Efficient Resource Provisioning and Rate Selection for Stream Mining in a Community Cloud

Shaolei Ren, *Member, IEEE*, and Mihaela van der Schaar, *Fellow, IEEE*

***Abstract*—Real-time stream mining such as surveillance and personal health monitoring, which involves sophisticated mathematical operations, is computation-intensive and prohibitive for mobile devices due to the hardware/computation constraints. To satisfy the growing demand for stream mining in mobile networks, we propose to employ a cloud-based stream mining system in which the mobile devices send via wireless links unclassified media streams to the cloud for classification. We aim at minimizing the classification-energy cost, defined as an affine combination of classification cost and energy consumption at the cloud, subject to an average stream mining delay constraint (which is important in real-time applications). To address the challenge of time-varying wireless channel conditions without *a priori* information about the channel statistics, we develop an online algorithm in which the cloud operator can dynamically adjust its resource provisioning on the fly and the mobile devices can adapt their transmission rates to the instantaneous channel conditions. It is proved that, at the expense of increasing the average stream mining delay, the online algorithm achieves a classification-energy cost that can be pushed arbitrarily close to the minimum cost achieved by the optimal offline algorithm. Extensive simulations are conducted to validate the analysis.**

***Index Terms*—Energy efficiency, mobile cloud, real-time stream mining, resource management, stochastic control.**

I. Introduction

With the advent of ubiquitous wireless access and affordable mobile devices, much of the Internet content has seen a sheering shift towards user-generated content, as attested by, for example, that millions of mobile users upload their video clips onto video-sharing sites like YouTube [24]. Unless appropriately processed (often with little delay), however, the maximum value of exabytes of user-generated content may not be realized. While computation capabilities of mobile devices have improved significantly, they have yet to catch up with the stringent resource requirement for locally performing computation-intensive real-time content processing (e.g., real-time video transcoding, wireless video surveillance). Resolving the conflicts between the soaring demand for seamless access to a large pool of computational resources and the resource scarcity of mobile devices calls for a new computing paradigm, for which mobile cloud computing has been hailed as a promising candidate due to its capability of providing elastic and scalable computational resources to mobile users.

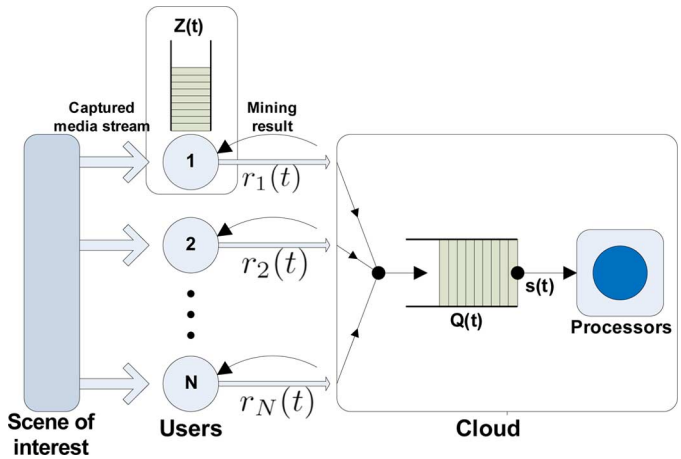


Fig. 1. Cloud-Based Multimedia Stream Mining System.

Multiples types of cloud computing platforms coexist: public cloud (which provides various computing services to the general public on the Internet), private cloud (which is operated solely for a single organization), and community cloud (which provides computing services to several users from a specific community with common concerns/interests such as security). In this paper, we focus on a community cloud that provides real-time stream mining services to multiple users over a wireless network. Example applications include wireless video surveillance, virtual multi-party games, medical services, visual search, spam classification, mobile education, and real-time media content analysis [2]. A block diagram illustrating the cloud-based stream mining system is shown in Fig. 1. Each user is equipped with an energy-constrained mobile device that monitors scenes of interest (e.g., building, person) and encodes the monitoring result into multimedia streams (e.g., video/image sequences). Stream mining is computation intensive, as it involves many mathematical operations and sophisticated machine learning algorithms [4]. Further fueled by the responsiveness and performance requirement, real-time stream mining is therefore computationally prohibitive for mobile devices. Thus, each user transmits the unclassified multimedia streams via wireless links to the community cloud, which then extracts valuable information out of the streams and

Manuscript received March 04, 2012; revised July 27, 2012; accepted November 13, 2012. Date of publication January 16, 2013; date of current version May 13, 2013. This work was supported in part by National Science Foundation under Grant No. 1016081. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xiaoqing Zhu.

S. Ren is with the School of Computing and Information Sciences, Florida International University, Miami, FL 33199 USA (e-mail: sren@cs.fiu.edu).

M. van der Schaar is with the Electrical Engineering Department, University of California, Los Angeles, Los Angeles, CA 90095 USA (e-mail: mihaela@ee.ucla.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2013.2240673

returns the mining results to the users. Nevertheless, mining multimedia streams requires the support from a large number of servers, which thereby consume a significant portion of energy in cloud computing [10].

In this paper, we consider the following three performance metrics:

- **Energy consumption:** Energy consumption is a key performance metric in cloud computing [9], [13].
- **Classification cost:** Errors (e.g., false alarm, missed detection) may occur in stream mining, resulting in classification costs for the users [4].
- **Queueing delay:** In real-time stream mining applications such as visual search, (average) queueing delay incurred at the community cloud needs to be kept under a certain threshold such that users can receive timely mining results [2], [3], [4].

Optimizing these three performance metrics simultaneously is not possible in practice.[1] Thus, depending on the specific application environment, the cloud operator needs to have the ability to perform a desired tradeoff, which, however, is challenged by the fact that the *environment* in which the stream mining system operates is randomly changing over time and the distribution of the underlying stochastic process is often unknown a priori. Specifically, the wireless channels linking the community cloud and the users are time varying with possibly unknown channel gain distributions. To address these challenges, we develop an online computational resource provisioning and rate selection algorithm to minimize the classification-energy cost (defined as an affine combination of the classification cost and energy consumption) while satisfying the average queueing delay constraint. The online decisions are: (1) each user decides its transmission rate; and (2) the cloud operator decides its computational resource provisioning for stream mining. Both decisions are made based on online information, i.e., instantaneous wireless channel conditions, without the knowledge of channel statistics. It can be shown that at the expense of increasing the average queueing delay, the classification-energy cost can be reduced and pushed arbitrarily close to the minimum cost achieved by the optimal offline algorithm. We also extend the analysis to a profit-maximizing cloud operator that can dynamically price its computational resource.

The rest of this paper is organized as follows. Related work is reviewed in Section II. Section III describes the model. In Section IV, we develop a provably-efficient online resource provisioning and rate selection algorithm to minimize the long-term average classification-energy cost subject to queueing delay constraint and the average energy constraint for each user. Simulation results are shown in Section V and finally, concluding remarks are offered in Section VI.

II. Related Work

In this section, we review the existing works related to ours from three perspectives: stream mining, energy-efficient cloud computing, and mobile cloud.

[1]Providing a high classification performance subject to a stringent delay requirement requires more computational resource, which in turn incurs more energy consumption.

Conceptually, a multimedia stream mining system can be viewed as a set of classifiers through which multimedia streams are filtered. Various techniques have been studied to improve the classification performance by optimally configuring networks of classifiers as well as their topologies. For instance, the authors in [3] proposed a randomized distributed algorithm that can iteratively yield the optimal configuration of classifier chains maximizing the classification performance subject to end-to-end delay constraints for real-time multimedia stream mining systems. More recently, [4] studied the topology selection for classifier chains and presented algorithms that can efficiently order and configure the classifiers to tradeoff classification performance with processing delay. Besides optimizing the classifier topology and configuration, other techniques have also been proposed to improve multimedia stream mining performance. By combining multiple musical properties based on principal component analysis and a multi-layer perception neural network, [5] presented an efficient algorithm to construct music descriptors for content-based music retrieval and classification. By using salient objects (e.g., connected image regions that capture the dominant visual properties) to characterize the intermediate image semantics, [6] proposed a framework for mining multilevel image semantics via hierarchical classification. Moreover, the authors proposed an algorithm, referred to as product of mixture-experts, to reduce the size of training images and speed up concept learning for learning more efficiently in high-dimensional feature space. To provide flexible and reliable image retrieval and mining, [7] proposed to utilize the scalability of peer-to-peer networks. In [8] where video event/concept detection was considered, a subspace-based multimedia data mining framework was proposed to deal with semantic gap and rare event/concept detection.

As many megawatts of electricity is required to support the soaring demand for cloud computing, there has been a growing interest in reduce the computing energy consumption. In [9], an online right-sizing algorithm which dynamically turns on/off servers was proposed to minimizes the delay plus energy cost, under the assumption that the electricity price is fixed over time. [10] considered a similar problem but proposed to predict the future service demand using a Markov chain to determine the number of active servers. Later, [11] studied the problem by taking the bandwidth cost into consideration, whereas the proposed solution does not address the queueing delay requirement. [12] quantified the economic gains by scheduling workloads across multiple data centers, which is an empirical study without providing analytical performance bounds on the proposed resource management algorithm. Other studies explored the opportunity of energy saving by executing jobs when and/or where the electricity prices are low (e.g., [13]).

More recently, mobile cloud computing has been hailed as an enabling solution to deliver elastic computing services to mobile devices, the computation capabilities of which have lagged far behind the soaring demand for ubiquitous digital services. Many research proposals have been initiated with the objective of making mobile cloud computing more robust, secure, and efficient. For example, to conserve energy for battery-powered mobile devices, [18] proposed an energy-optimal execution policy that can selectively offload some mobile applica-

tions to the cloud for execution. Similarly, to augment the capability of resource-constrained mobile devices, [19] considered the partition of a single mobile application into multiple components (called weblets) and dynamic adaptation of weblet execution between a mobile device and distant cloud servers. Identifying wide area network delay as a major factor affecting the mobile cloud computing performance, [20] proposed that mobile devices can instantiate a "cloudlet" on nearby infrastructure (achieved through dynamic virtual machine synthesis) and use it via a wireless local area network. Reference [23] reviewed computation offloading techniques for mobile devices and concluded that energy overheads for privacy, security, reliability, as well as data communication ought to be considered before offloading.

None of these works have considered cloud-based mobile stream mining systems operating in random environments with unknown distributions. Moreover, it remains an open problem to achieve a flexible tradeoff among the energy consumption, classification performance and responsiveness of stream mining, which are three important performance metrics that cannot be optimized simultaneously. In this paper, we shall study a cloud-based multimedia stream mining system and address these issues.

III. System Model

We consider a time-slotted system with time slots of equal length indexed by $t = 0, 1, \ldots$. In practice, the actual duration of a time slot depends on the specific application (e.g., every few seconds or minutes for video surveillance applications). We now describe the system implementation and the signaling between the cloud operator and mobile users as follows.

Step 1 (Wireless channel estimation): The cloud operator estimates the wireless channel condition and then sends back the channel state information to the respective mobile users.

Step 2 (Rate Selection and video transmission): The mobile users determine their own transmission rates, update their virtual power deficit queues, and then transmit the captured video streams to the cloud for mining.

Step 3 (Resource provisioning and stream mining): The cloud operator determines its provided computational resource, performs classification tasks on the uploaded streams following the first-in-first-out order, and then updates the job queue.

Step 4 (Classification result notification): The cloud operator notifies each mobile user of the mining result, based on which the mobile user shall take appropriate actions.

In the following, we provide the modeling details of the cloud operator and mobile users, as well as queue dynamics at the community cloud. Key notations are listed in Table I.

TABLE I
List of Notations

Notation	Description
$s(t)$	Provided computational resource
$e(s(t))$	Energy cost
$h_i(t)$	Wireless channel gain between user i and the cloud
$r_i(t)$	Transmission rate of user i
$c_i(t)$	Classification cost of user i
$Q(t)$	Job queue length at the cloud
$g(t)$	Classification-energy cost
β	Classification-energy parameter

A. Cloud Operator

The community cloud operates a cloud-based multimedia stream mining system, which extracts real-time valuable information out of unclassified video streams uploaded via wireless links. The stream mining system can be viewed as a set of classifiers/filters. We consider in this paper *binary* classifiers for the convenience of analysis, while other classifiers can also be considered. A multimedia stream filtered through a binary classifier may have two possible labels: "Positive" and "Negative". The classification result cannot be fully accurate, producing two types of classification errors:

(1) Missed detection: The classifier misses positive data and wrongly labels it as being negative. We denote the detection probability by p_D, and accordingly, the missed detection probability is $1 - p_D$.

(2) False alarm: The classifier labels negative data as being positive. We denote the false alarm probability by p_F.

The performance of a classifier is usually characterized by a two-element tuple (p_D, p_F), where p_F can be expressed as a function in $p_F(p_D, r)$ where $r \geq 0$ is the user's input bit rate and indicates the quality of the input video/image to the classifier. Given a fixed r, $p_F(p_D, r)$ is increasing in $p_D \in [0, 1]$ and the actual shape of $p_F(p_D, r)$ depends on the Detection Error Tradeoff (DET) curve [4].[2] Hence, given a fixed r, we can characterize a classifier using a scalar $p_D \in [0, 1]$. In this paper, we focus on only one binary classifier, and hence the overall classification performance can be conveniently characterized by one parameter p_D. Alternatively, multiple binary classifiers can be combined and abstracted into one binary classifier that yields the ultimate classification result of interest (e.g., whether a thief enters a building). Focusing on the users' rate selection, we assume in this paper that the cloud operator uses a fixed detection probability p_D when classifying all the streams. Note, however, that the false detection probability may not be constant as the input quality varies over time.

When mining multimedia streams, the cloud operator incurs various operational costs (e.g., processors and networking), which increase with the classification complexity. In particular, the computational resource needed to classify a multimedia stream is increasing in the input bit rate [4]. In this paper, we focus on processor energy consumption, which is a major cost factor in cloud computing [9] and can be approximately captured by an increasing and convex cost function of the provided computational resource. For example, given a fixed number of servers and with dynamic voltage and frequency scaling technique applied [14], the minimum energy consumption of all the processors is incurred if all the processors are set at the same frequency, and the resulting energy consumption is convex in the processor frequency (i.e., provided computational resource). Mathematically, we use $s(t) \geq 0$ to denote the total provided computational resource at time t and $e(s(t))$ to denote

[2]The classification performance also depends on many other factors such as training sets and contexts to be classified, which are beyond the scope of this paper.

the corresponding processor energy consumption in the community cloud. For the purpose of mathematical analysis, we assume that $e(0) = 0$ and $e(s(t))$ is a differentiable, increasing and convex function of $s(t)$. The community cloud houses a limited, albeit possibly large, number of servers, each with a maximum processor speed. For the convenience of analysis, we consider homogeneous servers. The server availability constraint naturally limits the total available computational resource. Specifically, the total computational resource that can be provided by the community cloud satisfies $s(t) \leq s_{\max}$, for $t = 0, 1, \ldots$.

B. User

There are N users/nodes gathering environment information (e.g., monitor a building) and uploading unclassified multimedia streams to the community cloud for classification. At time t, each user senses the scene of interest and encodes the sensing result into a video frame/image, and transmits it to the community cloud using a wireless link. Note that, during every time slot, the users transmit the same number of video/image frames. While both sensing and encoding are energy consuming, the (expected) energy consumption associated with sensing and video encoding is relatively stable over time. In contrast, the power consumption of a wireless transmitter is highly dependent on the time-varying channel condition. Thus, we isolate the energy consumed by sensing and encoding from our model and focus on the energy consumption by wireless transmissions.

Denote the channel (power) gain at time t between user i and the community cloud by $h_i(t) \in [h_{i,\min}, h_{i,\max}]$, where $h_{i,\min} \geq 0$ and $h_{i,\max} > h_{i,\min}$ are the minimum and maximum channel gains, respectively, for $i = 1, 2, \ldots, N$. We consider a block-fading channel, i.e., $h_i(t)$ remains constant throughout a frame but may vary across different frames. Without loss of generality, we impose the following assumption on the channel conditions experienced by the users.

Assumption 1: Each user experiences independent channel conditions, and $h_i(t) \in [h_{i,\min}, h_{i,\max}]$ follows a certain i.i.d. (but possibly unknown) distribution, for $i = 1, 2, \ldots, N$ and $t = 0, 1, 2, \ldots$.

Each user can vary its stream quality by adapting its transmission rate. We denote the transmission rate selected by user i by $r_i(t) \in [0, r_{i,\max}]$, where $r_{i,\max}$ is the maximum transmission rate supported by user i (e.g., due to modulation constraint). The transmission power can therefore be written as $d_i(t) = d_i(h_i(t), r_i(t))$, which is decreasing in $h_i(t)$ but increasing in $r_i(t)$. A higher transmission rate $r_i(t)$ indicates a higher stream quality.

Considering energy-constrained users (e.g., smart phones, batter-powered sensors), we impose a long-term average power constraint, denoted by $\bar{D}_i$, for each user i.[3] Due to the imperfection of classifiers at the community cloud, there exist classification errors and also classification costs for the users depending on the input stream quality (as well as the classifier setting/topology, which is beyond the scope of our paper and assumed to be fixed in our study). Assuming that the *a priori* probability of user i's stream being "positive" is ϕ_i, the expected classification cost for user i can be expressed as [4]

$$c_i(t) = c_{M,i}\phi_i(1 - p_D) + c_{F,i}(1 - \phi_i)p_{F,i}(p_D, r_i(t)), \quad (1)$$

where $c_{M,i}$ and $c_{F,i}$ are user i's missed detection cost and false alarm cost, respectively. Clearly, user i can acquire a better classification performance by transmitting at a higher rate (e.g., providing better video quality) to the community cloud. Note that the classification cost is subject to many factors such as the training set, classification algorithm, the specific context to be classified, as well as the input quality. In general, there exists no simple analytic expression of the classification cost in terms of the input bit rate, although it is expected that a higher input bit rate (i.e., better input quality) will generally result in a better classification performance [15]. In this paper, we assume that the classification cost $c_i(t) = c_i(r_i(t)) \geq 0$ is a decreasing function of the input rate $r_i(t)$.

C. Queue Dynamics

A job queue is maintained at the cloud community to store the unfinished or unclassified multimedia streams uploaded by the users. As aforementioned, the computational resource required to classify a stream is increasing in the input bit rate. Without loss of generality, we use a non-negative and finite function $a_i(t) = a_i(r_i(t))$, for $i = 1, 2, \ldots, N$, to represent the required computational resource for a stream with a rate of $r_i(t)$. Note that the specific form of $a_i(r_i(t))$ depends on many factors such as the processor architecture, application, and classifiers. We can use $a_1(t), a_2(t), \ldots, a_N(t)$ to quantify the input/arrival rate to the job queue maintained at the community cloud, and use the provided computational resource $s(t)$ as the *departure rate* of the queue (or service rate in the queueing system). Denote $Q(t)$ as the job queue length (i.e., the amount of computational resource needed to complete all the classification tasks). Thus, the queue length at the community cloud evolves over time following the dynamics specified by

$$Q(t+1) = \max\left[Q(t) - s(t), 0\right] + \sum_{i=1}^{N} a_i(t), \quad (2)$$

assuming that the initial queue is empty, i.e., $Q(0) = 0$. Although the uploaded streams may not necessarily be classified instantaneously (i.e., within the same time slot as they are uploaded) by the community cloud, the classification cost depends on the original transmission rate rather than the actual *departure* rate $s(t)$, since the streams will still be classified according to their transmission rates (which determine the stream quality) after some queueing delay. To address the responsiveness in real-time stream mining, we shall show later that the *average* queue length, which is closely related to the average queueing delay, is upper bounded and the queue length can be shortened at the expense of increasing the classification-energy cost.

D. Remarks

Before proceeding with the analysis, we provide the following remarks regarding our model.

[3]The power constraint is imposed for wireless transmission only. However, as we have mentioned, our model can be easily extended to study joint source-channel coding and take into account both sensing and encoding power consumption.

Remark 1: While we assume in the basic model that $h_i(t)$ is i.i.d. over time, it should be noted that, as shown in [27], more general channel fading processes such as discrete-time Markov chains and even an arbitrarily stochastic process can also be considered without affecting the approach of analysis. Nevertheless, considering more general fading processes does not provide much additional insight, but only complicates the notations and derivations.

Remark 2: We do not consider *job* queues at each individual user, as we try to keep the mobile device and operation as simple as possible. In other words, each user senses objects of interest and transmits real-time streams to the community cloud for stream mining without delay, although the community cloud may defer the classification of some streams. It should be noted that our analysis can be extended if we allow the streams to be buffered at each user, which can then *opportunistically* transmit its data to the community cloud only when the channel is in good conditions to save energy consumption.

IV. Dynamic Resource Provisioning and Rate Selection

In this section, we first formulate the problem of dynamic resource provisioning and rate selection subject to the average power constraint for each user. Then, we develop a provably-efficient online algorithm, which can be implemented in a distributed fashion.

A. Problem Formulation

Control decisions, i.e., the cloud operator's computational resource provisioning $s(t)$ and the transmission rate selection $r_i(t)$ of user i, are made at the beginning of each time slot t. In particular, the users will independently choose their own transmission rates and then the cloud operator chooses the resource provisioning $s(t)$. For notational convenience, we use $\mathbf{w}(t) = (r_1(t), r_2(t), \ldots, r_N(t), s(t))$ to represent the control decisions. Next, we formalize the performance metric. In our study, we concentrate on a *benevolent* cloud operator that aims at minimizing the long-term classification-energy cost.[4]

At time $t = 0, 1, \ldots$, the classification-energy cost can be expressed as

$$g(t) = \sum_{i=1}^{N} c_i(r_i(t)) + \beta e(s(t)), \tag{3}$$

where $\beta \geq 0$ is referred to as the *classification-energy* parameter indicating the relative importance of the energy cost of the community cloud. The long-term average classification-energy cost can therefore be written as

$$\bar{g} \triangleq \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left[g(\tau)\right], \tag{4}$$

where the expectation is taken with respect to the random environment (i.e., wireless fading channels in this paper) given the control decisions. Similarly, we define $\bar{a}_i \triangleq \lim_{t\to\infty} 1/t \sum_{\tau=0}^{t-1} \mathbb{E}\left[a_i(\tau)\right]$, $\bar{s} \triangleq \lim_{t\to\infty} 1/t \sum_{\tau=0}^{t-1} \mathbb{E}\left[s(\tau)\right]$, and $\bar{d}_i \triangleq \lim_{t\to\infty} 1/t \sum_{\tau=0}^{t-1} \mathbb{E}\left[d_i(\tau)\right]$, where $d_i(\tau)$ is the transmission power of user i at time τ. To minimize the long-term classification-energy cost, we aim at developing a control policy that solves the following problem

$$\min_{\mathbf{w}(t), t=0,1,2,\ldots} \quad \bar{g} \triangleq \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left[g(\tau)\right] \tag{5}$$

$$s.t., \quad \bar{d}_i - \bar{D}_i \leq 0, \ \forall i = 1, 2, \ldots, N, \tag{6}$$

$$\sum_{i=1}^{N} \bar{a}_i \leq \bar{s}, \tag{7}$$

$$s(t) \in [0, s_{\max}], \ \forall t, \tag{8}$$

$$r_i(t) \in [0, r_{i,\max}], \ \forall t, \tag{9}$$

where the constraint (6) is the average power constraint for the users, and (7) guarantees the mean-rate queue stability in the long term (i.e., $\lim_{t\to\infty} \mathbb{E}\left[Q(t)\right]/t = 0$). We assume throughout the paper that the problem (5)–(9) is feasible unless otherwise stated. Although the constraints on the maximum classification cost for each user are omitted for brevity, they can be included in our formulation using the same approach of analysis (i.e., adding virtual classification cost queues).

It can be shown based on Caratheodory's theorem that there exists a stationary and randomized offline control policy, solving (5)–(9)[27]. Nevertheless, it requires the full knowledge of channel statistics and is rather challenging, if not impossible, to obtain the optimal control policy explicitly. Hence, we resort to the development of a practical and online algorithm that is provably-efficient compared to the optimal offline algorithm.

B. Algorithm Design

In this subsection, we develop a provably-efficient online resource provisioning and rate selection algorithm. The key idea of the algorithm is as follows: for the queue stability constraint in (7), we use the actual job queue length $Q(t)$, which follows (2), to guide the resource provisioning decision made by the cloud operator. Specifically, when the job queue length is sufficient large, the cloud operator will process some jobs to reduce the queue length and avoid excessive delay. Similarly, for each average power constraint in (6), we define a *deficit* virtual queue and then use the virtual queue lengths to guide the user's rate selection decision. Using this approach, we can show that the online resource provisioning and rate selection algorithm is provably-efficient in the sense that it can yield a long-term classification-energy cost which is arbitrarily close to the minimum possible cost, while bounding the average job queue length at the community cloud and satisfying the average power constraint for each user.

We first define the power deficit queue for user i as $Z_i(t)$, which starts from $Z_i(0) = 0$ and evolves as

$$Z_i(t+1) = \max\left[Z_i(t) + d_i(t) - \bar{D}_i, 0\right], \tag{10}$$

where $d_i(t)$ is the power consumption of user i at time t and $\bar{D}_i$ is the long-term average power constraint for user i. The virtual queue $Z_i(t)$ is used to guarantee the average power constraint for user i. Specifically, it can be shown based on the sample path

[4]Other objective functions, such as resource allocation fairness, can be considered as well without affecting the approach of analysis.

recursion and the queue dynamics (39) that, for any $t > 0$, the following inequality holds

$$\frac{Z_i(t)}{t} \geq \frac{1}{t}\sum_{\tau=0}^{t-1}\left[d_i(t) - \bar{D}_i\right]. \tag{11}$$

Thus, by taking expectation and letting $t \to \infty$, we see that if the virtual queue $Z_i(t)$ is mean rate stable, the inequality (11) becomes $\bar{d}_i - \bar{D}_i \leq 0$, guaranteeing that the average power constraint for user i is satisfied.

Next, we combine the actual job queue at the community cloud with the virtual power deficit queues. Specifically, in the remainder of this paper, we shall use the vectorial expression $\mathbf{\Theta}(t) = [Q(t), Z_1(t), Z_2(t), \ldots, Z_N(t)]$ to conveniently represent the combined set of queues. To utilize the information of queue lengths as a guidance for the resource provisioning and rate selection decision, we define the following quadratic Lyapunov function

$$L(\mathbf{\Theta}(t)) \triangleq \frac{1}{2}Q^2(t) + \frac{1}{2}\sum_{i=1}^{N} Z_i^2(t), \tag{12}$$

where the constant $1/2$ is added for the convenience of mathematical derivations. Note that the quadratic Lyapunov function in (12) is essentially a scalar representation of the actual job queue length and virtual power queue lengths [27]. Next, we define the one-slot conditional Lyapunov drift as follows

$$\Delta(\mathbf{\Theta}(t)) \triangleq \mathbb{E}\left[L(\mathbf{\Theta}(t+1)) - L(\mathbf{\Theta}(t)) \,|\, \mathbf{\Theta}(t)\right]. \tag{13}$$

Using the fact that $\max[q-b,0]^2 \leq (q-b)^2$, it can be shown following the Lyapunov optimization technique [27] that

$$\Delta(\mathbf{\Theta}(t)) \leq B + \sum_{i=1}^{N} Z_i(t)\mathbb{E}\left[d_i(t) - \bar{D}_i \,|\, \mathbf{\Theta}(t)\right] + Q(t)\mathbb{E}\left[\sum_{i=1}^{N} a_i(t) - s(t) \,|\, \mathbf{\Theta}(t)\right], \tag{14}$$

where B is a finite constant satisfying

$$B \geq \frac{1}{2}\sum_{i=1}^{N}\mathbb{E}\left[\left(d_i(t) - \bar{D}_i\right)^2 \,|\, \mathbf{\Theta}(t)\right] + \frac{1}{2}\mathbb{E}\left[\left(\sum_{i=1}^{N} a_i(t)\right)^2 + s^2(t) \,|\, \mathbf{\Theta}(t)\right] - \mathbb{E}\left[\left(\sum_{i=1}^{N} a_i(t)\right)\cdot \min\left[Q(t), s(t)\right] \,|\, \mathbf{\Theta}(t)\right], \tag{15}$$

for any $t = 0, 1, \ldots$.

Next, we choose a parameter $V > 0$ as an indicator of how close the long-term average profit achieved by the online algorithm is to that by the optimal offline algorithm. By adding $V\mathbb{E}\left[g(t) \,|\, \mathbf{\Theta}(t)\right]$ (where $g(t)$ is the classification-energy cost at time t) on both sides of (14), we can obtain the following inequality

$$\Delta(\mathbf{\Theta}(t)) + V\mathbb{E}\left[g(t) \,|\, \mathbf{\Theta}(t)\right] \leq B + V\mathbb{E}\left[g(t) \,|\, \mathbf{\Theta}(t)\right] + \sum_{i=1}^{N} Z_i(t)\mathbb{E}\left[d_i(t) - \bar{D}_i \,|\, \mathbf{\Theta}(t)\right] + Q(t)\mathbb{E}\left[\sum_{i=1}^{N} a_i(t) - s(t) \,|\, \mathbf{\Theta}(t)\right] \tag{16}$$

where the left-hand side is the (one-slot) drift plus classification-energy cost. Instead of directly minimizing the drift plus cost which requires the knowledge of wireless channel statistics, we minimize its upper bound shown on the right-hand side of (16). The formal description of the algorithm is shown in Algorithm 1.

Algorithm 1 Online Algorithm for Benevolent Operator

1: At the beginning of every time slot t, observe the channel state information $(h_1(t), h_2(t), \ldots, h_N(t))$ and the current virtual queue lengths $\mathbf{Z}(t)$

2: Each user i chooses $r_i(t) \in [0, r_{i,\max}]$ to minimize

$$Z_i(t)d_i(t) + Vc_i(r_i(t)) + Q(t)a_i(t) \tag{17}$$

where the classification cost $c_i(r_i(t))$ is defined in (1).

3: The cloud operator chooses $s(t) \in [0, s_{\max}]$ to minimize

$$-Q(t)s(t) + V\beta e(s(t)) \tag{18}$$

where $e(s(t))$ is the energy consumption associated with providing $s(t)$ computational resource.

4: Update the job queue $Q(t)$ and virtual queues $\mathbf{Z}(t+1)$ according to (2) and (39), respectively.

In Algorithm 1, the control decisions are chosen to minimize the right-hand size of (16) based on the currently available information without knowing the distribution of the underlying channel fading processes. The choice of transmission rate (as well as maintaining the virtual power deficit queue lengths) can be performed either by the community cloud or by each individual user. The job queue at the community cloud will guide the cloud operator to select its resource provisioning. As each user only needs to observe its local information, Algorithm 1 can be easily implemented in a distributed fashion.

Now, let us explain the role of the parameter V in controlling the decisions. We first investigate the impacts of V on the cloud operator's resource provisioning decision in Proposition 1.

Proposition 1: The cloud operator's resource provisioning decision $s(t)$ has the following properties:

$$\begin{cases} s(t) = 0, & \text{if } Q(t) \leq b_1 \cdot V, \\ s(t) \in (0, s_{\max}), & \text{if } b_1 \cdot V < Q(t) < b_2 \cdot V, \\ s(t) = s_{\max}, & \text{if } Q(t) \geq b_2 \cdot V, \end{cases} \tag{19}$$

where $b_1 = \beta e'(0)$ and $b_2 = \beta e'(s_{\max})$.

Proof: It can be proved by taking the first-order derivative of $-Q(t)s(t) + V\beta e(s(t))$ with respect to $s(t)$. Specifically, when $Q(t) \leq b_1 \cdot V$, the first-order derivative of $-Q(t)s(t) + V\beta e(s(t))$ is always positive and thus, $s(t) = 0$ minimizes $-Q(t)s(t) + V\beta e(s(t))$. The other two cases can be similarly proved. ∎

Proposition 1 highlights the impact of the job queue length $Q(t)$ on the cloud operator's resource provisioning decision: the cloud operator will choose to perform classification tasks if and only if a sufficiently large number of jobs exist, i.e., $Q(t)$ is large enough. Nevertheless, following this strategy will inevitably result in deferment of classification tasks, which may not be desirable in some real-time stream mining systems. We see from Proposition 1 that by decreasing V, the cloud operator tends to increase the energy consumption while decreasing the queue length. A more rigorous statement about the tradeoff between the cost and queue length will be given Proposition 3 in the next subsection.

Since we have only considered monotonicity of $d_i(r_i)$, $a_i(r_i)$, and $c_i(r_i)$ without specifying the convexity or differentiability, it is difficult to show structural properties of user i's rate selection. Nonetheless, with more assumptions, we can show the following structural property with respect to when user i's should transmit and when it should transmit at the maximum rate.

Proposition 2: Assume that $d_i(r_i)$, $a_i(r_i)$ and $c_i(r_i)$ are twice differentiable and convex with respect to $r_i \in [0, r_{i,\max}]$. User i's transmit rate has the following structural property

$$\begin{cases} r_i(t) = 0, & \text{if } V \leq Q(t)b_{1,q} + Z_i(t)b_{1,z_i}, \\ r_i(t) = r_{i,\max}, & \text{if } V \geq Q(t)b_{2,q} + Z_i(t)b_{2,z_i}, \\ r_i(t) \in (0, r_{i,\max}), & \text{otherwise,} \end{cases} \quad (20)$$

where [5] $b_{1,q} = -d'(r=0)/c'(r=0) \geq 0$, $b_{1,z_i} = -z_i'(r=0)/c'(r=0) \geq 0$, $b_{2,q} = -d'(r=r_{i,\max})/c'(r=r_{i,\max}) \geq 0$, and $b_{2,z_i} = -z_i'(r=r_{i,\max})/c'(r=r_{i,\max}) \geq 0$, in which all the derivatives are respect to r.

Proof: In the second step of the proposed online algorithm, each user i chooses $r_i(t) \in [0, r_{i,\max}]$ to minimize

$$Z_i(t)d_i(t) + Vc_i(r_i(t)) + Q(t)a_i(t), \quad (21)$$

which by assumption is a twice differentiable and convex function of $r_i(t) \in [0, r_{i,\max}]$. Thus, (20) can be proved by the first-order optimality condition. ∎

Given a fixed value of $\beta \geq 0$, Proposition 2 shows that when $V \geq 0$ increases, the users will tend to care more about their classification costs and less about the queue length, and hence, they will choose positive rates even though there is already a long job queue in the community cloud and/or the power deficit queue is quite long.

C. Algorithm Analysis

This subsection proves that the developed online resource provisioning and rate selection algorithm can achieve a provably "good" performance in terms of average classification-energy cost by appropriately tuning the control parameter $V > 0$, as formalized in the following proposition.

Proposition 3: Under Assumption 1, the following statements hold:

a. The long-term average classification-energy cost satisfies

$$\bar{g} = \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left[g(\tau)\right] \leq \bar{g}^{opt} + \frac{B}{V} \quad (22)$$

where $\bar{g}^{opt}$ is the minimum average classification-energy cost that can be obtained by using any possible control policies (including those that have the knowledge of the channel gain distributions).

b. All queues $Q(t)$ and $Z_i(t)$ are mean rate stable (i.e., the constraints (6)–(7) are satisfied).

c. The average job queue length at the community cloud satisfies

$$\lim_{t\to\infty} \sup \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\left[Q(\tau)\right] \leq \frac{B + V \cdot D}{\epsilon}, \quad (23)$$

where $D = g(\epsilon) - \bar{g}^{opt}$, in which $g(\epsilon) \geq \bar{g}^{opt}$ is an average classification-energy cost such that $\sum_{i=1}^{N} \bar{a}_i \leq \bar{s} - \epsilon$ for some $\epsilon > 0$ (i.e., Slater's condition).

Proof: See Appendix. ∎

Proposition 3 shows that, by tuning the control parameter V, the developed online algorithm can achieve an arbitrarily close-to-minimum average classification-energy cost at the expense of increasing the queueing delay at the community cloud while satisfying the average power constraints for each user. Thus, by appropriately selecting the control parameter V, we can achieve a desired tradeoff between the classification-energy cost and queueing delay. Such a tradeoff is very important, especially for real-time stream mining applications which require various levels of interactivity/responsiveness.

We conclude this subsection by noting that in general, appropriate values of V and β need to be determined on a "trial-and-error" basis. Nonetheless, due to the fact that V and β "monotonically" determine the resulting system performance (e.g., increasing V will reduce the average classification-energy cost while increasing the average queue length), we can efficiently identify the appropriate values of V and β using bi-section methods.

D. Profit-Maximizing Cloud Operator

In this subsection, we briefly discuss the case of a profit-maximizing (i.e., "rational") cloud operator that dynamically sets prices for its stream mining service. We concentrate on uniform pricing, i.e., a uniform price is charged to all the users. We denote the price (per unit of computing) at time t by $p(t) \in [0, \bar{p}]$, where $\bar{p}$ is the maximum price that can be charged. Users are charged on a usage basis, which conforms to the common pricing model "pay as you go" in cloud computing services [25]. We can write the cloud operator's profit at time t as

$$\Pi(t) = p(t) \sum_{i=1}^{N} a_i(t) - \alpha \cdot e(s(t)), \quad (24)$$

[5]The derivative of the transmission power with respect to the transmission rate also depends on the channel condition.

where $a_i(t)$ is a function of $r_i(t)$ and also an implicit function in terms of $p(t)$ (which affects user i's decision $r_i(t)$), and α is the electricity/energy price.[6]

Considering rational and price-taking users, we assume that each user chooses its own transmission rate to minimize the weighted sum of its classification cost and payment made to the community cloud. Specifically, $r_i(t)$ is determined by solving the following minimization problem

$$r_i(t) = \arg \min_{r_i(t)\in[0,r_{i,\max}]} \left[c_i(t) + \eta_i a_i(t) p(t)\right], \tag{25}$$

where $\eta_i \geq 0$ is the tradeoff parameter indicating the relative importance of classification performance and monetary cost, and $c_i(t)$ is the classification cost as a function of $r_i(t)$. It is clear that the optimal $r_i(t)$ solving (25) depends on the price $p(t)$ charged by the cloud operator and hence, to emphasize the dependency of $r_i(t)$ on $p(t)$, we use $r_i(t) = r_i(p(t))$ without causing ambiguity. Note that (25) is actually equivalent to solving a utility maximization problem, which is a classic approach to determining the *demand* in economics literature.

Next, we formulate the problem of maximizing the community cloud's long-term profit as

$$\max_{p(t),s(t),t=0,1,2,\ldots} \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}[p(\tau) \sum_{i=1}^{N} a_i(\tau) - \alpha e(s(\tau))], \tag{26}$$

subject to (6)–(9), where $r_i(t)$ is determined by user i by solving (25). Following the analysis for a benevolent cloud operator, we develop an online algorithm (formally described in Algorithm 2), whose performance analysis is similar to Proposition 3 and hence omitted for brevity.

Algorithm 2 Online Algorithm for Profit-Maximizing Operator

1: At the beginning of every time slot t, observe the channel state information $(h_1(t), h_2(t), \ldots, h_N(t))$ and the current virtual queue lengths $\mathbf{Z}(t)$

2: The cloud operator chooses $s(t)$ and $p(t)$ to minimize

$$-Q(t)s(t) + \sum_{i=1}^{N} Z_i(t) d_i(t) - V \cdot \Pi(t), \tag{27}$$

where $\Pi(t)$ is defined in (24).

3: Update the job queue $Q(t)$ and virtual queues $\mathbf{Z}(t+1)$

V. Performance Evaluation

This section presents simulation studies to evaluate the performance of our proposed online resource provisioning and rate selection algorithm. We conduct three sets of simulations:

- Online resource provisioning and rate selection algorithm to minimize classification-energy cost with different V;
- Online resource provisioning and rate selection algorithm to minimize classification-energy cost with different values of β;
- Algorithm 1 versus another two algorithms (i.e., "Equal" "Always" and "Water-filling", which we shall specify later).[7]

In the following, we first discuss the simulation setup and then analyze the results from the simulations in details.

A. Setup

We consider a community cloud serving 10 mobile users. For illustration purposes, the numerical values have been normalized without carrying units wherever applicable.

- Cloud: The function that relates the provided computational resource to the energy consumption is determined as $e(s) = s^2 + 10s$, where the scaling constant is omitted. The maximum value of $s(t)$ is set as 100, which in practice is related to processor architecture and the number of servers housed in the community cloud.
- Users: The average transmission power for each user is normalized to 1, and the maximum transmission rate for each user is 8 bits/symbol. We assume that the wireless transmission from each user to the cloud experiences (independent) Rayleigh fading. Based on [21], [22], we instantiate the wireless transmission power function as $d_i(t) = d_i(h_i(t), r_i(t)) = \Gamma_i \left[2^{r_i(t)} - 1\right]/h_i(t)$, where $d_i(t)$ is the transmission power of user i, $h_i(t)$ is the channel power gain for the wireless link between user i and the cloud operator, and $\Gamma_i \geq 1$ captures factors (e.g., overhead, channel coding) which reduce the actual transmission rate. Note that our analysis also applies to other transmission power models such as the one studied in [16]. As aforementioned, the actual classification cost depends on many factors such as training set, classification algorithm, the specific context to be classified, the missed detection cost, and false alarm cost. For simplicity and to limit the number of free parameters, we assume a synthetic classification cost function in terms of the transmission rate, expressed as $c_i(t) = 2^{-r_i(t)}$, for $i = 1, 2, \ldots, 10$ and $t = 0, 1, 2, \ldots$. Moreover, for illustration purposes, we consider $a_i(r_i) = r_i$, i.e., the computational resource required for stream mining increases linearly with the transmission rate.

B. Main Results

In this subsection, we discuss three sets of simulation results.

1) Different Values of v: We show in Fig. 2 the average energy cost, average classification cost, and average queue length at the community cloud, given different values of V. Note that throughout the simulation, the "average" number at time t in all the figures is calculated by summing up all the past values (up to time t) and then dividing the sum by $t+1$. We see in Fig. 2 that as V increases, both the average energy cost and classification cost decrease, whereas the average queue length at the community cloud increases (i.e., average queueing delay increases). This is because, by increasing V, the cloud operator tends to reduce the energy consumption by queueing up the jobs until

[6]Time-varying electricity price can be considered as well by abstracting it into part of the random *environment*.

[7]As we have proved that the classification-energy achieved by our online algorithm can be arbitrarily close to that by the optimal offline algorithm (at the expense of increasing the average delay of stream mining) and a flexible tradeoff among classification cost, energy consumption, and queueing delay can be achieved, we do not show the performance comparison against other algorithms such as prioritized algorithms.

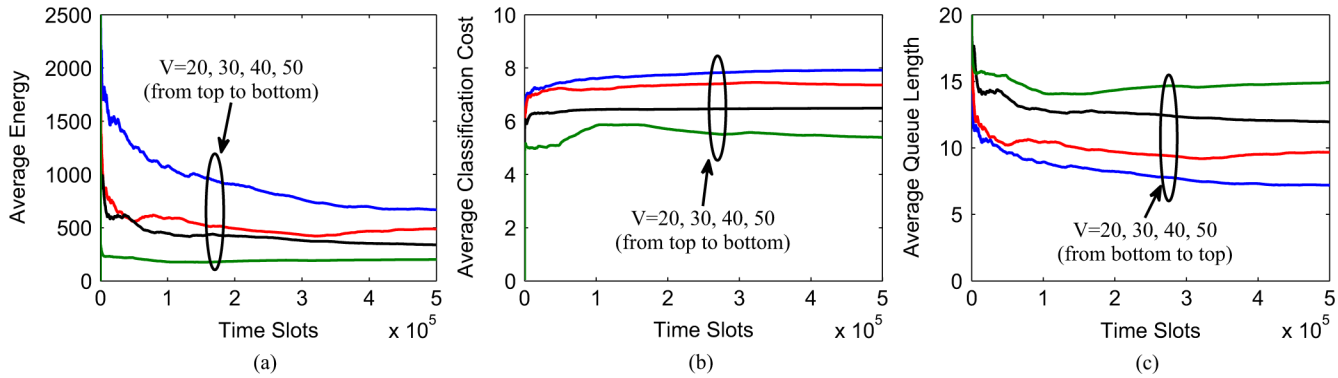


Fig. 2. Algorithm 1 with different values of $V \cdot \beta = 0.01$. (a) Energy cost. (b) Classification cost. (c) Queue length.

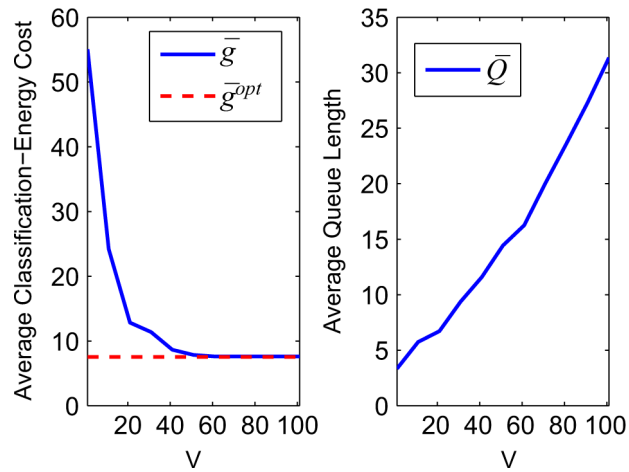


Fig. 3. Average classification-energy cost and average queue length under different $V \cdot \beta = 0.01$.

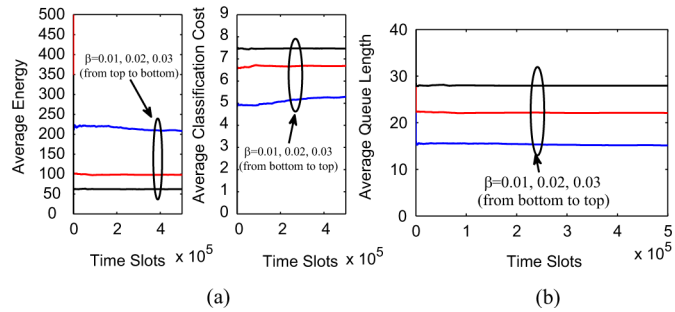


Fig. 4. Impact of $\beta \cdot V = 50$. (a) Average energy and classification cost. (b) Average queue length.

the queue length exceeds a certain threshold, which inevitably increases the waiting queue length. The results in Fig. 2 also verify Proposition 3, which shows the flexible tradeoff between the classification-energy cost and the queue length. Next, we show in Fig. 3 the average classification-energy cost and average queue length given different values of $V \in [1, 101]$. The result highlights again the role of V in governing the tradeoff between the classification-energy cost and queue length, thereby validating Proposition 3.

2) Different Values of β: The classification-energy parameter β governs the relative importance of energy cost compared to the classification cost. In particular, when β increases, the community cloud tends to care more about its power consumption. Thus, by choosing a larger value of β, we expect that the average energy cost will decrease. The result in Fig. 4(a) shows that when β increases, the average energy cost is significantly reduced while the average classification cost increases. Thus, the proposed algorithm provides a flexible tradeoff between the energy cost and classification cost. By appropriately tuning the value of β, we can achieve a desired performance balancing the energy consumption and classification cost. With $e(s) = s^2 + 10s$ for $s \in [0, 100]$, we can see from the third step in the online algorithm that the optimal $s(t)$ is chosen as $s(t) = \min[100, \max[Q(t)/2V\beta - 5, 0]]$. Thus, the actual queue length is related to the control parameter β, as shown in Fig. 4(b). In particular, when β increases, we notice from the decision $s(t) = \min[100, \max[Q(t)/2V\beta - 5, 0]]$ that the cloud operator tends to process fewer jobs each time slot, thereby increasing the average queue length.

3) Comparison With "Equal", "Always" and "Water-Filling": Next, we compare our proposed resource provisioning and rate selection algorithm against three other algorithms, i.e., "Equal" "Always" and "Water-filling".

- "Equal": In the "Equal" algorithm, the users split their power budgets equally over time. In other words, all the users transmit using their respective average powers, regardless of the instantaneous channel gains. The community cloud, however, applies "Step 3" to determine its resource provisioning.
- "Always": In the "Always" algorithm, all the users transmit using their respective average powers, regardless of the channel gains, and the community cloud always tries to finish processing the jobs without any queueing delay. This is essentially a myopic greedy algorithm.
- "Water-filling": All the users employ the classic "water-filling" power allocation algorithm, which has been shown to be rate maximizing [17].[8] The community cloud applies "Step 3" to determine its resource provisioning.

We expect that "Equal" outperforms "Always" in terms of the average classification-energy cost, as the community cloud

[8]"Water-filling" maximizes the average rate rather than minimizing the average classification cost.

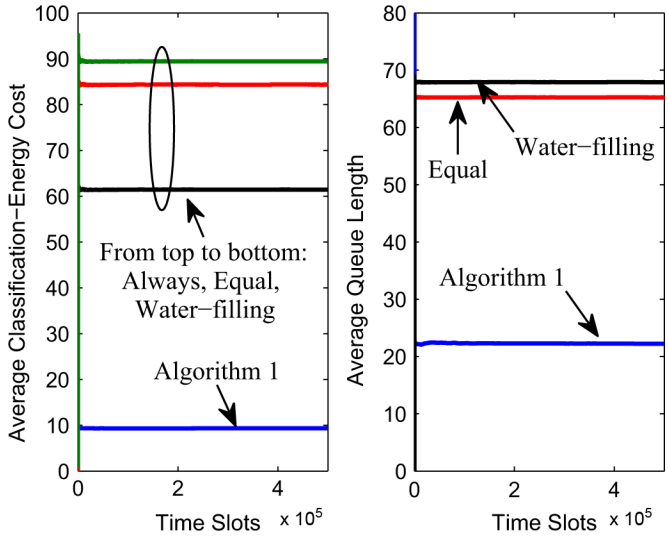


Fig. 5. Algorithm 1 versus "Equal", "Always" and "Water-filling". $V = 30$, $\beta = 0.02$.

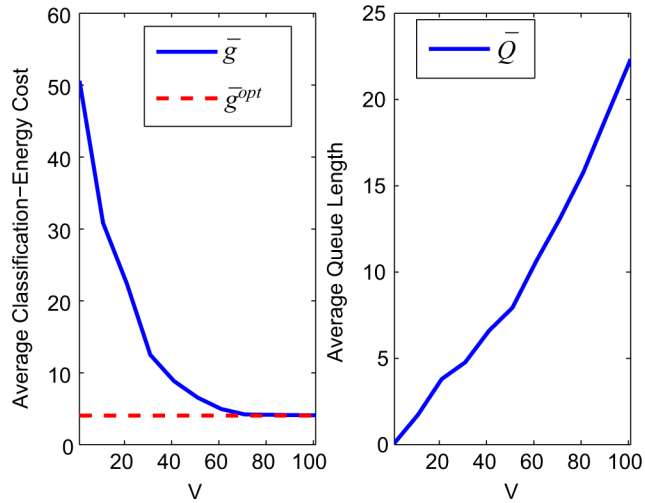


Fig. 6. Average classification-energy cost and average queue length under different V.

tries to delay processing the workloads to save energy consumption. Fig. 5 shows that "Equal" achieves a less classification-energy cost than "Always", while increasing the queue length at the community cloud.[9] Since "Water-filling" maximizes the (average transmission rate), it induces a lower classification cost and hence reduces the average classification-energy cost compared to "Equal" and "Always". Nevertheless, under "Water-filling", the users transmit at higher rates, which in turns results in a longer job queue at the cloud operator. Fig. 5 also shows that our proposed resource provisioning and rate selection algorithm achieves a much less classification-energy cost than "Equal", "Always" and "Water-filling", although it has a larger queueing delay than "Always". Since a flexible tradeoff between the classification cost and energy consumption can be achieved by appropriately tuning the parameter β, the proposed online algorithm can also outperform "Equal", "Always" and "Water-filling" in terms of the classification cost given an appropriate β. The result is similar to Fig. 5 and hence, is omitted for brevity. Moreover, if the stream mining performance (i.e., defined as the sum of classification cost in this paper) is a major concern, the proposed online algorithm can still achieve an arbitrarily close-to-optimal stream mining performance at the expense of increasing the queueing delay and energy consumption at the community cloud. The minimum classification-energy cost, which requires the information of channel statistics to compute in practice, is not shown in Fig. 5, as it has been proved that our proposed algorithm can achieve an average classification-energy cost arbitrarily close to the minimum value by increasing the value of V.

C. Others

In this subsection, we briefly provide numerical results for a practical stream-mining system and for a profit-maximizing cloud operator.

[9]The queue length associated with "Always" is almost zero all the time. Thus, we do not show its queue length in Fig. 5.

1) Application to a Real System: We now apply our online algorithm to a real-time stream mining system, which has been used extensively in the past to evaluate various classifier topology configuration algorithms (e.g., [4]). In particular, given an optimally configured stream mining system with four binary classifiers, we rescale classification costs to the interval $[0, 1]$ for our purpose. For wireless channels, we consider a correlated Rayleigh fading channel model (with a normalized Doppler frequency shift of 0.02) simulated based on the Jake's model [26]. The energy cost function of the community cloud is still $e(s) = s^2 + 10s$ for $s \in [0, 100]$. The simulation results are shown Fig. 6 which illustrates the average classification-energy cost and average queue length under different $V \in [1, 101]$. Fig. 6 validates our analysis regarding the role of V: as V increases, the average classification-energy cost decreases whereas the average queue length increases. Other results are also similar to those we have obtained using synthetic settings, and thus they are omitted for brevity.

2) Profit-Maximizing Cloud Operator: Here, we provide numerical results to illustrate the impact of V on the average profit and queue length at the cloud. It can be seen from Fig. 7 that as V increases, the average profit obtained by the cloud operator also increases (at the expense of increasing the queue backlog at the cloud), which validates our analysis. The other numerical results (e.g., heterogeneous networks) are omitted, as they are similar to the results obtained for a benevolent cloud operator which aims at minimizing the average classification-energy cost.

VI. Conclusion

In this paper, we considered a community cloud performing real-time stream mining for multiple users over a wireless network, and studied the problem of dynamic resource provisioning and rate selection without knowing the distribution of wireless fading channel gains. We developed an online resource provisioning and rate selection algorithm that can achieve an arbitrarily close-to-minimum average classification-energy cost while satisfying the average power constraint for each user.

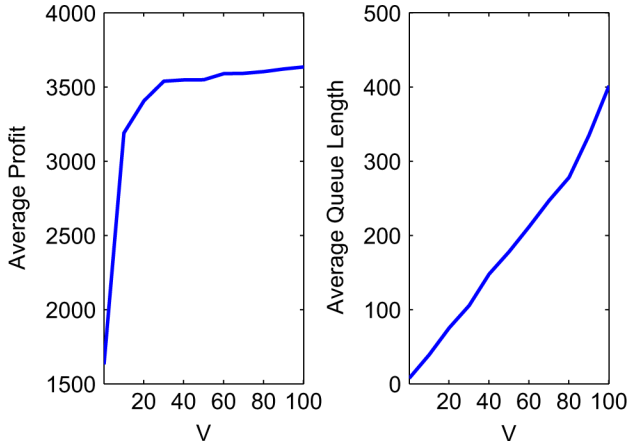


Fig. 7. Average profit and average queue length under different V.

It was shown that a desired tradeoff among the classification performance, energy consumption, and queueing delay of stream mining can be achieved by appropriately choosing the relevant control parameters. The simulation result validated our analysis and also showed that the proposed algorithm outperforms other algorithms (that do not dynamically perform resource provisioning and/or rate selection) in terms of the average classification-energy cost.

APPENDIX
PROOF OF PROPOSITION 3

First, we note that at every time slot $t = 0, 1, 2, \ldots$, Algorithm 1 essentially minimizes the right hand side of the upper bound on the Lyapunov drift plus the classification-energy cost. Thus, for every time slot t, we have

$$\begin{aligned}&\Delta(\mathbf{\Theta}(t)) + V\mathbb{E}\left[g(t)\,|\,\mathbf{\Theta}(t)\right]\\ &\le B + V\mathbb{E}\left[g^*(t)\,|\,\mathbf{\Theta}(t)\right]\\ &\quad + \sum_{i=1}^{N} Z_i(t)\mathbb{E}\left[d_i^*(t) - \bar{D}_i\,|\,\mathbf{\Theta}(t)\right]\\ &\quad + Q(t)\mathbb{E}\left[\sum_{i=1}^{N} a_i^*(t) - s^*(t)\,|\,\mathbf{\Theta}(t)\right], \end{aligned} \tag{28}$$

where $a_i^*(t)$ and $s^*(t)$ are the input rate to the job queue and resource provisioning under any alternative control policies, respectively, and $g^*(t)$ and $d_i^*(t)$ are the resulting classification-energy cost and energy consumption, respectively. We consider a stationary and randomized offline control policy $\mathcal{Z}$, which minimizes the classification-energy cost. In particular, given the control policy $\mathcal{Z}$, we have

$$\mathbb{E}\left[g^*(t)\,|\,\mathbf{\Theta}(t)\right] = \mathbb{E}\left[g^*(t)\right] = \bar{g}^{opt}, \tag{29}$$

$$\mathbb{E}\left[d_i^*(t)\,|\,\mathbf{\Theta}(t)\right] = \mathbb{E}\left[d_i^*(t)\right] \le \bar{D}_i, \tag{30}$$

$$\mathbb{E}\left[\sum_{i=1}^{N} a_i^*(t) - s^*(t)\,|\,\mathbf{\Theta}(t)\right] \le 0. \tag{31}$$

Then, by substituting $\mathcal{Z}$ and (29)–(31) into the right hand side of(28), we obtain

$$\Delta(\mathbf{\Theta}(t)) + V\mathbb{E}\left[g(t)\,|\,\mathbf{\Theta}(t)\right] \le B + V\bar{g}^{opt}. \tag{32}$$

Then, by taking the expectation on both sides of (32) with respect to $\mathbf{\Theta}(t)$, we obtain

$$\mathbb{E}\left[L(\mathbf{\Theta}(t+1))\right] - \mathbb{E}\left[L(\mathbf{\Theta}(t))\right] + V\mathbb{E}\left[g(t)\right] \le B + V\bar{g}^{opt}. \tag{33}$$

Thus, by summing up (33) from $0, 1, \ldots, t$ and by dividing the sum by V, we have, for any $t > 0$,

$$\begin{aligned}\frac{1}{t}\sum_{\tau=0}^{t-1}\mathbb{E}\left[g(t)\right] &\le \bar{g}^{opt} + \frac{B}{V} + \frac{\mathbb{E}\left[L(\mathbf{\Theta}(0))\right]}{Vt} - \frac{\mathbb{E}\left[L(\mathbf{\Theta}(t+1))\right]}{Vt}\\ &\le \bar{g}^{opt} + \frac{B}{V} + \frac{\mathbb{E}\left[L(\mathbf{\Theta}(0))\right]}{Vt}. \end{aligned} \tag{34}$$

By taking $t \to \infty$, we prove part **a** of Proposition 3.

By summing up (33) from $0, 1, \ldots, t$, it follows directly that

$$\mathbb{E}\left[L(\mathbf{\Theta}(t+1))\right] \le \mathbb{E}\left[L(\mathbf{\Theta}(0))\right] + \left[B + V\left(\bar{g}^{opt} - g_{\min}\right)\right]\cdot t, \tag{35}$$

where $g_{\min}$ satisfies $g_{\min} \le \mathbb{E}\left[g(t)\right]$ and $g_{\min}$ exists due to the boundedness condition. Then, by the definition of $L(\mathbf{\Theta}(t)) \triangleq 1/2Q^2(t) + 1/2\sum_{i=1}^{N} Z_i^2(t)$, we can derive

$$\mathbb{E}\left[Z_i^2(t)\right] \le 2\mathbb{E}\left[L(\mathbf{\Theta}(0))\right] + 2\left[B + V\left(\bar{g}^{opt} - g_{\min}\right)\right]\cdot t, \tag{36}$$

for $i = 1, 2, \ldots, N$. However, because the variance of $|Z_i(t)|$ is always non-negative, we have $\mathrm{var}(|Z_i(t)|) = \mathbb{E}\left[Z_i^2(t)\right] - \mathbb{E}\left[|Z_i(t)|\right]^2 \ge 0$. Hence, for any $t > 0$, we have

$$\mathbb{E}\left[|Z_i(t)|\right] \le \sqrt{2\mathbb{E}\left[L(\mathbf{\Theta}(0))\right] + 2\left[B + V\left(\bar{g}^{opt} - g_{\min}\right)\right]\cdot t}. \tag{37}$$

Therefore, by dividing both sides of (37) by t and taking the limit $t \to \infty$, we prove

$$\lim_{t\to\infty}\frac{\mathbb{E}\left[|Z_i(t)|\right]}{t} \le 0 \tag{38}$$

which is equivalent to $\lim_{t\to\infty}\mathbb{E}\left[Z_i(t)\right]/t = 0$ (i.e., mean-rate stable). Recall that the virtual power deficit queue $Z_i(t)$ evolves following

$$Z_i(t+1) = \max\left[Z_i(t) + d_i(t) - \bar{D}_i, 0\right], \tag{39}$$

starting from $Z_i(0) = 0$. Hence, $Z_i(t)$ satisfies

$$Z_i(t+1) - Z_i(t) \ge d_i(t) - \bar{D}_i. \tag{40}$$

By the basic sample path property and dividing the sum by t, we have, for any $t > 0$, we obtain

$$\frac{Z_i(t)}{t} = \frac{Z_i(t)}{t} - \frac{Z_i(0)}{t} \ge \frac{1}{t}\sum_{\tau=0}^{t-1} d_i(t) - \bar{D}_i. \tag{41}$$

Then, by taking the limit $t \to 0$, we obtain $\bar{d}_i - \bar{D}_i \le \lim_{t\to\infty}\mathbb{E}\left[Z_i(t)\right]/t = 0$, which means that the average power constraint is satisfied for user $i = 1, 2, \ldots, N$. Other parts of

Proposition 3 can be proved similarly, following the Lyapunov optimization technique [27]. The details are omitted for brevity. ■

REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, Above the Clouds: A Berkeley View of Cloud Computing, UC Berkeley, 2009, Tech. Rep. UCB/EECS-2009-28.

[2] B. Girod, V. Chandrasekhar, R. Grzeszczuk, and Y. A. Reznik, "Mobile visual search: Architectures, technologies, and the emerging MPEG standard," *IEEE Multimedia*, vol. 18, no. 3, pp. 86–94, Jul.–Sep. 2011.

[3] B. Foo and M. v. d. Schaar, "Distributed classifier chain optimization for real-time multimedia stream mining systems," in *IS&T/SPIE Multimedia Content Access, Algorithms and Systems II*, Jan. 2008.

[4] R. Ducasse, D. Turaga, and M. v. d. Schaar, "Adaptive topologicoptimization for large-scale stream mining," *IEEE J. Select. Topics Signal Process.*, vol. 4, no. 3, pp. 620–636, Jun. 2010.

[5] J. Shen, J. Shepherd, and A. H. H. Ngu, "Towards effective content-based music retrieval with multiple acoustic feature combination," *IEEE Trans. Multimedia*, vol. 8, no. 6, pp. 1179–1189, Dec. 2006.

[6] J. Fan, Y. Gao, H. Luo, and R. Jain, "Mining multilevel image semantics via hierarchical classification," *IEEE Trans. Multimedia*, vol. 10, no. 2, pp. 167–187, Feb. 2008.

[7] J. J. Chen, C. J. Hu, and C. R. Su, "Scalable retrieval and mining with optimal peer-to-peer configuration," *IEEE Trans. Multimedia*, vol. 10, no. 2, pp. 209–220, Feb. 2008.

[8] M. L. Shyu, Z. Xie, M. Chen, and S. C. Chen, "Video semantic event/concept detection using a subspace-based multimedia data mining framework," *IEEE Trans. Multimedia*, vol. 10, no. 2, pp. 252–259, Feb. 2008.

[9] M. Lin, A. Wierman, L. L. H. Andrew, and E. Thereska, "Dynamic right-sizing for power-proportional data centers," in *Proc. IEEE Infocom*, 2011.

[10] B. Guenter, N. Jain, and C. Williams, "Managing cost, performance and reliability tradeoffs for energy-aware server provisioning," in *Proc. IEEE Infocom*, 2011.

[11] N. Buchbinder, N. Jain, and I. Menache, "Online job migration for reducing the electricity bill in the cloud," *IFIP Netw.*, 2011.

[12] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs, "Cutting the electric bill for Internet-scale systems," in *Proc. ACM Sigcomm*, 2009.

[13] Z. Liu, M. Lin, A. Wierman, S. Low, and L. H. Andrew, "Greening geographical load balancing," in *Proc. ACM Sigmetrics*, 2011.

[14] J. R. Lorch and A. J. Smith, "Improving dynamic voltage scaling algorithms with PACE," in *Proc. ACM Sigmetrics*, 2001.

[15] X. Zhu and X. Wu, "Class noise vs. attribute noise: A quantitative study of their impacts," *Artif. Intell. Rev.*, vol. 22, pp. 177–210, 2004.

[16] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, "Energy consumption in mobile phones: A measurement study and implications for network applications," in *Proc. ACM/USENIX Internet Measurement Conf.*, 2009.

[17] A. Goldsmith and S.-G. Chua, "Adaptive coded modulation for fading channels," *IEEE Trans. Commun.*, vol. 46, no. 5, pp. 595–602, May 1998.

[18] Y. G. Wen, W. W. Zhang, and H. Y. Luo, "Energy-optimal mobile application execution: Taming resource-poor mobile devices with cloud clones," in *Proc. IEEE Int. Conf. Computer Commun.*, 2012.

[19] X. W. Zhang, A. Kunjithapatham, S. Jeong, and S. Gibbs, "Towards an elastic application model for augmenting the computing capabilities of mobile devices with cloud computing," *Mobile Netw. Applicat.*, vol. 16, no. 3, pp. 270–284, Jun. 2011.

[20] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, Oct.–Dec. 2009.

[21] C.-H. F. Fung, W. Yu, and T. J. Lim, "Precoding for the multiantenna downlink: Multiuser SNR gap and optimal user ordering," *IEEE Trans. Commun.*, vol. 55, no. 1, pp. 188–197, Jan. 2007.

[22] S. Jagannatha and J. M. Cioffi, "Optimality of FDMA in Gaussian multiple-access channels with non-zero SNR margin and gap," in *Proc. IEEE Globecom*, 2006.

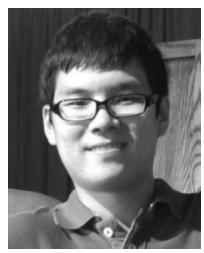
[23] K. Kumar and Y. H. Lu, "Cloud computing for mobile users: Can offloading computation save energy?," *IEEE Comput.*, vol. 43, no. 4, pp. 51–56, Apr. 2010.

[24] V. K. Singh, R. Jain, and M. S. Kankanhalli, "Motivating contributors in social media networks," in *Proc. ACM SIGMM Workshop on Social Media*, 2009.

[25] [Online]. Available: http://aws.amazon.com/ec2/pricing/.

[26] W. C. Jakes, Jr, *Microwave Mobile Communications*. New York, NY, USA: Wiley, 1974.

[27] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*. San Rafael, CA, USA: Morgan & Claypool, 2010.



Shaolei Ren (M'12) received the B.E., M.Phil., and Ph.D. degrees, all in electrical engineering, from Tsinghua University, Beijing, China, in July 2006, Hong Kong University of Science and Technology in August 2008, and University of California, Los Angeles, CA, USA, in June 2012, respectively.

Since August 2012, he has been with School of Computing and Information Sciences, Florida International University, Miami, FL, USA, as an Assistant Professor. His research interests include network economics and cloud computing, with an emphasis on energy-efficient scheduling for delay-sensitive services.

Dr. Ren received the Best Paper Award from IEEE International Conference on Communications in 2009.

Mihaela van der Schaar (F'10) is Chancellor's Professor in the Electrical Engineering Department at the University of California, Los Angeles, Los Angeles, CA, USA. Her research interests include multimedia systems, networking, communication, and processing, dynamic multi-user networks and system designs, online learning, network economics and game theory.

Prof. van der Schaar is a Distinguished Lecturer of the Communications Society for 2011–2012, the Editor-in-Chief of the IEEE TRANSACTIONS ON MULTIMEDIA and a member of the Editorial Board of the IEEE JOURNAL ON SELECTED TOPICS IN SIGNAL PROCESSING. She received an NSF CAREER Award (2004), the Best Paper Award from IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (2005), the Okawa Foundation Award (2006), the IBM Faculty Award (2005, 2007, and 2008), and the Most Cited Paper Award from *EURASIP: Image Communications Journal* (2006). She was formerly an Associate Editor for the IEEE TRANSACTIONS ON MULTIMEDIA, SIGNAL PROCESSING LETTERS, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, Signal Processing Magazine, etc. She received three ISO awards for her contributions to the MPEG video compression and streaming international standardization activities, and holds 33 granted US patents.