applications have recently attracted large number of users in the Internet. In 2010, the number of video streams served increased 38.8 percent to 24.92 billion as compared to 2009 [1]. This huge demand creates a burden on centralized data centers at media content providers such as Video-on-Demand (VoD) providers to sustain the required QoS guarantees [2]. The problem becomes more critical with the increasing demand for higher bit rates required for the growing number of higher-definition video quality desired by consumers. In this paper, we explore new approaches that mitigate the cost of streaming distribution on media content providers using cloud computing.

A media content provider needs to equip its data-center with over-provisioned (excessive) amount of resources in order to meet the strict QoS requirements of streaming traffic. Since it is possible to anticipate the size of usage peaks for streaming capacity in a daily, weekly, monthly, and yearly basis, a media content provider can make long term investments in infrastructure (e.g., bandwidth and computing capacities) to target the expected usage peak. However, this causes economic inefficiency problems in view of flash-crowd events. Since data-centers of a media content provider are equipped with resources that target the peak expected demand, most servers in a typical data-center of a media content provider are only used at about 30 percent of their capacity [3]. Hence, a huge amount of capacity at the servers will be idle most of the time, which is highly wasteful and inefficient.

Cloud computing creates the possibility for media content providers to convert the upfront infrastructure investment to operating expenses charged by cloud providers (e.g., Netflix moved its streaming servers to Amazon Web Services (AWS) [4], [5]). Instead of buying over-provisioned servers and building private data-centres, media content providers can use computing and bandwidth resources of cloud service providers. Hence, a media content provider can be viewed as a re-seller of cloud resources, where it pays the cloud service provider for the streaming resources (bandwidth) served from the cloud directly to clients of the media content provider. This paradigm reduces the expenses of media content providers in terms of purchase and maintenance of over-provisioned resources at their data-centres.

In the cloud, the amount of allocated resources can be changed adaptively at a fine granularity, which is commonly referred to as auto-scaling. The auto-scaling ability of the cloud enhances resource utilization by matching the supply with the demand. So far, CPU and memory are the common resources offered by the cloud providers (e.g., Amazon EC2 [6]). However, recently, streaming resources (bandwidth) have become a feature offered by many cloud providers to users with intensive bandwidth demand (e.g., Amazon CloudFront and Octoshape) [5], [7], [8], [9].

- *A. Alasaad and H.M. Behairy are with the National Center for Electronics, Communications, and Photonics, King Abdulaziz City for Science and Technology, Riyadh, Saudi Arabia. E-mail: {alasaad, hbehairy}@kacst.edu.sa.*
- *K. Shafiee and V.C.M. Leung are with the Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, BC, Canada. E-mail : {kshafiee, vleung}@ece.ubc.ca.*

Manuscript received 7 Nov. 2013; revised 23 Jan. 2014; accepted 24 Mar. 2014. Date of publication 10 Apr. 2014; date of current version 6 Mar. 2015. Recommended for acceptance by H. Wu.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPDS.2014.2316827

The delay sensitive nature of media streaming traffic poses unique challenges due to the need for guaranteed throughput (i.e., download rate no smaller than the video playback rate) in order to enable users to smoothly watch video content on-line. Hence, the media content provider needs to allocate streaming resources in the cloud such that the demand for streaming capacity can be sustained at any instant of time.

The common type of resource provisioning plan that is offered by cloud providers is referred to as on-demand plan. This plan allows the media content provider to purchase resources upon needed. The pricing model that cloud providers employ for the on-demand plan is the pay-per-use. Another type of streaming resource provisioning plans that is offered by many cloud providers is based on resource reservation. With the reservation plan, the media content provider allocates (reserves) resources in advance and pricing is charged before the resources are utilized (upon receiving the request by the cloud provider, i.e., prepaid resources). The reserved streaming resources are basically the bandwidth (streaming data-rate) at which the cloud provider guarantees to deliver to clients of the media content provider (content viewers) according to the required QoS.

In general, the prices (tariffs) of the reservation plan are cheaper than those of the on-demand plan (i.e., time discount rates are only offered to the reserved (prepaid) resources). We consider a pricing model for resource reservation in the cloud that is based on non-linear time-discount tariffs. In such a pricing scheme, the cloud service provider offers higher discount rates to the resources reserved in the cloud for longer times. Such a pricing scheme enables a cloud service provider to better utilize its abundantly available resources because it encourages consumers to reserve resources in the cloud for longer times. This pricing scheme is currently being used by many cloud providers [10]. See for example the pricing of virtual machines (VM) in the reservation phase defined by Amazon EC2 in February 2010. In this case, an open problem is to decide on both the optimum amount of resources reserved in the cloud (i.e., the prepaid allocated resources), and the optimum period of time during which those resources are reserved such that the monetary cost on the media content provider is minimized. In order for a media content provider to address this problem, prediction of future demand for streaming capacity is required to help with the resource reservation planning. Many methods have been proposed in prior works to predict the demand for streaming capacity [11], [12], [13], [14].

Our main contribution in this paper is a practical—easy to implement—Prediction-Based Resource Allocation algorithm (PBRA) that minimizes the monetary cost of resource reservation in the cloud by maximally exploiting discounted rates offered in the tariffs, while ensuring that sufficient resources are reserved in the cloud with some level of confidence in probabilistic sense. We first describe the system model. We formulate the problem based on the prediction of future demand for streaming capacity (Section 3). We then describe the design of our proposed algorithm for solving the problem (Section 4).

The results of our numerical evaluations and simulations show that the proposed algorithms significantly reduce the monetary cost of resource allocations in the cloud as compared to other conventional schemes.

2 Related Work

The prediction of CPU utilization and user access demand for web-based applications has been extensively studied in the literature. A prediction method has been proposed with respect to upcoming CPU utilization pattern demands based on neural networking and linear regression that is of interest in e-commerce applications [15]. Y. Lee et al. proposed a prediction method based on radial basis function (RBF) networks to predict the user access demand request for web type of services in web-based applications [16].

Although the demand prediction for CPU utilization and web applications has been studied for a relatively long period of time, the prediction of demand for media streaming has gained popularity more recently [11], [12], [13], [14]. The access behaviour of users in peer-to-peer (P2P) streaming with time-series analysis techniques using non-stationary time-series models was predicted in [11]. The method of time-series prediction based on wavelet analysis was studied in [12]. In [13], principal component analysis is employed by the authors to extract the access pattern of streaming users. Although most of the above studies predict the average streaming capacity demands, few papers have also studied the volatility of the capacity demand, i.e., the demand variance at any future point in time, which yields more accurate risk factors [14]. The prediction of streaming bandwidth demand is outside the scope of this paper. In this work, we formulate the problem considering a given probability distribution function of prediction of future demand for streaming bandwidth. In addition to demand prediction for resource reservation, other relevant studies have addressed the appropriate joint reservation of bandwidth resources on multiple cloud service providers with the purpose of maximizing bandwidth utilization [12], [14]. In [17], an adaptive resource provisioning scheme is presented that optimizes the bandwidth utilization while satisfying the required levels of QoS. Maximization of bandwidth utilization in turn helps cloud service providers reduce their expenses and maximize their revenues. In [18], an optimization framework for making dynamic resource allocation decisions under risky and uncertain operating environments was developed to maximize revenue while reducing operating costs. This framework considered multiple client QoS classes under uncertainty of workloads.

Recently, streaming resources (e.g., bandwidth) have become a feature offered by many cloud providers to content providers with intensive bandwidth demand. The streaming of media content to content viewers located at different geographical regions at guaranteed data-rate is a part of the service offered by the cloud provider. The common way of implementing this service in the cloud is by having multiple data-centres inside the networks of the access connection providers (e.g., Internet Service Providers, ISPs) located at appropriate geographical locations (Fig. 1) [5], [19], [20]. Cloud service providers may need to negotiate contracts with a number of ISPs to co-locate their servers into the networks of those ISPs. In this regard, another group of papers have focused on studying different

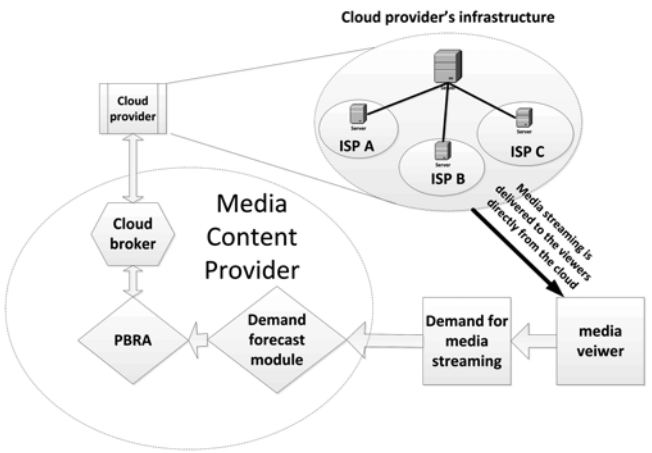

Fig. 1. System model.

types of contracts between cloud service providers and ISPs with the purpose of minimizing the expenses of cloud providers [21]. However, an interesting design approach is to look at the resource reservation problem from the viewpoint of content providers. Obviously, content providers are more interested in minimizing their costs, i.e., the amount of money that they are charged directly by cloud providers.

To the best of our knowledge, very few studies have investigated the problem of optimizing resource reservation with the objective of minimizing the monetary costs for content providers. A good example is presented in [22], wherein a resource reservation optimization problem was formulated to minimize the costs of content providers, so-called cloud consumers, using a stochastic programming model. In the process of problem formulation, uncertain demand and uncertain cloud providers' resource prices are considered. In contrast, the optimization problem formulated in our work takes into account a given probability distribution function obtained from aforementioned studies for the prediction of media streaming demands. Furthermore, the problem of cost minimization is addressed by utilizing the discounted rates offered in the non-linear tariffs. To the best of our knowledge, none of the previous papers has investigated the problem of cost minimization for media content providers in terms of monetary expenses by taking into account both the penalties caused by the over-provisioned or under-provisioned reserved resources, and the advance purchase of resources at cloud providers for just the right period of time.

3 System Model and Problem Formulation

The system model that we advocate in this paper for media streaming using cloud computing consists of the following components (Fig. 1).

- Demand forecasting module, which predicts the demand of streaming capacity for every video channel during future period of time.
- Cloud broker, which is responsible on behalf of the media content provider for both allocating the appropriate amount of resources in the cloud, and reserving the time over which the required resources are allocated. Given the demand prediction, the broker implements our proposed algorithm to make decision on resource allocations in the cloud.
  Both the demand forecasting module and the cloud broker are located in the media content provider site.
- Cloud provider, which provides the streaming resources and delivers streaming traffic directly to media viewers.

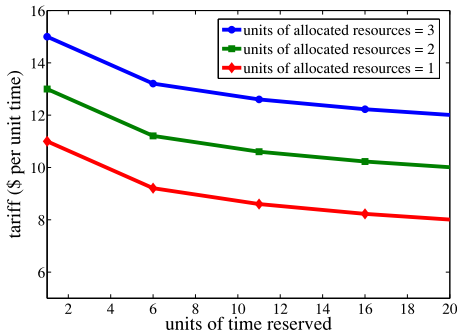

Fig. 2. An example of tariffs as function of allocated resources and reservation time.

In this paper, we consider the case, wherein the cloud provider charges media content providers for the reserved resources according to the period of time during which the resources are reserved in the cloud. In this case, the cloud provider offers higher discount rates to the resources reserved in the cloud for longer times.

Non-linear time-discount is a very popular pricing model. Non-linear tariffs are those with marginal rates varying with quantity purchased and time rented. Time discount rates are available in purchasing most types of goods. Products or services with time usage (e.g., rental cars, rental real-estates, loans, long distance telephone cards, photocopiers) are typically offered with variety of plans (pricing schemes) depending on the period of time the product is consumed (reserved). It has been shown that such pricing schemes enable sellers to increase their revenues [23]. Many cloud providers also use such a pricing scheme [10]. See for example pricing of virtual machines in reservation phase defined by Amazon EC2 in February 2010. An example of tariffs using such a pricing scheme is shown in Fig. 2. We can see that the tariff is a function of both units of allocated resources and reservation time.

We observe the following dilemma: how can the media content provider reserve sufficient resources in the cloud—based on the prediction of future streaming demand—such that no resource wastage is incurred, while QoS for the actual (real) streaming traffic is maintained with some level of confidence (η) in probabilistic sense? Moreover, how can the media content provider utilize the non-linear tariffs (time discount rates offered to the reserved (prepaid) resources) to minimize its monetary cost?

Consider a video channel offered by a media content provider. Let $D(t)$ be the actual demand for streaming capacity of the video channel at an instant of time t, and measured as the number of users that stream the channel at instant of time t multiplied by the data rate required for every downloading user to meet QoS guarantees. It has been shown that $D(t)$ is a random process that follows a log-normal

distribution with mean $E[D(t)]$ and variance (σ) characterized in [11] and [14], respectively.

We denote the amount of streaming bandwidth that the media content provider allocates in the cloud at any time instant t by $Alloc(t)$. Since $D(t)$ is a random process, the media content provider needs to maintain reserved resources in the cloud $Alloc(t)$ such that in any instant of time,

$$Probability(D(t) \leq Alloc(t)) \geq \eta, \quad (1)$$

where η is a pre-determined threshold (level of confidence). Note that a higher η means a higher degree of confidence, in a probabilistic sense, that the reserved resources in the cloud $Alloc(t)$ meet the QoS guarantees for the actual streaming traffic at any future time instant t. However, increasing η increases the probability of wastage of reserved bandwidth (i.e., over-subscribed cost). Hence, proper selection of η is necessary. We shall propose an algorithm that determines the best value of η in Section 5. In this section, our objective is to find the right amount of reserved resources and their corresponding reservation time such that the monetary cost required for streaming a video content (channel) is minimized given the constraint in Eq. (1).

4 Algorithm Design

We summarize the assumptions that we use in our analysis as follows:

1) We assume that upon receiving the resource allocation request by the cloud provider from the media content provider, the resources required are immediately allocated in the cloud, i.e., updating the cloud configuration and launching instances in cloud data-centres incurs no delay.
2) Since the only resource that we consider in this work is bandwidth, it would be important to delve into the relation between the cloud provider and content delivery networks (CDN). However, we assume that the provisioning of media content to media viewers (clients of the media content provider) located at different geographical regions at guaranteed data-rate is a part of the service offered by the cloud provider. The common way of implementing this service in the cloud is by having multiple data-centres inside the networks of the access connection providers (e.g., ISPs) located at appropriate geographical locations (Fig. 1) [5], [19], [20].
3) We assume that the media content provider is charged for the reserved resources in the cloud upon making the request for resource reservation (i.e., prepaid resources); and therefore, the media content provider cannot revoke, cancel, or change a request for resource reservation previously submitted to the cloud.
4) In clouds, tariffs (prices of different amount of reserved resources in \$ per unit of reservation time) are often given in a tabular form. Therefore, the cloud service provider requires a minimum reservation time for any allocated resources, and only allows discrete levels (categories) of the amount of allocated resources in the cloud. See for example the reservation phase in the Amazon CloudFront resource provisioning plans [7].

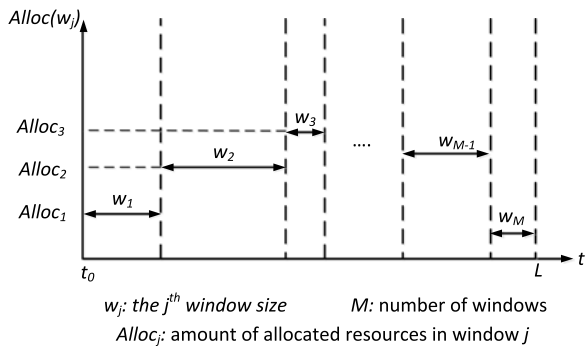

Fig. 3. PBRA algorithm design.

We take into account the aforementioned constraints and propose a practical—easy to implement—algorithm for resource reservation in the cloud, such that the financial cost on the media content provider is minimized.

Suppose that the media content provider can predict the demand for streaming capacity of a video channel (i.e., the statistical expected value of the demand $E[D(t)]$ is known) over a future period of time L using one of the methods in [11], [12], [13], [14]. The content provider reserves resources in the cloud according to the predicted demand. The proposed algorithm is based on time-slots with varied durations (sizes). In every time-slot, the media content provider makes a decision to reserve amount of resources in the cloud. Both the amount of resources to be reserved and the period of time over which the reservation is made (duration of time-slots) vary from one time-slot to another, and are determined in our algorithm to yield the minimum overall monetary cost (Fig. 3).

We alternatively call a time-slot a *window*, and denote the window size (duration of the time-slot) by w. Since the actual demand varies during a window size, while allocated resources in the cloud remain the same for the entire window size (according to the third assumption above), the algorithm needs to reserve resources in every window j that are sufficient to handle the maximum predicted demand for streaming capacity during that window with some probabilistic level of confidence η.

We denote the amount of reserved resources in window j by $Alloc_j$. Since the decision on the amount of reserved resources is affected by the wrong prediction of future streaming demand, our on-line algorithm is carefully designed to obtain accurate demand prediction (by enabling a mechanism that continuously updates the demand forecast module according to the actual demand received at the media content provider over time) in order to reduce the risk of making wrong resource reservation decisions (Fig. 1).

We denote the monetary cost of the reserved resources during window j by $Cost(w_j, Alloc_j)$, and can be computed as

$$Cost(w_j, Alloc_j) = \text{tariff}(w_j, Alloc_j) \times w_j, \quad (2)$$

where $\text{tariff}(w_j, Alloc_j)$ represents the price (in \$ per time unit) charged by the cloud provider for amount of resources

$Alloc_j$ reserved for period of time (window size) w_j. Note that the values of tariff and $Cost$ in any window j depend on both the amount of allocated resources ($Alloc_j$) and the period of time over which resources are reserved (w_j). Also note that the algorithm runs on-the-fly. More specifically, the demand forecast module predicts streaming capacity demand in the upcoming period of time L and feeds this information to our algorithm. The algorithm upon receiving the demand prediction, computes the right size of window j (i.e., w_j^*), and the right amount of reserved resources in window j (i.e., $Alloc_j^*$), such that the cost of the reserved resources during window j (i.e., $Cost(w_j, Alloc_j)$ in (2)) is minimized; or equivalently, the discounted rates offered in the tariffs are maximally utilized.

Hence, the objective of our algorithm is to minimize $Cost(w_j, Alloc_j)\ \forall j$, subject to

$$Probability(D(t) \leq Alloc(t)) \geq \eta, \quad \forall\, t \in L.$$

In other words, our objective is to minimize the monetary cost of reserved resources such that the amount of reserved resources at any instant of time is guaranteed to meet the actual demand with probabilistic confidence equals to η. As we have discussed earlier, $D(t)$ is a random process that follows a log-normal distribution with mean $E[D(t)]$ and variance (σ) characterized in [11] and [14], respectively. Thus, using the constraint above, and for any window size w_j, we can compute the minimum amount of required reserved resources during window j ($Alloc_j$) by solving the following formula for $Alloc_j$

$$\int_0^{Alloc_j} \frac{1}{x \cdot \sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{ln(x)-\mu_{max}}{\sigma})^2} dx = \eta, \tag{3}$$

where μ_{max} is the maximum value of the predicted streaming demand during the window j (i.e., $\mu_{max} = argmax(E[D(t)])\ \forall\, t \in w_j$). Note that the Equation (3) follows from the log-normal probabilistic distribution of the demand for streaming capacity.

As we have discussed earlier, the cloud service provider often requires a minimum reservation time for any allocated resources (w_{min}), and only allows discrete levels (categories) of reservation times for any amount of allocated resources in the cloud. We therefore, assume that any reservation time required at the cloud has to be in multiplicative order of w_{min} (i.e., $w_j = k \cdot w_{min}$, where k is a positive integer). Thus, the algorithm employs a trial window (w_h) to assist in making optimum decision on the size of every window j. In particular, for every window j, the algorithm starts an iteration process with a trial window of size $w_h = w_{min}$, and computes the cost rate ($X_h = \text{tariff}(w_h, Alloc_h)$, where h is iteration index), and $Alloc_h$ is computed by solving Eq. (3) for $Alloc$.

Recall that due to the time discount rates offered in the tariffs, increasing the time during which the allocated resources are reserved may lead to less monetary cost (higher discounted rate) on the media content provider (Fig. 2). However, increasing the window size (time-slot) significantly may also result in high over-provisioning (over-subscribed) cost as the media content provider has to allocate resources in the cloud that meet the highest demand during the window period. Thus, in order to recognize whether the cost is decreasing or increasing with increasing the window size, the trial window size (w_h) is increased one w_{min} unit in every iteration (i.e., $w_h = w_h + w_{min}$) and the cost rate of this new trial window size is computed (X_{h+1}). The algorithm keeps increasing the trial window size until $w_h = L$ in order to scan the entire period of time over which the demand was predicted (L) (Fig. 3), and finds the value of w_h that yields the minimum cost; that is the optimum size of window j (w_j^*). Since L is the period of time over which the future demand is predicted, then $w_{min} \leq w_j^* \leq L$.

During every window, the media content provider receives the real (actual) streaming demand for the video channel, which may be different from the predicted demand. According to the actual demand, the demand forecast module updates its prediction and feeds the algorithm with a newly predicted demand for another future period of time L (Fig. 1). The algorithm upon receiving the updated demand prediction, computes the optimum size of the next window, and reserves optimum resources in the next window, and so on. The pseudo code for the proposed algorithm is shown in Algorithm 1. In order to further clarify operations of the proposed algorithm (which we call it Prediction-Based Resource Allocation algorithm), an example is given in the following.

Algorithm 1 Pseudo code for determining optimum window sizes and optimum resource allocations in every window.

Given the predicted demand ($E[D(t)]$) over a future period of time L,
Define:
w_h as a trial window size that the algorithm uses to make decision on the optimum size of window j,
w_{min} as the minimum reservation time that is required by the cloud provider for any amount of resources reserved in the cloud,
j refers to the j-th window,
To compute w **and** $Alloc$ **for every window** j**, do**
$w_h \leftarrow 0$, {initial value}
$h \leftarrow 1$, {start iterations}
while $w_h \leq L$, **do**
  $w_h = w_h + w_{min}$, {increment the trial window}
  Compute μ_{max_h},
  Compute $Alloc_h$ by solving Eq. (3) for $Alloc$,
  $X_h = \text{tariff}(w_h, Alloc_h)$,
  $h \leftarrow h + 1$,
end while
$X_F = argmin(X_h\ \ \forall\, h)$, {out of all X_h values, find the one with least value}
Find h^* corresponding to X_F, {pick the value of h that yields the least X_F}
$w_j^* \leftarrow w_{h^*}$
$Alloc_j^* \leftarrow Alloc_{h^*}$
$j \leftarrow j + 1$,

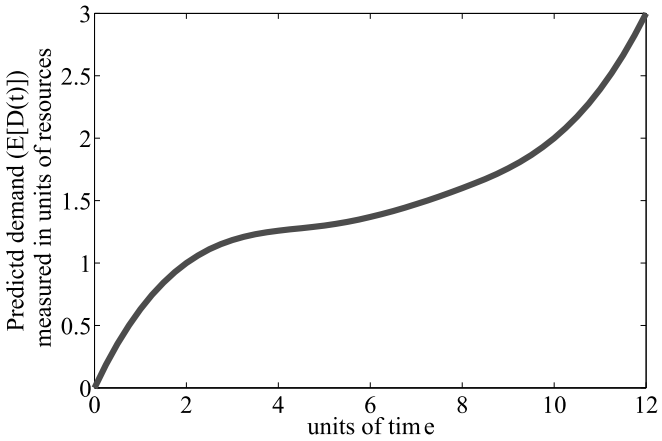

Fig. 4. An example of predicted demand over a period of future time $L = 12$.

4.1 Example: Finding the Right Amount of Reserved Resources in Window j and Their Reservation time

Consider the normalized predicted streaming demand given in Fig. 4 for a future period of time $L = 12$. Let $w_{min} = 1$; and let $\eta = 0.75$. Assume that the amount of reserved resources in the cloud can only take integer numbers of unit of resources (i.e., cloud provider applies certain levels (categories) on the amount of allowed reserved resources, $Alloc(t) \in \{1, 2, 3, \ldots\}$.

For the given predicted demand, our algorithm finds the optimum size of every window j and optimum amount of reserved resources in window j as follows. The algorithm starts iterations to determine the size of the first window (i.e., $w_{j=1}$). In the first iteration ($h = 1$), $w_{h=1} = 1$, we can see that the maximum predicted demand when $w_{h=1} = 1$ is 0.63 (Fig. 4). Thus, we have $\mu_{max_h} = 0.63$. Using Eq. (3), we have $Alloc_{h=1} = 0.81$. Since the cloud allows only discrete levels for reserved resources in the cloud, then $Alloc_{h=1}$ must be rounded to the nearest upper value allowed in the cloud. Thus, $Alloc_{h=1} = 1$. Using tariff functions shown in Fig. 2, we have the cost rate $X_h = \text{tariff}(w_{h=1} = 1, Alloc_{h=1} = 1) = 11$. The iterations continue until $w_h = L$.

We summarize the results of all iterations h performed for window $j = 1$ using our proposed algorithm in Table 1. From the table, we can see that the minimum value of cost rate X_h is when $h^* = 10$. Hence, the optimum window size is $w^*_{j=1} = w_{h=10} = 10$, and the optimum amount of reserved resources during window $j = 1$ is $Alloc^*_{j=1} = Alloc_{h=10} = 2$. Similarly, we can find the optimum window size and optimum amount of resources in the next window ($j = 2$) given an updated prediction of the demand in another period of future time L.

5 Hybrid Approach for Resource Provisioning

In this section, we consider the case, wherein the cloud provider offers two different types of streaming resource provisioning plans: the reservation plan and the on-demand plan. With the reservation plan, the media content provider reserves resources in advance and pricing is charged before the resources are utilized (upon receiving the request at the cloud provider, i.e., prepaid resources). With the on-demand plan, the media content provider allocates streaming resources upon needed. Pricing in the on-demand plan is charged by pay-per-use basis. In general, the prices (tariffs) of the reservation plan are cheaper than those of the on-demand plan (i.e., time discount rates are only offered to the reserved (prepaid) resources). Amazon CloudFront [7], Amazon EC2 [6], GoGrid [24], MS Azure, Op-Source, and Terre-mark are examples of cloud providers which offer Infrastructure-as-a-Service (IaaS) with both plans [10].

When the media content provider only uses the resource reservation plan, the *under-provisioning* problem can occur if the reserved (prepaid) resources are unable to fully meet the actual demand due to high fluctuating demand or prediction mismatch. Also, *over-provisioning* problem can occur if the reserved (prepaid) resources are more than the actual demand, in which parts of the reserved resources are wasted. However, when the cloud provider offers both the reservation plan and the on-demand plan, the media content provider can allocate resources in the cloud more efficiently. In particular, the media content provider can use reservation plan to benefit from the time-discounted rate, while use the on-demand plan to dynamically allocate streaming resources to its clients at the moment when the reserved resources allocated using the reservation plan are unable to meet the actual demand and extra resources are needed to fit the fluctuated and unpredictable demands (e.g., flash crowd). We call this approach *hybrid resource provisioning*. This hybrid approach eliminates both the over-provisioning (over-subscribed) cost and the under-provisioning problem that may occur when using the reservation plan only.

In this hybrid resource provisioning approach, tradeoff between the amount of resources allocated using the on-demand plan and the amount of resources allocated using the reservation plan needs to be adjusted in which the hybrid approach can optimally perform. In this section, we propose an algorithm for this hybrid resource provisioning approach that maximally benefits from the time discounted rate offered in the resource reservation plan, while eliminating any over-provisioning cost of reserved resources such that the overall monetary cost of resource allocations in the cloud (including both the reserved resources and the on-demand resources) is minimized.

TABLE 1
Example: Summary of Results for Iterations Executed for Window $j = 1$

iteration (h)	1	2	3	4	5	6	7	8	9	10	11	12
w_h	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0	11.0	12.0
μ_{max}	0.63	1.0	1.184	1.26	1.3	1.37	1.47	1.60	1.76	2.0	2.4	2.7
$Alloc_h$	1.0	1.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0	2.0	3.0	3.0
$X_h = \text{tariff}(w_h, Alloc_h)$	11	10.85	12.25	12.0	11.75	11.50	11.28	11.06	10.84	10.62	12.65	12.43

As we have described in the previous section (Section 4), the cost of allocated resources using the reservation plan depends on the parameter η. We referred to η as the level of confidence. We have shown that using higher value of η results in higher amount of reserved resources in the cloud, and vice-versa. However, increasing the value of η for the reserved resources may lead to the *over-provisioning* problem, while decreasing the value of η may lead to the *under-provisioning* problem. Since pricing of resource allocation in the on-demand plan is higher than the reservation plan, one may erroneously believe that increasing the value of η would always reduce the overall monetary cost since the portion of reserved (discounted) resources in the cloud is increased. However, reserving too many resources (i.e., using high value of η for the reserved (prepaid) resources) may be far from optimal because it may significantly increase the *over-provisioning* (over-subscribed) cost. Hence, this hybrid approach requires that the content provider select the right value of η for the reserved resources. Our proposed algorithm in this section computes the optimum value of η (η^*) that yields the minimum overall monetary cost of resource allocations in the cloud (both reserved and on-demand resources) when the media content provider uses this hybrid resource provisioning approach.

Let us again assume that the media content provider can predict the demand for a future period of time L. Let C_{hybrid} be the price that the media content provider expects to pay to the cloud provider for all streaming resource allocated in the cloud using the hybrid approach (i.e., C_{hybrid} is the statistical mean of the cost). We can see that C_{hybrid} is the summation of two terms: the price charged for the reserved resources in every window j using the reservation plan (denoted by C_{RSV_j}), and the expected cost of resources allocated in the cloud during every window j using the on-demand plan (denoted by C_{OD_j}). Hence,

$$C_{hybrid} = \sum_j (C_{RSV_j} + C_{OD_j}). \tag{4}$$

Let $Alloc_{RSV_j}$ be the amount of reserved resources in window j, while $Alloc_{OD_j}$ be the amount of on-demand resources allocated in window j. Let $\text{tariff}(w_{RSV_j}, Alloc_{RSV_j})$ be the tariff charged for the reserved resources in window j, while $\text{tariff}(Alloc_{OD_j})$ be the tariff charged for the on-demand resources in window j. Note that the cost rate of the resources reservation plan, $\text{tariff}(w_{RSV_j}, Alloc_{RSV_j})$, depends on both w_{RSV_j} and $Alloc_{RSV_j}$; while $\text{tariff}(Alloc_{OD_j})$ depends only on the amount of allocated resources $Alloc_{OD_j}$. This is because no time discount rate is offered to the on-demand resources.

Let x be a random variable representing the demand for streaming capacity in any instant of time during window j, and $f(x)$ be the probability density function of variable x. Note that when the amount of reserved resources in window j ($Alloc_{RSV_j}$) is known, C_{OD_j} can be computed by considering the event when $Alloc_{RSV_j} < x < \infty$. This is because when $x < Alloc_{RSV_j}$, the amount of reserved resources in the cloud is sufficient to handle the actual streaming demand and no need to allocate extra resources using the on-demand plan. Thus, we can compute the cost of reserved resources in window j (in \$) as

$$C_{RSV_j} = w_j \cdot \text{tariff}(w_{RSV_j}, Alloc_{RSV_j}), \tag{5}$$

and consequently the expected (statistical mean) cost of the on-demand resources in window j can be computed as

$$C_{OD_j} = w_j \cdot \int_{Alloc_{RSV_j}}^{\infty} f(x) \cdot \text{tariff}(x - Alloc_{RSV_j})\, dx. \tag{6}$$

We shall consider a log-normal statistical probability distribution $f(x)$ as discussed earlier [11], [14]. Thus, $f(x)$ in Eq. (6) can be written as

$$f(x) = \frac{1}{x \cdot \sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{ln(x)-\mu_{max}}{\sigma})^2}.$$

As we have described in Section 4, the right amount of reserved resources in window j ($Alloc_{RSV_j}$) can be determined given the parameter η. Thus, C_{hybrid} in Eq. (4) is a function of the parameter η only. Our objective is to minimize C_{hybrid} in Eq. (4), or equivalently determining the value of η that minimizes the overall cost of allocated resources using the hybrid approach. It is straight forward to show that C_{hybrid} is convex with respect to η. Thus, in order to minimize C_{hybrid}, we need to find the optimum value of η (η^*) using Equations (5) and (6).

We can see that η^* can be easily solved numerically for every window j if tariff functions are given (i.e., $\text{tariff}(t, Alloc_{RSV}(t))$ and $\text{tariff}(Alloc_{OD}(t))$ for any duration of resource allocation). However, as we have discussed earlier, tariffs are often given in a tabular form. Moreover, the cloud service provider often requires a minimum reservation time for any allocated resources, and only allows discrete levels (categories) of allocated resources in the cloud. We take into account those constraints and propose an efficient heuristic algorithm for this hybrid resource provisioning approach. The pseudo code of the proposed algorithms is shown in Algorithm 2.

The algorithm works as follows. Suppose that η takes discrete values, and the total possible values of η is S. For every window j, the iteration process described in Algorithm 1 is performed for every value of η in order to compute both the right amount of reserved resources ($Alloc_{RSV_j}$) and the right time over which these resources are reserved (w_{RSV_j}). When the amount of reserved resources in window j is determined, the amount of extra resources that must be allocated using the on-demand plan in order to fulfil the predicted streaming demand can be easily computed as $Alloc_{OD_j} = \mu_{max} - Alloc_{RSV_j}$, where μ_{max} is the maximum value of the predicted streaming demand during window j. Thus, the total corresponding cost rate of allocated resources in window j is computed as $X_h = \text{tariff}(RSV_j,\ Alloc_{RSV_j}) + \text{tariff}(Alloc_{OD_j})$, where h is the iteration index. The iteration process continues, and out of all values of X_h computed for different values of η, the algorithm finds η^* corresponding to the minimum value. The algorithm is repeated for every window.

We can see that the complexity of the proposed algorithm (measured in terms of number of iterations required for every window) is $O(\frac{L}{w_{min}} \cdot S)$. Thus, increasing the size of

S increases the complexity of the algorithm, but also increases the accuracy of the algorithm. However, the complexity of our algorithm linearly scales with size of the input (S), which means that our algorithm executes efficiently.

Algorithm 2 Pseudo code for determining optimum resource allocations in the cloud using two resource provisioning plans.

Define:
S as the set of all values of η that the algorithm needs to test in order to determine the best amount of allocated resources that minimizes C_{hybrid},
For every window j, **do**
for every value η in the set S, **do**
$h \leftarrow 1$, {start iterations}
Run Algorithm 1 to find the best size of window j (w_j^*) and the best amount of resource allocation ($Alloc^*_{RSV_j}$) for this particular value of η using the reservation plan,
Compute $Alloc_{OD_j} = \mu_{max} - Alloc_{RSV_j}$, where μ_{max} is the maximum value of the predicted streaming demand during window j,
Compute $X_h = \text{tariff}(RSV_j, Alloc_{RSV_j}) + \text{tariff}(Alloc_{OD_j})$,
$h \leftarrow h + 1$
end for
$Y_F = argmin(X_h \ \forall h)$, {out of all values of X_h, find the one with the least value}
Find h^* corresponding to Y_F, {pick the value of h that yields the least Y_F}
$\eta^* \leftarrow \eta_{h^*}$, {pick value of η corresponding to h^*}
$Alloc^*_{RSV_j} \leftarrow Alloc^{h^*}_{RSV_j}$,
$Alloc^*_{OD_j} \leftarrow Alloc^{h^*}_{OD_j}$

6 Performance Evaluation

We first analytically derive a demand prediction function that we shall use in our performance evaluations (Section 6.1). We then investigate the performance of our simple "on-line" Prediction-Based Resource Allocation algorithm proposed for reserving resources in the cloud, in terms of both monetary cost of reserved resources in the cloud and complexity (CPU time) (Section 6.2). We then compare the performance of PBRA proposed for reserving resources in the cloud against two other schemes: Fixed window size resource reservation scheme, and pay-as-you-go resource allocation scheme (Section 6.2.2). Finally, we evaluate the performance of our hybrid resource allocation algorithm proposed for the case when the cloud provider offers two streaming resource provisioning plans: the reservation and on-demand, and show that our algorithm significantly reduces the overall cost of resource allocation (Section 6.3).

6.1 Demand Model

As we have discussed so far, prediction of the future demand for streaming capacity is required in order for the media content provider (e.g., VoD) to optimally reserve resources in the cloud. In this section, we use a special case of the demand in which the function of expected (mean) future streaming demand for a video channel (i.e., $E[D(t)]$) can be easily formulated analytically. Specifically, we assume that all media streaming demand for a video channel available at a local VoD provider is generated from users located in a single private network (e.g., users in a college or office campuses).

What distinguishes the evolution of interest in a media content among users of a private network from the Internet is that users in a private network are often socially connected (e.g., friends/colleagues in a social network). Those users form a community and share similar interests. Thus, the demand of a media content grows quickly in the private network as interested users contact others (by either broadcasting the knowledge about existence of the media content to their friends in the social network, e.g., facebook, or using Email-group broadcast) and make them interested. However, the interest (demand) tapers off when a certain cumulative level of interest among users of the private network is reached. For example, a student, in a class of 100 students, can spread the knowledge about a video content to his classmates. If the popularity of this content among students in the class is 0.2, the evolution of the demand increases quickly over time as interested users contact others, but tapers off when all potential number of interested students in the class (20 students) get interested in the content and viewed the content. When all 20 students finish viewing the video content, the life-time of that content in this community network expires.

We analytically characterize this viral evolution of interest in a media content among users of a private network. Let us assume that the number of friends to whom a user is connected in a social network (node's degree) at any instant of time on average is N. Let us further assume that a user who receives the notification about the existence of the content gets interested with probability p and re-broadcasts the notification, in turn, to his friends on the social network, where p is the expected popularity of the content among users of the private network. We further assume that users who receive multiple notifications for the same content do not rebroadcast the message.

If the social network graph is fully connected (i.e., a notification about existence of the content reaches all users in the private network), we can then use the fluid-flow model to write the evolution of interest in a media content as

$$\frac{dI(t)}{dt} = I(t)[p(N - \gamma(t) \times N)],$$

where $I(t)$ be the total number of interested users in the content at time t (cumulative interest). $(\gamma(t) \times N)$ accounts for the fraction of N users who received multiple notifications by time instant t, $\gamma(t) := \frac{I(t)}{N_T}$, where N_T is the potential number of users in the network who will ultimately become interested in the content ($N_T = 100$ in Fig. 5), i.e., N_T be the maximum expected level of the content cumulative interest in the private network.

The above formula is a second order Bernoulli differential equation and can be solved as

$$I(t) = \frac{N_T \times I(0)}{I(0) + (N_T - I(0))e^{-p \times N \times t}}, \quad (7)$$

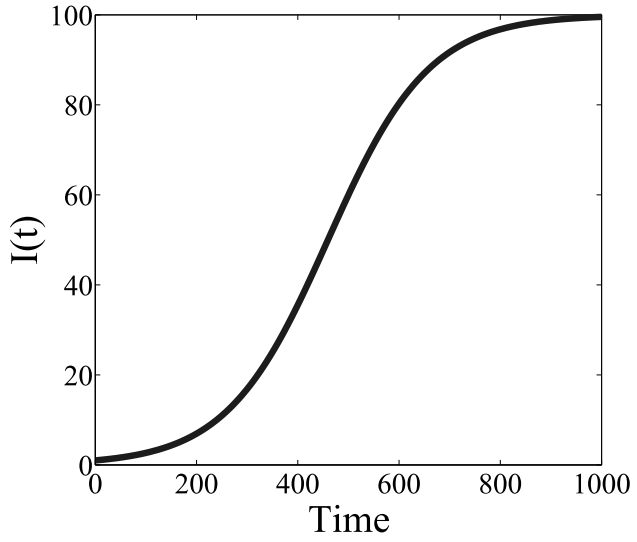

Fig. 5. The evolution of interest in the video channel.

where $I(0)$ be the number of interested users at time $t = 0$. We note that $I(t)$ has an S-shape (Fig. 5). It shows that the number of interested users increases quickly when the content becomes available and then gradually decreases and tapers off once the level of interest reaches N_T. This is similar to the demand function that was obtained using word-of-mouth spread of information by interested users (Bass model). Similar interest evolution was also observed when measuring user interest in a video file on YouTube server [25], and when measuring user interest in popular video hosted on a university infrastructure (CoralCDN) [26].

Given the evolution of interest in a media content $I(t)$ in Eq. (7), we can now use fluid-flow model to write the rate at which downloading users are completely served (finish downloading the media content) as

$$\frac{dS(t)}{dt} = \mu_Q \cdot [I(t) - S(t)],$$

where μ_Q is the required QoS streaming rate for every downloading user (measured in bits/second), and $S(t)$ is the number of completely served users at time instant t. The above differential equation can be easily solved for $S(t)$. Hence, the expected value of demand for stream capacity of the content at any time t (measured in bits/second) is

$$E[D(t)] = \frac{dS(t)}{dt} = \mu_Q \cdot [I(t) - S(t)]. \quad (8)$$

6.2 Evaluation of the Algorithm (PBRA) Proposed for Reserving Resources in the Cloud

The algorithm that we evaluate in this section is the very first algorithm that was proposed in Section 4 for resource reservation in the cloud. We used time-discount rates similar to those used in the pricing model employed by Amazon EC2 [6] in order to derive tariff functions that we used in our evaluations. Those tariffs are non-linear functions of both the amount of reserved resources and reservation time. An example of a tariff function that we used in our evaluations for units of reserved resources equal to 3 is depicted in Fig. 6. Note that time discounts are given to the reserved resources. For example, we can see that if the media content provider wants to reserve (prepaid purchase) 3 units of streaming resources for 6 time units, then the tariff is 13 $ per unit of reserved time; whereas the tariff is 14.25 if the same amount of resources is reserved for only 1 time unit. We consider a log-normal probability distribution of the demand for streaming capacity with mean (i.e., predicted demand $E[D(t)]$) computed by Eq. (8) for $I(t)$ given in Fig. 5, $\mu_Q = 1$, and variance of 3.

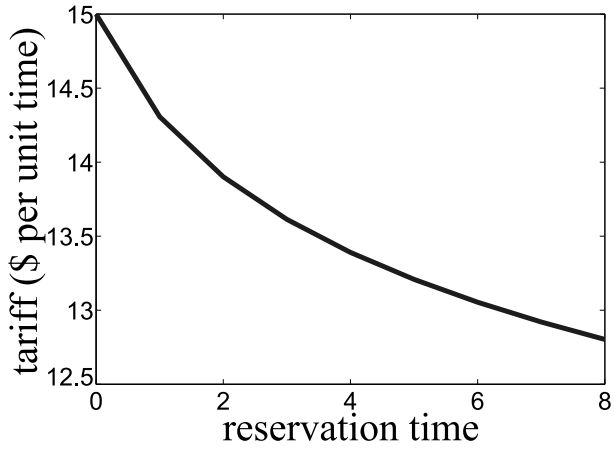

Fig. 6. A tariff function for units of reserved resources equal to 3.

6.2.1 Performance versus Complexity

As we have discussed in Section 4, our proposed algorithm (PBRA) employs a trial window w_{try} with size taking values in multiplicative order of w_{min}, where w_{min} can be defined as the granularity of the resource allocation in the cloud (i.e., it is the minimum reservation time that the cloud provider requires for any amount of resource reserved in the cloud), and it is measured in units of time. To investigate the impact of the value of w_{min} on the performance of our algorithm, we compared the financial cost of media streaming when using our algorithm for varied sizes of w_{min} at $\eta = 0.75$. To plot the comparison figure, we computed the ratio of the overall cost of resource reservation for every value of w_{min} to the overall cost when using $w_{min} = 1$ (i.e., normalized cost) (Fig. 7).

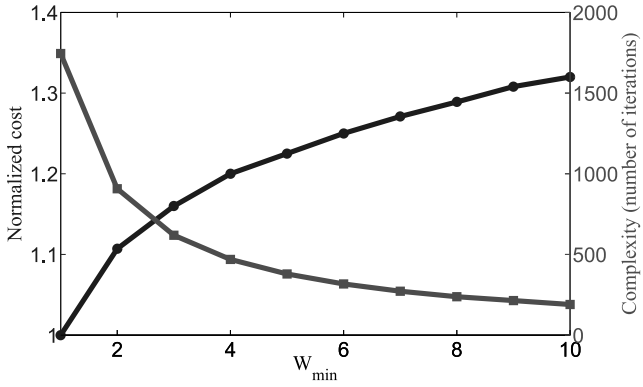

Fig. 7. Performance versus complexity of the PBRA algorithm for resource reservation in the cloud.

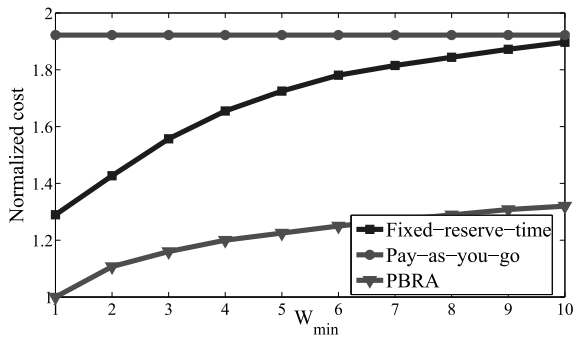

Fig. 8. Performance comparisons.

The results show that the algorithm provides the least cost of resource allocation in the cloud when $w_{min} = 1$. Hence, we can see that the finer granularity that we have in resource allocation in the cloud (i.e., the smaller value of w_{min}), the better performance we get in our algorithm. The better performance, however, comes at higher algorithm complexity, where complexity is measured in terms of total number of iterations (h). We can see that h is higher for smaller w_{min} (Fig. 7). However, even for the highest number of iterations (when $w_{min} = 1$), total CPU time was only 1.02 second using Intel(R) Core(TM)2 Quad CPU @ 2.82 GHz. If we compare this execution time with the period of time over which the algorithm is operating $0 \leq t(sec) \leq 1{,}000$ (Fig. 5), we can see that our algorithm executes very efficiently.

6.2.2 Comparison with Other Resource Provisioning Algorithms

Recall that our proposed algorithm for resource reservation in the cloud (PBRA) is based on windows with variable sizes (i.e., variable time slots as shown in Fig. 3). The size of every window and the amount of reserved resources in every window is determined to minimize the financial cost on the media content provider. We evaluate the performance of our PBRA algorithm against two other resource provisioning schemes: fixed window size scheme (denoted by fixed-reserve-time), and the pay-as-you-go resource allocation scheme which is widely used in the clouds (denoted by pay-as-you-go). The fixed window size scheme is based on resource reservation wherein all time-slots (windows) are of the same size (i.e., w_j is the same $\forall j$). The pay-as-you-go scheme is based on on-demand resource allocation wherein resources are allocated upon needed. The price of reserved resources is less than the on-demand resources since time-discounted rates are only given to the reserved resources.

We computed the overall financial cost when using each of the above schemes for resource allocation in the cloud. To plot the comparison figure, we computed the ratio of the overall cost for every value of w_{min} to the cost when using our PBRA algorithm with $w_{min} = 1$ (Normalized cost) (Fig. 8). In the case of Fixed-reserve-time, we set w_j always fixed as $w_j = w_{min}\ \forall j$, and $w_j = 10$. We can see that PBRA outperforms the Fixed-reserve-time scheme for all values of w_{min}. This is because PBRA selects window sizes according to the predicated demand such that the right amount of resource is reserved in the cloud that maximally benefits from the time-discount rates in the tariffs, and ensures that reserved resources meet the actual demand without incurring wastage. PBRA also outperforms the pay-as-you-go scheme because it maximally benefits from the time-discounted rates given to the reserved resources, while no discount is given to resources allocated using the on-demand scheme.

6.2.3 Impact of Different Probability Distributions of the Demand

In the next set of evaluations, we considered three log-normal probability distribution functions for the demand with same mean but varied variances. The mean of all log-normal distributions $E[D(t)]$ is given in Eq. (8), where $I(t)$ is given in Fig. 5, $\mu_Q = 1$, while variances of the log-normal distributions were set to 3, 6, and 8.

The stochastic effect of demand on the cost of reserved resources using PBRA is shown in Table 2 when $\eta = 0.75$. We observe that the overall resource reservation cost increases as the variance of the log-normal distribution increases. This is because larger variance means higher likelihood that the reserved resources in the cloud do not meet the actual demand. Consequently, higher reserved resources are required in the cloud to meet the actual demand given a certain probabilistic confidence η, which results in higher cost for resource reservation in the cloud.

6.3 Evaluation of the Hybrid Approach for Resource Allocation in the Cloud

In this section, we evaluate the performance of our hybrid resource allocation algorithm proposed in Section 5. Our hybrid approach enables the media content provider to efficiently allocate resources in the cloud using both the reservation resource provisioning plan and the on-demand resource provisioning plan offered by the cloud provider.

As we have discussed in Section 5, the right value of parameter η has to be determined for this hybrid approach to optimally perform. To investigate the impact of different values of η on the performance of the hybrid approach, we considered continuous non-linear tariffs that are functions of both the allocated resources and reservation time. We used time-discount rates similar to those used in the pricing model employed by Amazon EC2 [6] in order to derive tariff functions that we used in our evaluations. Time discount rates are only offered to reserved resources, while no time discount rates are offered to resources allocated using the on-demand plan. An example of a tariff function that we

TABLE 2
Media Streaming Cost Given Different Probability Distributions of the Demand (in $)

Distribution	log-normal ($\sigma = 3$)	log-normal ($\sigma = 6$)	log-normal ($\sigma = 8$)
Cost	$34,457$	$41,543$	$48,393$

TABLE 3
Media Streaming Cost Using Two Resource Allocation Plans Provided by the Cloud (Hybrid Resource Provisioning Approach) (in $)

η	Cost of reservation plan	Cost of on-demand plan	Total cost
0.75	34,457	12,213	46,670
0.8	36,979	8,854	45,833
0.9	44,033	2,821	46,854
0.95	46,324	2,741	49,065

used in our evaluations for units of allocated resources equal 3 is depicted in Fig. 6. Referring to Fig. 6, if the average units of resources allocated in the cloud for 6 time units using the on-demand plan is 3, then the cost is $15 \cdot 6 = \$90$; whereas if the media content provider reserves (prepaid purchase) the same amount of resources for 6 time units using the reservation plan, then the price charged is only $13 \cdot 6 = \$78$.

In the next set of simulations, we consider a demand with mean $E[D(t)]$ given in Eq. (8), where $I(t)$ is given in Fig. 5, $\mu_Q = 1$, and variance of 3. Recall that our hybrid approach selects the right value of η in every window. In every window j, different values of η are tested to selects the one that yields the least overall cost. Table 3 show the cost of resources allocated using both the resource reservation plan and resource on-demand plan when $j = 7$ (corresponding to $t = 650$), which results from using our hybrid algorithm. We observe that when η increases, the cost of the resources allocated using the reservation plan increases, while the cost of resources allocated using the on-demand plan decreases. This is because higher amount of reserved resources is required in the cloud for higher η and, consequently, less amount of on-demand resources is needed. We also observe that when η increases from 0.75 to 0.8 the overall cost (i.e., the cost of both reservation and on-demand resources) decreases; whereas when η increases beyond 0.8 the overall cost increases. This is because the over-subscribed (over-provisioning) cost of the reserved resources becomes very high when $\eta > 0.8$. We can see that the optimum value of η (i.e., the value of η that yields the least overall cost) when $j = 7$ is about 0.8.

To get a sense of how the optimal selection of the value of η can significantly reduce the overall monetary cost on the media content provider when using this hybrid streaming resource provisioning approach, let us compare the total cost when using our hybrid resource allocation algorithm at $j = 7$ against two cases: the case when the media content provider uses the on-demand plan only (*pay-as-you-go*), and the case when the media content provider uses the reservation plan only (*fixed-reserve-time*). We observed that the cost of our hybrid approach when $\eta^* = 0.8$ is $\$45{,}833$; while the cost of allocated resource in the case of *pay-as-you-go* is fixed at about $\$52{,}000$ (does not depend on the value of η), and the cost of allocated resources in the case of *fixed-reserve-time* when $\eta = 0.8$ is about $\$48{,}000$ (Fig. 9). Hence, our algorithm reduces the cost by an amount of about $\$6{,}200$ compared to *pay-as-you-go* (i.e., about 12 percent cost saving), and reduces the cost by an amount of $\$2{,}200$ compared to *fixed-reserve-time* (i.e., 4.5 percent cost saving). We note here that the cost was computed for only one video channel. However, a media content provider generally offers hundreds of video channels to its clients. Therefore, the overall cost-saving using our proposed algorithm can be significantly high for large number of video channels offered by the media content provider.

7 Conclusion and Future Work

This paper studies the problem of resource allocations in the cloud for media streaming applications. We have considered non-linear time-discount tariffs that a cloud provider charges for resources reserved in the cloud. We have proposed algorithms that optimally determine both the amount of reserved resources in the cloud and their reservation time—based on prediction of future demand for streaming capacity—such that the financial cost on the media content provider is minimized. The proposed algorithms exploit the time discounted rates in the tariffs, while ensuring that sufficient resources are reserved in the cloud without incurring wastage. We have evaluated the performance of our algorithms numerically and using simulations. The results show that our algorithms adjust the tradeoff between resources reserved on the cloud and resources allocated on-demand. In future work, we shall perform experimental measurements to characterize the streaming demand in the Internet and develop our own demand forecasting module. We shall also investigate the case of multiple cloud providers and consider the market competition when allocating resources in the clouds.

Acknowledgments

This work was supported by the National Center of Electronics, Communication, and Photonics at King Abdulaziz City for Science and Technology (Saudi Arabia). This paper was based in part on a paper appeared in the proceeding of the IEEE Globecom 2012.

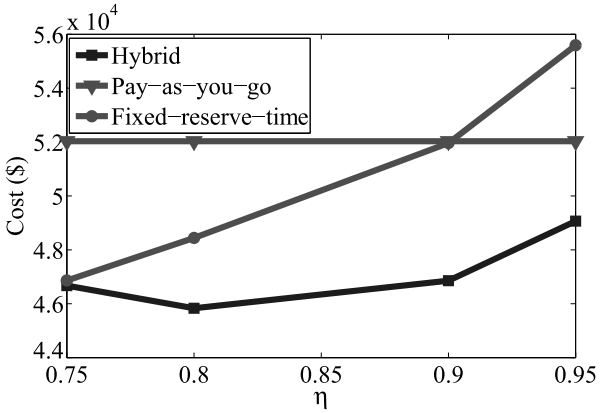

Fig. 9. Hybrid approach performance comparisons.

REFERENCES

[1] Cisco Systems Inc., San Jose, CA, USA, "Cisco visual networking index: Forecast and methodology, 2010–2015," *White Paper*, 2010.

[2] Y. Liu, Y. Guo, and C. Liang, "A survey on peer-to-peer video streaming systems," *Peer-to-Peer Netw. Appl.*, vol. 18, no. 1, pp. 18–28, 2008.

[3] Cisco Systems Inc., San Jose, CA, USA, "Data center virtualization and orchestration: Business and financial justification," *White Paper*, 2007.

[4] Four Reasons We Choose Amazons Cloud as Our Computing Platform, *The Netflix Tech. Blog*, Dec., 2010.

[5] (2014). [Online]. Available: http://www.octoshape.com/

[6] Amazon EC2 Reserved Instances, (2012.) [Online]. Available: http://aws.amazon.com/ec2/reserved-instances

[7] Amazon CloudFront, (2012.) [Online]. Available: http://aws.amazon.com/cloudfront/

[8] G. Chuanxiong, G. Lu, H. Wang, S. Yang, C. Kong, P. Sun, W. Wu, and Y. Zhang, "SecondNet: A data center network virtualization architecture with bandwidth guarantees," in *Proc. ACM 6th Int. Conf. Emerg. Netw. Exp. Technol.*, 2010, pp. 15:1–15:12.

[9] H. Ballani, P. Costa, T. Karagiannis, and A. Rowstron, "Towards predictable datacenter networks," in *Proc. ACM SIGCOMM Conf.*, 2011, pp. 242–253.

[10] E. White, M. O'Gara, P. Romanski, P. Whitney. (2012). Cloud pricing models, *Cloud Expo: Article*, white paper, [Online]. Available: http://java.sys-con.com/node/2409759?page=0,1

[11] D. Niu, Z. Liu, B. Li, and S. Zhao, "Demand forecast and performance prediction in peer-assisted on-demand streaming systems," in *Proc. IEEE Conf. Comput. Commun.*, 2011, pp. 421–425.

[12] S. Peichang, W. Huaimin, Y. Gang, L. Fengshun, and W. Tianzuo, "Prediction-based federated management of multi-scale resources in cloud," *Adv. Inform. Sci. Serv. Sci.*, vol. 4, no. 6, pp. 324–334, 2012.

[13] G. Gursun, M. Crovella, and I. Matta, "Describing and forecasting video access patterns," in *Proc. IEEE Infocom Mini-Conf.*, 2011, pp. 16–20.

[14] D. Niu, H. Xu, B. Li, and S. Zhao, "Quality-assured cloud bandwidth auto-scaling for video-on-demand applications," in *Proc. IEEE Infocom Conf.*, 2012, pp. 421–425.

[15] S. Islam, J. Keung, K. Lee, and A. Liu, "Empirical prediction models for adaptive resource provisioning in the cloud," *Future Gener. Comput. Syst.*, vol. 28, no. 1, pp. 155–162, 2012.

[16] Y. C. Lee and Y. Albert, "Zomaya: Rescheduling for reliable job completion with the support of clouds," *Future Gener. Comput. Syst.*, vol. 26, no. 8, pp. 1192–1199, 2010.

[17] A. Filali, A. S. Hafid, and M. Gendreau, "Adaptive resources provisioning for grid applications and services," in *Proc. IEEE Int. Conf. Commun.*, 2008, pp. 186–191.

[18] D. Kusic and N. Kandasamy, "Risk-aware limited lookahead control for dynamic resource provisioning in enterprise computing systems," *Cluster Comput.*, vol. 10, no. 4, pp. 395–408, 2007.

[19] W. Zhu, C. Luo, J. Wang, and S. Li, "Multimedia cloud computing," *IEEE Signal Process. Mag.*, vol. 28, no. 3, pp. 59–69, May 2011.

[20] T. Hobfeld, R. Schatz, M. Varela, and C. Timmerer, "Challenges of QoE management for cloud applications," *IEEE Commun. Mag.*, vol. 50, no. 4, pp. 28–36, Apr. 2012.

[21] A. Micah, K. S. Ramesh, and V. Harish, "Algorithms for optimizing the bandwidth cost of content delivery," *Comput. Netw.*, vol. 55, no. 18, pp. 4007–4020, 2011.

[22] S. Chaisiri, B-S Lee, and D. Niyato, "Optimization of resource provisioning cost in cloud computing," *IEEE Trans. Serv. Comput.*, vol. 5, no. 2, pp. 164–177, Apr.-Jun. 2012.

[23] M. Armstrong and J. Vickers, "Competitive nonlinear pricing and bundling," Working paper, Univ. College London, London, United Kingdom, Nov. 2006..

[24] GoGrid. (2012). [Online]. Available: http://www.gogrid.com

[25] M. Cha, H. Kwak, P. Rodriguez, Y-Y Ahn, and S. Moon, "I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system," in *Proc. 7th ACM SIGCOMM Conf. Internet Meas.*, 2007, pp. 1–14.

[26] M. J. Freedman, E. Freudenthal, and D. Mazières, "Democratizing content publication with coral," in *Proc. 1st Conf. Symp. Netw. Syst. Des. Implementation*, 2004, vol. 1, pp. 239–252.

Amr Alasaad [S'09 M'13] received the BSc degree in electrical engineering from King Saud University, the MS degree in electrical and computer engineering from the University of Southern California, and the PhD degree in electrical and computer engineering from the University of British Columbia, in 2000, 2005, and 2013, respectively. He is currently an assistant professor with the National Center for Electronics, Communications and Photonics, King Abdulaziz City for Science and Technology, Riyadh, Saudi Arabia. His research interests are in the broad areas of wireless networks and mobile system, P2P resource sharing protocols, routing and scheduling in wireless mesh networks, content sharing and replication schemes, in which he has co-authored more than 30 technical papers in international journals and conference proceedings. He is a member of the IEEE.

Kaveh Shafiee received the BSc degree in electrical and computer engineering from the University of Tehran, Tehran, Iran, in 2005, MSc degree in electrical engineering, telecommunications from the Sharif University of Technology, Tehran, Iran in 2007, and the PhD degree in electrical and computer engineering from the University of British Columbia, Vancouver, BC, Canada, in 2012. He has been involved in research and development activities in wireless communications since 2005. He is currently with Mojino Inc., Vancouver, BC, involved in developing wireless wearable technologies, where he is responsible for the product design and development and strategic business planning. From 2012 to 2013, he was a research engineer at the Research and Development department, Huawei Technologies Co., Ottawa, ON, where he was involved in the design and evaluation of network architecture and traffic engineering, admission control protocols for next-generation wireless mobile networks. Prior to that, he was a postdoctoral fellow at the department of electrical and computer engineering, University of British Columbia. His research interests include system architecture design for communication networks, the design and development of communication, mobility management, resource management and interworking protocols for communication networks, and modeling, optimization and performance analysis of communication networks including Wireless Sensor Networks, Body Area Networks, Vehicular Ad hoc Networks and Intelligent Transport Systems. He is a member of the IEEE.

Hatim M. Behairy received the MSc degree in electrical engineering and the PhD degree in information technology from George Mason University, Virginia, USA in 1997 and 2002, respectively. He has been with the National Electronics, Communication, and Photonics Center at King Abdulaziz City for Science and Technology, Riyadh, Saudi Arabia since 2003. His research interests are in the design of new error correction coding techniques for next-generation broadband wireless communication systems using turbo-coding principles. He is a member of the IEEE.

Victor C.M. Leung (S'75-M'89-SM'97-F'03) received the BASc (Hons.) degree in electrical engineering from the University of British Columbia (UBC) in 1977, and was awarded the APEBC Gold Medal as the head of the graduating class in the Faculty of Applied Science. He attended graduate school at UBC on a Natural Sciences and Engineering Research Council Postgraduate Scholarship and completed the PhD degree in electrical engineering in 1981. From 1981 to 1987, he was a senior member of Technical Staff and satellite system specialist at MPR Teltech Ltd., Canada. In 1988, he was a lecturer in the Department of Electronics at the Chinese University of Hong Kong. He returned to UBC as a faculty member in 1989, and currently holds the positions of professor and TELUS Mobility Research chair in Advanced Telecommunications Engineering in the Department of Electrical and Computer Engineering. His research interests are in the broad areas of wireless networks and mobile systems, in which he has co-authored more than 700 technical papers in international journals and conference proceedings, 27 book chapters, and co-edited 6 book titles. Several of his papers had been selected for best paper awards. He is a registered professional engineer in the Province of British Columbia, Canada. He was a distinguished lecturer of the IEEE Communications Society. He is a member of the editorial boards of the *IEEE Wireless Communications Letters, Computer Communications*, and several other journals, and has served on the editorial boards of the *IEEE Journal on Selected Areas in Communications Wireless Communications Series*, and *IEEE Transactions on Wireless Communications, Vehicular Technology, and Computers*. He has guest-edited many journal special issues, and provided leadership to the organizing committees and technical program committees of numerous conferences and workshops. He received the IEEE Vancouver Section Centennial Award and 2012 UBC Killam Research Prize. He is a fellow of the IEEE, the Royal Society of Canada, the Engineering Institute of Canada, and the Canadian Academy of Engineering.