Information Sciences 301 (2015) 169–180

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Optimal scheduling for data transmission between mobile devices and cloud

Weiwei Fang [a,c,*], Xiaoyan Yin [b], Yuan An [c], Naixue Xiong [d], Qiwang Guo [c], Jing Li [c]

[a] School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China
[b] School of Information Science and Technology, Northwest University, Xi'an 710127, China
[c] State Key Lab of Astronautical Dynamics of China, Xi'an 710043, China
[d] School of Computer Science, Colorado Technical University, Colorado Springs, CO 80907, USA

ARTICLE INFO

Article history:
Received 14 August 2014
Received in revised form 23 November 2014
Accepted 31 December 2014
Available online 8 January 2015

Keywords:
Mobile cloud computing
Energy efficiency
Lyapunov optimization
Dynamic scheduling
Queue stability

ABSTRACT

Mobile cloud computing has emerged as a new computing paradigm promising to extend the capabilities of resource-constrained mobile devices. In this new paradigm, mobile devices are enabled to offload computing tasks, report sensing records, and store large files on the cloud through wireless networks. Therefore, efficient data transmission has become an important issue affecting user experiences on mobile cloud. Considering the limited battery energy of mobile devices and different application requirements on transmission delay, this study presents an online control algorithm (OPERA) based on the Lyapunov optimization theory for optimally scheduling data transmission between mobile devices and cloud. The OPERA algorithm is able to make control decisions on application scheduling, interface selection and packet dropping to minimize a joint utility of network energy cost and packet dropping penalty, without requiring any statistical information of traffic arrivals and link throughputs. Rigorous analysis and extensive simulations have demonstrated its distinguished performance in terms of utility optimality, system stability and service delay.

1. Introduction

Mobile devices, e.g., smartphone, tablet computer and wearable device, have become an essential part of human life as the most convenient information tools. However, the advances in hardware and battery have been slow to respond to the application demands evolved over the years, which significantly impede the improvement of service qualities and user experiences. In recent years, this problem has been addressed by researchers through cloud computing [34]. Mobile cloud computing is a new paradigm that leverages cloud computing principles and technologies to extend capabilities of mobile devices by executing computing tasks, analyzing sensing results and storing large files in resource-rich cloud environments [1,10]. With the surging popularity of mobile cloud computing, it has been predicted that global mobile data traffic will grow three times faster than traffic from wired devices from 2013 to 2018, and will exceed the latter by 2018 [7]. Unlike wired devices, current mobile devices and their daily use are seriously constrained by the limited capacity of battery energy. The frequent interactions between mobile devices and cloud would incur a significant energy burden on mobile devices.

[*] Corresponding author at: School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China.
E-mail addresses: wwfang@bjtu.edu.cn (W. Fang), yinxy@nwu.edu.cn (X. Yin), yanadl@gmail.com (Y. An), nxiong@coloradotech.edu (N. Xiong), qwguoadl@gmail.com (Q. Guo), jliadl@gmail.com (J. Li).

http://dx.doi.org/10.1016/j.ins.2014.12.059
0020-0255/© 2015 Elsevier Inc. All rights reserved.

Thus, the data transmission strategy for mobile cloud devices has to take into consideration not only satisfying application QoS requirements for guaranteeing user experiences, but also reducing network energy consumption for prolonging operational lifetime.

To support pervasive Internet access, current mobile devices are increasingly being equipped with more than one wireless interfaces, such as WiFi, 3G, and LTE, that are heterogeneous in terms of network availability, achievable throughput and energy expenditure [10,12]. Previous studies [10,22,24] have exploited the characteristics of heterogeneous wireless interfaces to propose energy-efficient network selection strategies for the delay-tolerant mobile applications, e.g., mobile application update [3] and mobile cloud storage [22]. These algorithms make control decisions to determine whether and when to defer a transmission as well as which link from a set of current available ones to use for a transmission, so as to balance the tradeoff between energy consumption and transmission delay. While network energy minimization has been well studied for delay-tolerant applications on mobile devices, existing solutions provide either no delay guarantees or provide weak guarantees on average delay [18]. Actually, not all cloud-based mobile applications can tolerant unbounded, excessive transmission delay. For example, the mobile crowd-sensing data is fresh and valid in a certain time for target/event detection, classification and tracking [5,6,31]. Another typical example is the cloud-based online games in which user inputs should be uploaded to the game servers in time for guaranteeing gaming experience [4,9]. The problem would be more complicated when there exist traffic flows from both delay-unbounded and delay-bounded applications [21]. To the best of our knowledge, no work has studied joint scheduling of delay-unbounded and delay-bounded application traffic for optimizing energy consumption on mobile devices with multiple wireless interfaces.

To address these challenges, we present a new control algorithm, OPERA, for OPtimal schEduling for data tRAnsmission of both delay-unbounded and delay-bounded applications between mobile devices and cloud, based on the Lyapunov optimization theory. OPERA is an online algorithm that minimizes a joint utility of network energy cost and packet dropping penalty, by making greedy decisions to schedule applications, select interfaces and drop packets every time slot. We show that OPERA is able to achieve a time average utility [8] within a deviation of $\mathcal{O}(1/V)$ from optimality, while bounding the system queue length by $\mathcal{O}(V)$, where V is a non-negative control parameter representing a design knob of the stability-utility tradeoff (i.e., how much we emphasize utility maximization compared to system stability). Meanwhile, OPERA is able to ensure persistent service with bounded worst-case delay for delay-bounded applications. OPERA operates without requiring any statistical knowledge of traffic arrivals and link qualities, and is computationally efficient for implementing on resource-constrained mobile devices. Results from rigorous theoretical analysis and trace-driven empirical evaluation demonstrate its effectiveness in terms of utility optimality, system stability and service delay.

The remainder of this paper is organized as follows. Section 2 reviews some related studies. In Section 3, we present the problem formulation, and in Section 4, we develop our online algorithm, OPERA, as well as provide its performance analysis. The analysis is further validated by extensive simulation experiments introduced in Section 5. Finally, Section 6 concludes this research work.

2. Related work

The contribution of this work lies in the intersection of the following two important cutting-edge research issues.

2.1. Energy-efficient transmission strategies for mobile devices

Current mobile devices are severely constrained by limited battery capacity. However, technology trends for batteries indicate that energy will still remain as the primary bottleneck for mobile devices [28]. Existing researches have revealed that the power drained by network interfaces constitutes a large fraction of the total energy consumption of a mobile device [12,25]. With the rising popularity of mobile cloud computing, the situation will become even worse due to the frequent interactions between mobile devices and cloud [16]. Hence, research efforts have been focused on how to schedule wireless interfaces adaptively to achieve energy efficient data transmission for mobile devices. One interesting issue is the so-called “tail energy” [12], i.e., the amount of energy that a wireless interface consumes in the high-power state after the completion of data transmission. Existing schemes (e.g., [3,29]) have proposed to aggregate small transmissions into large ones through prefetching and delayed transfer, so that the period of time that the wireless interface stays in the high-power state can be well reduced for saving additional energy. Based on the measurement studies of the energy consumption characteristics of different mobile networking technologies [3,12], some prior researches have attempted to exploit multiple wireless interfaces to improve energy efficiency on mobile devices. CoolSpots [21] aims to reduce the energy consumption by deciding whether and when to use WiFi or Bluetooth based on the current application’s bandwidth requirement. Context-for-Wireless [23] employs the statistical information of historical context to decide whether and when to power on WiFi to improve the energy efficiency of data transmission in cellular networks. These two schemes are mainly interested in determining the lowest energy link among a set of available ones at a given instant, rather than trading off delay for saving energy. Based on the Lyapunov optimization theory [20], researchers conducted a few studies on balancing the tradeoff between energy consumption and transmission delay for delay-unbounded mobile applications [10,22,24]. However, none of these work takes the scenarios involve both delay-unbounded and delay-bounded applications into consideration.

2.2. Lyapunov optimization algorithms for stochastic systems

Lyapunov optimization [20] is a recently proposed technique for solving problems of joint system stability and performance optimization on stochastic networks, especially communication and queueing systems. Generally, to minimize the time average objective $\bar{y}$ for a queueing system with the queue backlog vector $\mathbf{\Theta}(t)$, the Lyapunov optimization algorithm is designed to make control decisions that greedily minimize a bound on the following drift-plus-penalty expression in each time slot t:

$$\Delta(\mathbf{\Theta}(t)) + V\mathbb{E}\{y(t)|\mathbf{\Theta}(t)\}$$

where $\Delta(\mathbf{\Theta}(t))$ (Lyapunov drift) represents the congestion state of queue backlog, $y(t)$ denotes the penalty function mapped from $\bar{y}$, and $V \geqslant 0$ represents a design knob of the stability-optimization tradeoff, i.e., how much we emphasize the penalty minimization compared to system stability. Unlike Dynamic Programming [2] and Decision Process [30], Lyapunov optimization does not require knowledge of the statistics of relevant stochastic models, but instead the queue backlog information, to make online control decisions. The two traditional techniques usually suffer from the so-called "curse of dimensionality" problem [20], and result in hard-to-implement systems where significant re-computation is inevitable when statistics change. In contrast, Lyapunov optimization algorithms commonly have a better computational complexity, and are easy to be implemented in practice [22,24]. By now, this new theory has been widely applied in solving many optimization problems on stochastic systems, including workload/resource scheduling among data centers [11,27,32], power/cost management in smart grid [6,14,17], and energy/throughput optimization for wireless systems [10,22,24]. Among them, the work most relevant to ours is that in [18], which can provide worst-case delay guarantees for data traffic sessions in single and multi-hop wireless networks. However, the objective of this work is to maximize a throughput utility, without any consideration on energy consumption of wireless nodes.

3. Problem formulation

Consider a single mobile device that transmits its data to a cloud platform for some specific purpose, e.g., analysis or storage. The whole system operates in discrete time with normalized timeslots $t \in \{0, 1, 2, \ldots\}$. Based on the delay requirements on data transmission, the mobile applications are classified into two types, i.e., delay-bounded and delay-unbounded. Accordingly, the application data is distinguished and processed separately in two different queues Q_{bd} and Q_{ubd}. Let $Q(t) = (Q_{bd}(t), Q_{ubd}(t))$ denote the queue backlog of each queue in time slot t. Depending on the context, the backlog can either take integer units of packets or real units of bits. Let $\mathbf{A}(t) = (A_{bd}(t), A_{ubd}(t))$ be the vector of new arrivals to the above two queues. It is assumed that there exists some A_{max} such that $0 \leqslant A_{bd}(t),\ A_{ubd}(t) \leqslant A_{max}$ for all t. We assume that $\mathbf{A}(t)$ are independent and identically distributed over time slots [20,22].

Under the traffic arrival model above, we focus on the following three important control decisions to be made on the mobile device.

- *Application scheduling*: In each time slot, the first control decision is to determine which queue to serve (i.e., transmit data in this queue). For application scheduling, $\phi(t) \in \{0, 1\}$ denotes the scheduling indicator in time slot t: if $\phi(t) = 0$, then Q_{ubd} will be served; otherwise, Q_{bd} will be served.
- *Interface selection*: The next control decision in time slot t is to determine whether to use any of the available wireless links to transmit data, and if so, which one would be used. Let $\mathcal{L}$ denote the abstract set that defines transmission options, and $l(t) \in \mathcal{L}$ denote the control decision on interface selection. The achievable uplink throughput (also called the service rate [18]) $\mu_l(t) = \hat{\mu}(l(t))$ is stochastic and unpredictable as the actual connectivity depends on the selected link capacity and the current channel state [12,24]. We assume that there exists some finite constant μ_{max} such that $0 \leqslant \mu_l(t) \leqslant \mu_{max}$ for all time under any decision on interface selection. As revealed in [12,24], the energy consumption for data transmission by a mobile device is primarily associated with $l(t)$ and $\mu_l(t)$, which can be modelled as $P(t) = \hat{P}(l(t))$.
- *Packet dropping*: For delay-bounded applications, we define the packet drop decision $d(t)$. It allows data packets already in the queue Q_{bd} to be dropped if their delay is too large. The drop decision is chosen subject to the constraint $0 \leqslant d(t) \leqslant d_{max}$, where d_{max} is a finite value that specified the maximum amount of data allowed to be dropped in one slot. We assume that $d_{max} \geqslant A_{max}$ so that it is always possible to stabilize the queue Q_{bd}, and this can be done with one slot delay if we choose $d(t) = d_{max}$ for all t [19]. However, a penalty of θ is enforced for dropping packets at the network queue, and θ is a constant that satisfies $0 \leqslant \theta \leqslant \infty$. This penalty can represent the additional energy for remediation operations [33] or the degradation level of user experiences [13]. Hence, the dropping penalty occurs in time slot t if there are dropped packets, with the total amount of $\theta d(t)$. Intuitively, a lower energy link can help to save energy (e.g., WiFi vs. LTE [12]), but the low data rates it provides may bring about additional penalty due to packet dropping. On the other hand, data packets in the queue Q_{ubd} are able to tolerant to long service delays and won't be dropped arbitrarily [24].

Based on the system model above, we can capture the following queueing dynamics over time for delay-bounded and delay-unbounded applications on the mobile device:

$$Q_{bd}(t+1) = \max[Q_{bd}(t) - \phi(t)\mu_l(t) - d(t), 0] + A_{bd}(t) \tag{1}$$

$$Q_{ubd}(t+1) = \max[Q_{ubd}(t) - (1-\phi(t))\mu_l(t), 0] + A_{ubd}(t) \tag{2}$$

Accordingly, we can formally define the stability constraint on the queues, which ensures that the average queue length is finite. The queue stability of the whole system can be defined as:

$$\bar{Q} \triangleq \limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\{Q_{bd}(t) + Q_{ubd}(t)\} < \infty \tag{3}$$

Our objective under the above queueing model is to develop joint application scheduling, interface selection and packet dropping policies to minimize a joint utility of network energy cost and packet dropping penalty. The mobile device can serve all arrival traffic in Q_{bd} and Q_{ubd} within the capacity region which is the set of all acceptable arrival rates with guaranteeing queue stabilities. This problem can be formulated into a stochastic optimization problem as below:

$$\begin{aligned} \min \quad & \bar{U} \triangleq \limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\{P(t) + \theta d(t)\} \\ \text{s.t.} \quad & 0 \leqslant A_{bd}(t), \quad A_{ubd}(t) \leqslant A_{max}, \quad \forall t \\ & 0 \leqslant \mu_l(t) \leqslant \mu_{max}, \quad \forall t \\ & \bar{Q} < \infty \\ & \phi(t) \in \{0,1\}, \quad \forall t \\ & l(t) \in \mathcal{L}, \quad \forall t \\ & 0 \leqslant d(t) \leqslant d_{max}, \quad \forall t \end{aligned} \tag{4}$$

To guarantee the worst-case bounded delay for delay-bounded applications, we define a ϵ-persistent service queue [20], being a virtual queue Z with $Z(0) = 0$. The dynamics of this virtual queue is given as follows:

$$Z(t+1) = \max[Z(t) - \phi(t)\mu_l(t) - d(t) + \epsilon 1_{\{Q_{bd}(t)>0\}}, 0] \tag{5}$$

where $\epsilon > 0$ is a pre-specified constant, and $1_{\{Q_{bd}(t)>0\}}$ is an indicator function that is 1 if $Q_{bd}(t) > 0$, and 0 else. The intuition is that Z has the same service process as Q_{bd}, but has an arrival process that adds ϵ whenever the actual queue backlog is nonempty, which ensures that Z grows if there is data traffic in the Q_{bd} queue that has not been serviced for a long time. In this way, the size of the queue Z can provide a bound on the delay of the head-of-line data in the FIFO queue Q_{bd}. If our algorithm can control the system to ensure that $Q_{bd}(t)$ and $Z(t)$ have finite upper bounds, then we can ensure persistent service to the queued data of delay-bounded applications with bounded worst-case delay, as shown in the following lemma.

Lemma 1. *Suppose an algorithm can control the system to ensure that $Q_{bd}(t) \leqslant Q_{bd}^{max}$ and $Z(t) \leqslant Z^{max}$ for all slots $t \in \{0,1,2,\ldots\}$, where Q_{bd}^{max} and Z^{max} are finite bounds on actual and virtual queue backlog. If service and drops are all done in FIFO order, then the worst-case delay of all non-dropped data in queue $Q_{bd}(t)$ is bounded by the constant W_{max} defined below:*

$$W_{max} \triangleq \lceil (Q_{bd}^{max} + Z^{max})/\epsilon \rceil \tag{6}$$

Proof. Fix any slot $t \geqslant 0$. From (1), the earliest time it can depart the queue is time slot $t+1$. We show that all arrivals $A_{bd}(t)$ is either served or dropped on or before time slot $t + W_{max}$. Suppose that this assumption is not true, we shall reach a contradiction. It must be that $Q_{bd}(\varphi) > \phi(\varphi)\mu_l(\varphi) + d(\varphi)$ for all $\varphi \in \{t+1, \ldots, t+W_{max}\}$ (else, the backlog on slot φ would be cleared). Therefore, $1_{\{Q_{bd}(\varphi)>0\}} = 1$, and from (5) we have for all slots $\varphi \in \{t+1, \ldots, t+W_{max}\}$:

$$Z(\varphi+1) = \max[Z(\varphi) - \phi(\varphi)\mu_l(\varphi) - d(\varphi) + \epsilon, 0]$$

In particular, for all slots $\varphi \in \{t+1, \ldots, t+W_{max}\}$:

$$Z(\varphi+1) \geqslant Z(\varphi) - \phi(\varphi)\mu_l(\varphi) - d(\varphi) + \epsilon$$

Summing the above over $\varphi \in \{t+1, \ldots, t+W_{max}\}$ yields:

$$Z(t+W_{max}+1) - Z(t+1) \geqslant -\sum_{\varphi=t+1}^{t+W_{max}} [\phi(\varphi)\mu_l(\varphi) + d(\varphi)] + W_{max}\epsilon$$

Rearranging terms in the above inequality and using the fact that $Z(t+1) \geqslant 0$ and $Z(t+W_{max}+1) \leqslant Z^{max}$ yields:

$$W_{max}\epsilon \leqslant \sum_{\varphi=t+1}^{t+W_{max}} [\phi(\varphi)\mu_l(\varphi) + d(\varphi)] + Z^{max} \tag{7}$$

Note that $A_{bd}(t)$ that arrives in time slot t is placed at the end of the queue Q_{bd} in time slot $t+1$ according to the queue dynamics, and will be served only when all of the backlog $Q_{bd}(t+1)$ has departed. Because $Q_{bd}(t+1) \leqslant Q_{bd}^{max}$ and the service is FIFO, $A_{bd}(t)$ is served on or before time $t+W_{max}$ whenever there are at least Q_{bd}^{max} units of data served during the interval $\varphi \in \{t+1, \ldots, t+W_{max}\}$. Because we have assumed that $A_{bd}(t)$ is not served by time $t+W_{max}$, it must be that $\sum_{\varphi=t+1}^{t+W_{max}}[\phi(\varphi)\mu_l(\varphi)+d(\varphi)] < Q_{bd}(t+1) \leqslant Q_{bd}^{max}$. Combining this and (7) yields:

$$W_{max}\epsilon < Q_{bd}^{max} + Z^{max}$$

This implies $W_{max} < (Q_{bd}^{max}+Z^{max})/\epsilon$, which contradicts the original definition of W_{max} given in the lemma. □

4. Algorithm design

4.1. Lyapunov optimization

Let $\mathbf{\Theta}(t) \triangleq (Q_{bd}(t), Q_{ubd}(t), Z(t))$ be a concatenated vector of all actual and virtual queues, and define a quadratic Lyapunov function [20] as follows:

$$L(\mathbf{\Theta}(t)) \triangleq \frac{1}{2}[Q_{bd}(t)^2 + Q_{ubd}(t)^2 + Z(t)^2] \tag{8}$$

Then, the one-slot conditional Lyapunov drift $\Delta(\mathbf{\Theta}(t))$ is defined as:

$$\Delta(\mathbf{\Theta}(t)) \triangleq \mathbb{E}[L(\mathbf{\Theta}(t+1)) - L(\mathbf{\Theta}(t))|\mathbf{\Theta}(t)] \tag{9}$$

Following the drift-plus-penalty framework in Lyapunov optimization theory, we design the control algorithm to make decisions on $\phi(t),\ l(t)$, and $d(t)$ to minimize an upper bound on the following drift-plus-penalty term in each time slot:

$$\Delta(\mathbf{\Theta}(t)) + V\mathbb{E}\{P(t)+\theta d(t)|\mathbf{\Theta}(t)\} \tag{10}$$

where $V \geqslant 0$ is a non-negative parameter set by system operators to control the tradeoff between utility minimization (i.e., problem (4)) and system stability.

Theorem 1 (*Drift-plus-penalty Bound*). *Under any control algorithm, the drift-plus-penalty expression has the following upper bound for all t, all possible values of* $\mathbf{\Theta}(t)$, *and all parameters* $V \geqslant 0$:

$$\begin{aligned}\Delta(\mathbf{\Theta}(t)) + V\mathbb{E}\{P(t)+\theta d(t)|\mathbf{\Theta}(t)\} \leqslant{}& B + \mathbb{E}\{Z(t)\epsilon + Q_{bd}(t)A_{bd}(t) + Q_{ubd}(t)A_{ubd}(t)|\mathbf{\Theta}(t)\} + \mathbb{E}\{d(t)[V\theta - Q_{bd}(t)\\ &- Z(t)]|\mathbf{\Theta}(t)\} + \mathbb{E}\{VP(t) - Q_{ubd}(t)\mu_l(t) + \phi(t)\mu_l(t)[Q_{ubd}(t) - Q_{bd}(t)\\ &- Z(t)]|\mathbf{\Theta}(t)\}\end{aligned} \tag{11}$$

where $B = \frac{1}{2}[(\mu_{max}+d_{max})^2 + \mu_{max}^2 + 2A_{max}^2] + \frac{1}{2}\max[\epsilon^2, (\mu_{max}+d_{max})^2]$.

Proof. We use the fact that $(\max[Q-b,0]+A)^2 \leqslant Q^2+A^2+b^2+2Q(A-b)$ for any $Q \geqslant 0,\ b \geqslant 0,\ A \geqslant 0$. Squaring the updates for $Q_{bd}(t)$ in (1) and $Q_{ubd}(t)$ in (2), and using $\phi(t) \in \{0,1\},\ A_{bd}(t) \leqslant A_{max},\ A_{ubd}(t) \leqslant A_{max},\ d(t) \leqslant d_{max}$ gives:

$$Q_{bd}(t+1)^2 - Q_{bd}(t)^2 \leqslant A_{max}^2 + (\mu_{max}+d_{max})^2 + 2Q_{bd}(t)[A_{bd}(t) - \phi(t)\mu_l(t) - d(t)]$$
$$Q_{ubd}(t+1)^2 - Q_{ubd}(t)^2 \leqslant A_{max}^2 + \mu_{max}^2 + 2Q_{ubd}(t)[A_{ubd}(t) - (1-\phi(t))\mu_l(t)]$$

Squaring the update for $Z(t)$ in (5), using the fact that $(\max[Q-b+A,0])^2 \leqslant Q^2 + \max(A^2,b^2) + 2Q(A-b)$ for any $Q \geqslant 0, b \geqslant 0,\ A \geqslant 0$, gives:

$$Z(t+1)^2 - Z(t)^2 \leqslant \max[\epsilon^2, (\mu_{max}+d_{max})^2] + 2Z(t)[\epsilon - \phi(t)\mu_l(t) - d(t)]$$

Combining these three bounds together, and taking the expectation with respect to $\mathbf{\Theta}(t)$ on both sides, we arrive at the following one-slot conditional Lyapunov drift:

$$\begin{aligned}\Delta(\mathbf{\Theta}(t)) \leqslant{}& B + \mathbb{E}\{Q_{bd}(t)[A_{bd}(t) - \phi(t)\mu_l(t) - d(t)]|\mathbf{\Theta}(t)\} + \mathbb{E}\{Q_{ubd}(t)[A_{ubd}(t) - (1-\phi(t))\mu_l(t)]|\mathbf{\Theta}(t)\} + \mathbb{E}\{Z(t)[\epsilon\\ &- \phi(t)\mu_l(t) - d(t)]|\mathbf{\Theta}(t)\}\end{aligned}$$

where $B = \frac{1}{2}[(\mu_{max}+d_{max})^2 + \mu_{max}^2 + 2A_{max}^2] + \frac{1}{2}\max[\epsilon^2, (\mu_{max}+d_{max})^2]$.

Now adding to both sides the penalty expression, i.e., the term $V\mathbb{E}\{P(t)+\theta d(t)|\mathbf{\Theta}(t)\}$, we prove the theorem. □

Therefore, rather than directly minimizing the drift-plug-penalty expression in every time slot, we can actually seek to minimize the upper bound given in the right-hand-side of (11).

4.2. OPERA algorithm

OPtimal schEduling for data tRAnsmission (OPERA) Algorithm: In every time slot t, observe the queue states $\mathbf{\Theta}(t)$ and $\mathcal{L}$, and perform the following:

- *Application scheduling*: Choose $\phi(t)$ by:

$$\phi(t) = \begin{cases} 0, & Q_{ubd}(t) > Q_{bd}(t) + Z(t) \\ 1, & Q_{ubd}(t) \leqslant Q_{bd}(t) + Z(t) \end{cases} \tag{12}$$

- *Interface selection*: If $\phi(t) = 0$, then choose $l(t)$ to solve:

$$\begin{aligned} \min_{l(t)} \quad & V\hat{P}(l(t)) - Q_{ubd}(t)\hat{\mu}(l(t)) \\ \text{s.t.} \quad & l(t) \in \mathcal{L} \end{aligned} \tag{13}$$

Else, choose $l(t)$ to solve:

$$\begin{aligned} \min_{l(t)} \quad & V\hat{P}(l(t)) - [Q_{bd}(t) + Z(t)]\hat{\mu}(l(t)) \\ \text{s.t.} \quad & l(t) \in \mathcal{L} \end{aligned} \tag{14}$$

- *Packet dropping*: Choose $d(t)$ by:

$$d(t) = \begin{cases} d_{max}, & \text{if } Q_{bd}(t) + Z(t) > V\theta \\ 0, & \text{if } Q_{bd}(t) + Z(t) \leqslant V\theta \end{cases} \tag{15}$$

- *Queue update*: Update $Q_{bd}(t),\ Q_{ubd}(t)$ and $Z(t)$ according to the dynamics (1), (2) and (5), respectively.

4.3. Performance analysis

In this subsection, we analyze the performance bound of the OPERA algorithm.

Theorem 2 (*Algorithm Performance*). *Implementing the OPERA algorithm in every time slot for any fixed control parameter $V \geqslant 0$, yields the following performance bounds:*

(1) The queues $Q_{bd}(t)$ and $Z(t)$ are upper bounded by constants Q_{bd}^{max} and Z^{max} that are defined respectively as follows:

$$Q_{bd}^{max} \triangleq V\theta + A_{max} \tag{16}$$

$$Z^{max} \triangleq V\theta + \epsilon \tag{17}$$

(2) The worst case delay for all non-dropped data in queue $Q_{bd}(t)$ is:

$$W_{max} \triangleq \lceil (2V\theta + A_{max} + \epsilon)/\epsilon \rceil \tag{18}$$

(3) For any control variable $V > 0$, OPERA can stabilize the system, with a resulted time average utility and queue backlog satisfying the following inequalities:

$$\bar{Q} \leqslant \frac{B + VU^{opt}}{\eta} \tag{19}$$

$$\bar{U} \leqslant U^{opt} + \frac{B}{V} \tag{20}$$

where $\eta > 0$ is a constant, and U^{opt} is a theoretical lower bound on the time average utility.

Proof.

(1) We first show that $Q_{bd}(t) \leqslant V\theta + A_{max}$ for all t. Suppose it holds for time slot t. We show it also holds for slot $t+1$. Consider the case when $Q_{bd}(t) \leqslant V\theta$. Then $Q(t+1) \leqslant V\theta + A_{max}$, because the queue can increase by at most A_{max} in any slot according to (1). Thus, the result holds in this case. On the other hand, when $V\theta < Q(t) \leqslant V\theta + A_{max}$, we have $Q_{bd}(t) + Z(t) \geqslant Q_{bd}(t) > V\theta$ and hence the algorithm will choose $d(t) = d_{max}$ according to (15). If $Q_{bd}(t) - \phi(t)\mu_l(t) - d(t) > 0$, then in time slot t at least d_{max} will be dropped. Since $A_{max} \leqslant d_{max}$, the queue cannot

increase on the next slot, i.e., $Q_{bd}(t+1) \leqslant Q_{bd}(t) \leqslant V\theta + A_{max}$. Otherwise, if $Q_{bd}(t) - \phi(t)\mu_l(t) - d(t) \leqslant 0$, then by (1) we have $Q_{bd}(t+1) = A(t) \leqslant A_{max} \leqslant V\theta + A_{max}$. The proof that $Z(t) \leqslant V\theta + \epsilon$ for all t is proven similarly and omitted for brevity.

(2) This follows immediately from Lemma (1) together with part (1).

(3) Because, every time slot t, our implementation seeks to minimize the right-hand-side of the drift-plus-penalty expression in (11):

$$\begin{aligned}\Delta(\mathbf{\Theta}(t)) + V\mathbb{E}\{P(t) + \theta d(t)|\mathbf{\Theta}(t)\} \leqslant B &+ V\mathbb{E}\{\hat{P}(l^*(t)) + \theta d^*(t)|\mathbf{\Theta}(t)\} + \mathbb{E}\{Q_{bd}(t)[A_{bd}(t) - \phi(t)\hat{\mu}(l^*(t)) - d^*(t)]|\mathbf{\Theta}(t)\}\\ &+ \mathbb{E}\{Q_{ubd}(t)[A_{ubd}(t) - (1-\phi^*(t))\hat{\mu}(l^*(t))]|\mathbf{\Theta}(t)\} + \mathbb{E}\{Z(t)[\epsilon - \phi^*(t)\hat{\mu}(l^*(t))\\ &- d^*(t)]|\mathbf{\Theta}(t)\}\end{aligned} \tag{21}$$

where * represents any alternative policy other than the optimal one. According to Theorem 4.8 in [20], the resulting values of $\phi^*(t),\ l^*(t),\ d^*(t)$ are independent of the current queue backlogs $\mathbf{\Theta}(t)$. According to Caratheodory's theorem [32], there exists at least one randomized stationary control policy * and $\eta > 0$ [20] such that

$$\mathbb{E}\{\hat{P}(l^*(t)) + \theta d^*(t)|\mathbf{\Theta}(t)\} = \mathbb{E}\{\hat{P}(l^*(t)) + \theta d^*(t)\} = U^{opt} \tag{22}$$

$$\mathbb{E}\{A_{bd}(t) - \phi(t)\hat{\mu}(l^*(t)) - d^*(t)|\mathbf{\Theta}(t)\} = \mathbb{E}\{A_{bd}(t) - \phi(t)\hat{\mu}(l^*(t)) - d^*(t)\} \leqslant -\eta \tag{23}$$

$$\mathbb{E}\{A_{ubd}(t) - (1-\phi^*(t))\hat{\mu}(l^*(t))|\mathbf{\Theta}(t)\} = \mathbb{E}\{A_{ubd}(t) - (1-\phi^*(t))\hat{\mu}(l^*(t))\} \leqslant -\eta \tag{24}$$

$$\mathbb{E}\{\epsilon - \phi^*(t)\hat{\mu}(l^*(t)) - d^*(t)|\mathbf{\Theta}(t)\} = \mathbb{E}\{\epsilon - \phi^*(t)\hat{\mu}(l^*(t)) - d^*(t)\} \leqslant -\eta \tag{25}$$

Plugging the above (22)–(25) into the right-hand-side of (21) yields:

$$\Delta(\mathbf{\Theta}(t)) + V\mathbb{E}\{P(t) + \theta d(t)|\mathbf{\Theta}(t)\} \leqslant B + VU^{opt} - \eta[Q_{bd}(t) + Q_{ubd}(t) + Z(t)]$$

This is in the exact form for application of the Lyapunov Optimization Theorem (Theorem 4.2 in [20]). Hence, all queues are mean rate stable. Taking a conditional expectation over $\mathbf{\Theta}(t)$ for the above, and using the iterative expectation law, we have:

$$\mathbb{E}\{L(\mathbf{\Theta}(t+1)) - L(\mathbf{\Theta}(t))\} + V\mathbb{E}\{P(t) + \theta d(t)\} \leqslant B + VU^{opt} - \eta\mathbb{E}\{Q_{bd}(t) + Q_{ubd}(t) + Z(t)\} \tag{26}$$

Summing the above over all time slots $t \in \{0, 1, \ldots, T-1\}$, rearranging the terms, using the fact that $L(\mathbf{\Theta}(t)) \geqslant 0$ and $L(\mathbf{\Theta}(t)) = 0$ for all t, and dividing both sides by T, we have:

$$\frac{\eta}{T}\sum_{t=0}^{T-1}\mathbb{E}\{Q_{bd}(t) + Q_{ubd}(t) + Z(t)\} \leqslant B + VU^{opt}$$

Taking a lim sup as $T \to \infty$, using the fact that $\mathbb{E}\{Z(t)\} \geqslant 0$, we can prove (19):

$$\bar{Q} \leqslant \limsup_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\{Q_{bd}(t) + Q_{ubd}(t) + Z(t)\} \leqslant \frac{B + VU^{opt}}{\eta}$$

To prove (20), using (26), we have:

$$V\mathbb{E}\{P(t) + \theta d(t)\} \leqslant VU^{opt} + B$$

Summing the above over all time slots $t \in \{0, 1, \ldots, T-1\}$, dividing both sides by TV, and taking a lim sup as $T \to \infty$, we can prove (20):

$$\bar{U} \triangleq \limsup_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\{P(t) + \theta d(t)\} \leqslant U^{opt} + \frac{B}{V} \qquad \square$$

5. Performance evaluation

In this section, we evaluate the proposed OPERA algorithm using datasets from real-world measurements on wireless uplink throughput and corresponding energy consumption of mobile devices.

5.1. Simulation setup

We consider a mobile device equipped with three state-of-the-art wireless interfaces, i.e., WiFi, 3G and LTE [10]. Then, $\mathcal{L} = \{\text{WiFi, 3G, LTE, Idle}\}$. The uplink throughput traces we use in the simulations are the UMICH measurement datasets from 4GTest Project [12]. According to observed bandwidth statistics in the traces and an existing study [15], we set the length of one time slot $\chi = 10$ seconds. A portion of twenty-minutes throughput variation for different interfaces is shown in Fig. 1. Meanwhile, we also use the power model derived in [12], i.e., $P(t) = [\alpha\mu_l(t) + \beta] * \chi$. Specifically, the parameters for energy consumption of wireless interfaces are listed in Table 1. For the data traffic, we consider a delay-bounded application and a delay-unbounded application, and assume that their data arrivals both follow the Poisson Process with $\mathbb{E}\{A_{bd}(t)\} = \mathbb{E}\{A_{ubd}(t)\} = 4$ Mb [15,24]. (Note that our algorithm does not have any special requirement on this traffic pattern.) Besides, it is assumed that $d_{max} = A_{max}$ and $\epsilon = 10$.

To fully investigate the OPERA performance, we compare it with an online algorithm called "Fastest". It always chooses the interface currently providing the highest throughput among the three, and prefer to serve queue Q_{bd} than Q_{ubd} to prevent potential dropping penalty as long as Q_{ubd} can be kept stable. According to Theorem 2.4 in [20], we use a simple control policy to enforce that $\phi(t) = 0$ to serve queue Q_{ubd} if the average volume of ingress traffic is larger than that of egress traffic in time slot t [10]. Intuitively, this algorithm should have good delay characteristics for the delay-bounded applications.

5.2. Results and analysis

In the following, we conduct analysis on critical factors, i.e., V, T and θ, to characterize their impacts on the performance of OPERA [10,32]. Note that the average service delay metric in the results is calculated by weighted averaging the queueing delay for transmitted data, and the weight coefficient is the serviced amount [10].

5.2.1. Impact of V

We first fix $T = 1000$ time slots and $\theta = 1000$, and then run simulation experiments with different V values. The simulation results are shown in Fig. 2. We have several observations concerning these results. Firstly, as the value of V increases from 0 to 0.1, the time-average utility achieved by OPERA drops from 48.4 to 7.61, while the time-average queue backlog grows from 7.15 to 34.6. Note that the utility falls quickly at the beginning and then tends to descend slowly (Fig. 2(a)), while the average delay almost increases linearly (Fig. 2(b)). This quantitatively confirms the $[\mathcal{O}(1/V), \mathcal{O}(V)]$ utility-stability tradeoff in part (3) of Theorem 2. Although the Fastest algorithm indeed has lower delay (Fig. 2(c)), OPERA can reduce significant cost with acceptable delay by choosing an appropriate value of V, e.g., $V \geqslant 0.01$. As a result, it is possible for OPERA to balance the tradeoff between the energy cost requirement of energy-constrained device and the delay bound requirement of delay-tolerant applications. Secondly, as shown in Fig. 2(c), the average service delay for data packets in Q_{bd} increases linearly with V, and the maximum service delay never exceeds the theoretical worst-case delay bound. These results in Fig. 2(c) are consistent with (18) in part (2) of Theorem 2. Thirdly, as the value of V increases from 0 to 0.1, OPERA is more inclined to serve Q_{bd} than Q_{ubd} (Fig. 2(d)), so as to reduce excessive penalties on packet dropping and minimize the joint utility. Fourthly, Fastest preferentially chooses to serve Q_{bd} with the highest-throughput interface (Fig. 2(d)), as long as Q_{ubd} is stable. Such a control policy is able to actively avoid packet dropping and minimize penalty loss, but will bring about significant amount of energy consumption (Fig. 2(a)). Due to the same reason, queue Q_{ubd} is rarely served, resulting in very high time-average backlogs (Fig. 2(b)). As shown in Fig. 2(c), the average service delay under the Fastest algorithm is slightly larger than one time slot. That's because most of the traffic in Q_{bd} will be served in the next time slot upon their arrivals, unless queue Q_{ubd} is currently found congested and has to be served immediately.

Moreover, we compare the theoretical and the experimental upper bound of queue backlogs of Q_{bd} under this setting. The results are shown in Table 2. In our experiments, the queue backlogs are smaller than the mathematical bounds, especially when V is relatively larger. These results are consistent with (16) in part (1) of Theorem 2.

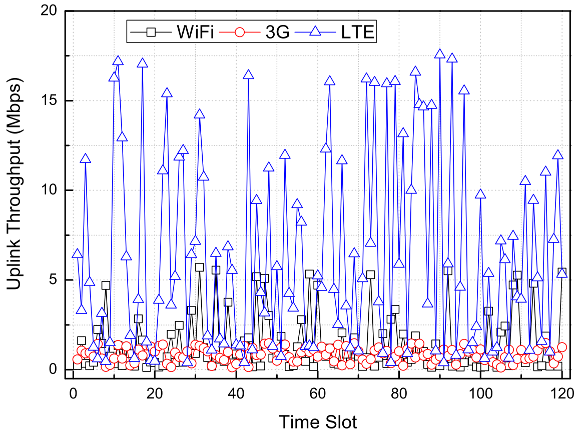
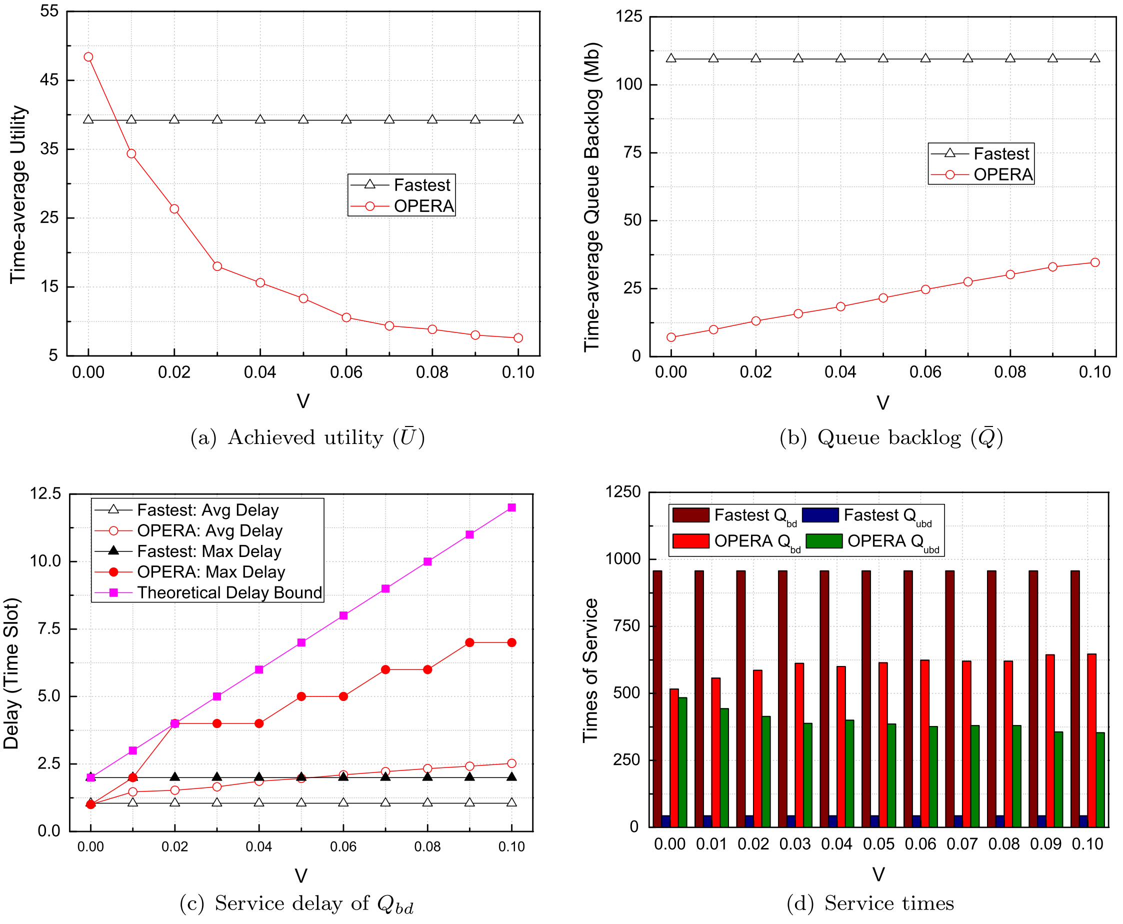


Fig. 1. Twenty-minutes trace of uplink throughput for different wireless interfaces.

Table 1
Power parameters for different wireless interfaces.

	α (mW/Mbps)	β (mW)
WiFi	283.17	132.86
3G	868.98	817.88
LTE	438.39	1288.04



(a) Achieved utility ($\bar{U}$)

(b) Queue backlog ($\bar{Q}$)

(c) Service delay of Q_{bd}

(d) Service times

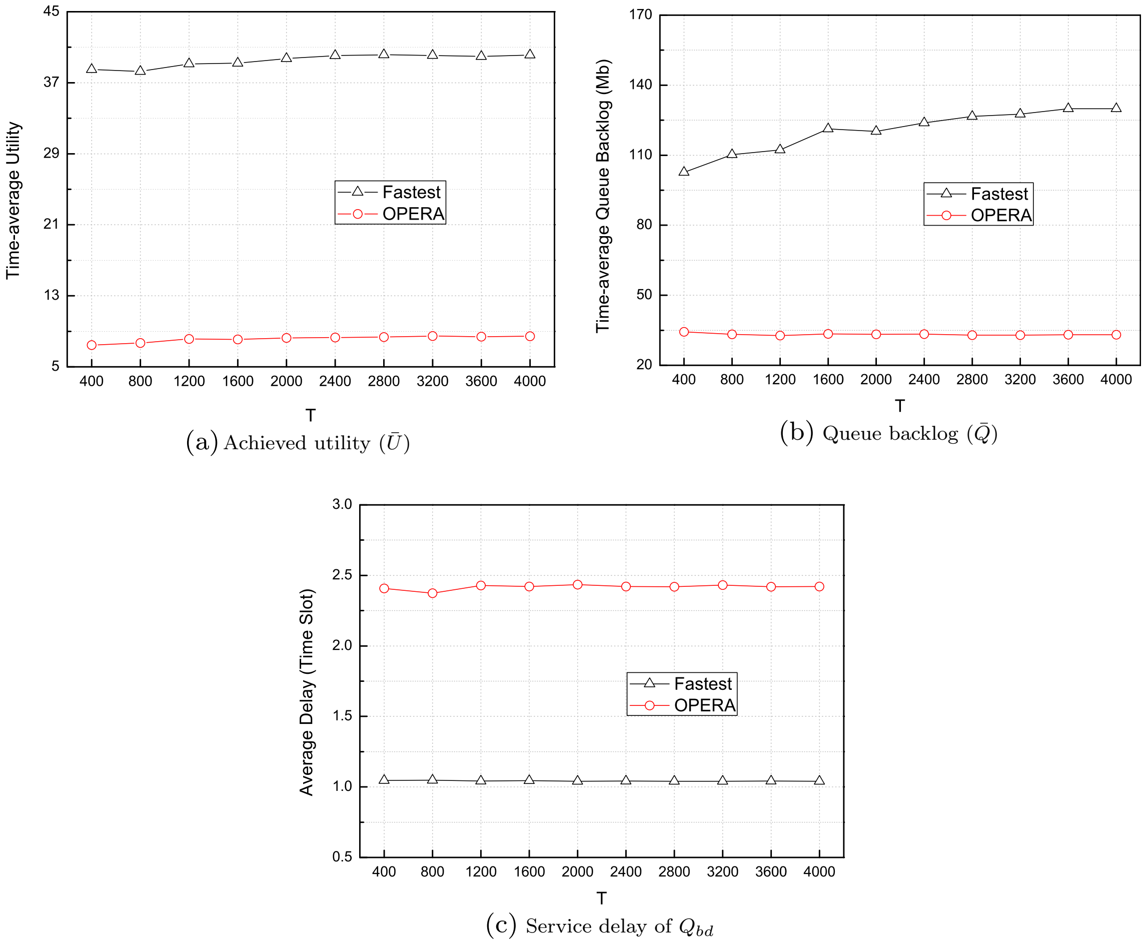
Fig. 2. OPERA performance under different V value.

Table 2
Comparison on theoretical (T) and experimental (E) upper bounds of Q_{bd} under different V value.

V	0	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
T	8	18	28	38	48	58	68	78	88	98	108
E	5	9	12	14	14	16	18	19	18	20	20

5.2.2. Impact of T

We fix $V = 0.09$ and $\theta = 1000$, and then vary T from 400 to 4000 time slots, which is a sufficient range for exploring the characteristics of OPERA. Corresponding results of the two algorithms are shown in Fig. 3. These results are consistent with those in Fig. 2(a)–(c) (when $V = 0.09$). It is obvious that changing T has relatively small impacts on the system stability. The fluctuations on the time-average utility, time-average queue backlog, and queue delay of Q_{bd} are $[-8.85\%, +3.91\%]$, $[-1.63\%, +3.25\%]$, and $[-1.84\%, +0.67\%]$, respectively, for OPERA, and are $[-3.16\%, +1.59\%]$, $[-14.76\%, +7.86\%]$, and $[-0.27\%, 0.53\%]$, respectively, for Fastest. The observations above confirm that the usability and the effectiveness of stability guarantee mechanisms employed by OPERA and Fastest.



(a) Achieved utility ($\bar{U}$)

(b) Queue backlog ($\bar{Q}$)

(c) Service delay of Q_{bd}

Fig. 3. OPERA performance under different T value.

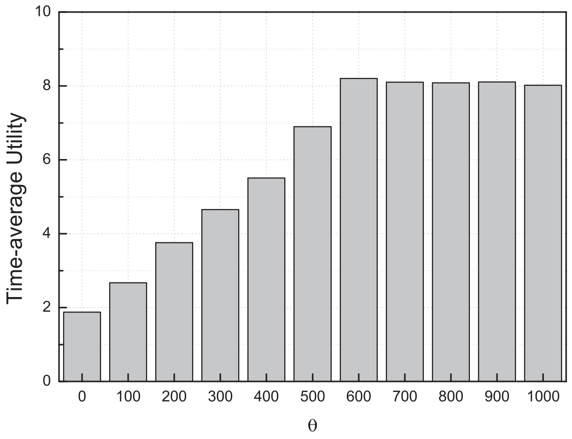


Fig. 4. OPERA performance on achieved utility ($\bar{U}$) under different θ value.

5.2.3. Impact of θ

We fix $V = 0.09$ and $T = 1000$ time slots, and then run simulation experiments with different θ values. Fig. 4 plots the simulation results under OPERA with various values of parameter θ (the results under Fastest are omitted since it incurs no packet drop and no penalty loss, and $\bar{U}$ are always equal to 39.21 as those in Fig. 2(a)). When θ is relatively small (e.g, $\theta = 0$), packet dropping leads to no or neglectable penalties, and thus will occur very frequently. Meanwhile, queue Q_{bd}

and Q_{ubd} have almost equal opportunities to be served. As θ grows from 0 to 600, OPERA will be more inclined to serve Q_{bd} using the wireless interface with higher throughput capability, so as to reduce packet dropping and penalty loss based on the rule defined in (15). That's why the time-average utility $\bar{U}$ increases with θ. However, when $\theta \geqslant 700$, there will be few or even no packet dropping. Therefore, the time-average utility gradually converges to the optimal value 8.02 under this setting.

6. Conclusion

In this paper, we investigated the network energy consumption problem widely observed in mobile cloud computing, by introducing a new transmission scheduling algorithm. We formulated the transmission scheduling problem as a stochastic optimization problem, whose objective is to minimize the joint utility of network energy cost and packet dropping penalty while satisfying different application requirements on service delay. Leveraging the Lyapunov optimization theory, we derived a new online transmission scheduling algorithm OPERA, which adaptively makes control decisions on application scheduling, interface selection and packet dropping in response to the stochastic traffic arrivals and the time-varying link qualities. This algorithm is easy to be implemented online, and can give analytical bound on the performance. Numerical evaluations based on real-world traces indicate the effectiveness of our algorithm and the correctness of our analysis. As future work, we will extend the system model to consider fine-grained delay requirements of different delay-bounded applications [18,26]. Another important work is to further evaluate OPERA using a prototype implement on the modern smartphone platform [22,24].

Acknowledgements

We would like to thank Dr. Junxian Huang (University of Michigan) for providing us the UMICH dataset from 4GTest measurement [12]. This work was supported in part by the National Science Foundation of China (NSFC) under Grants 61202430, 61202393 and 61401516, the State Key Lab of Astronautical Dynamics of China under Grant 2014ADL-DW0401, and the China Scholarship Council under Grant 201308110150.

References

[1] S. Abolfazli, Z. Sanaei, E. Ahmed, A. Gani, R. Buyya, Cloud-based augmentation for mobile devices: motivation, taxonomies, and open challenges, IEEE Commun. Surv. Tutorials 16 (1) (2014) 337–368.
[2] T. Amin, I. Chikalov, M. Moshkov, B. Zielosko, Dynamic programming approach to optimization of approximate decision rules, Inf. Sci. 221 (0) (2013) 403–418.
[3] N. Balasubramanian, A. Balasubramanian, A. Venkataramani, Energy consumption in mobile phones: a measurement study and implications for network applications, in: Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement Conference, New York, NY, USA, 2009.
[4] W. Cai, V. Leung, M. Chen, Next generation mobile cloud gaming, in: 2013 IEEE 7th International Symposium on Service Oriented System Engineering (SOSE), 2013.
[5] J. Chen, J. Li, T. Lai, Trapping mobile targets in wireless sensor networks: an energy-efficient perspective, IEEE Trans. Veh. Technol. 62 (7) (2013) 3287–3300.
[6] S. Chen, N. Shroff, P. Sinha, Heterogeneous delay tolerant task scheduling and energy management in the smart grid with renewable energy, IEEE J. Sel. Areas Commun. 31 (7) (2013) 1258–1267.
[7] Cisco, Cisco Visual Networking Index: Forecast and Methodology, 2013-2018, Cisco Systems Inc., 2014. <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/>.
[8] R. Deng, Z. Yang, J. Chen, M.-Y. Chow, Load scheduling with price uncertainty and temporally-coupled constraints in smart grids, IEEE Trans. Power Syst. 29 (6) (2014) 2823–2834.
[9] J. Fan, J. Chen, Y. Du, W. Gao, J. Wu, Y. Sun, Geocommunity-based broadcasting for data dissemination in mobile social networks, IEEE Trans. Parallel Distrib. Syst. 24 (4) (2013) 734–743.
[10] W. Fang, Y. Li, H. Zhang, N. Xiong, J. Lai, A.V. Vasilakos, On the throughput-energy tradeoff for data transmission between cloud and mobile devices, Inf. Sci. 283 (0) (2014) 79–93.
[11] Y. Guo, Y. Gong, Y. Fang, P. Khargonekar, X. Geng, Energy and network aware workload management for sustainable data centers with thermal storage, IEEE Trans. Parallel Distrib. Syst. 25 (8) (2014) 2030–2042.
[12] J. Huang, F. Qian, A. Gerber, Z.M. Mao, S. Sen, O. Spatscheck, A close examination of performance and power characteristics of 4g lte networks, in: Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services, MobiSys '12, ACM, New York, NY, USA, 2012.
[13] L. Huang, Optimal sleep-wake scheduling for energy harvesting smart mobile devices, in: 2013 11th International Symposium on Modeling Optimization in Mobile, Ad Hoc Wireless Networks (WiOpt), 2013.
[14] L. Huang, J. Walrand, K. Ramchandran, Optimal demand response with energy storage management, in: 2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm), 2012.
[15] J. Kwak, O. Choi, S. Chong, P. Mohapatra, Dynamic speed scaling for energy minimization in delay-tolerant smartphone applications, in: INFOCOM, 2014 Proceedings IEEE, 2014.
[16] F. Liu, P. Shu, H. Jin, L. Ding, J. Yu, D. Niu, B. Li, Gearing resource-poor mobile devices with powerful clouds: architectures, challenges, and applications, IEEE Wirel. Commun. 20 (3) (2013) 14–22.
[17] A.H. Mahmud, S. Ren, Online capacity provisioning for carbon-neutral data center with demand-responsive electricity prices, SIGMETRICS Perform. Eval. Rev. 41 (2) (2013) 26–37.
[18] M. Neely, Opportunistic scheduling with worst case delay guarantees in single and multi-hop networks, in: INFOCOM, 2011 Proceedings IEEE, 2011.
[19] M. Neely, A. Tehrani, A. Dimakis, Efficient algorithms for renewable energy allocation to delay tolerant consumers, in: 2010 First IEEE International Conference on Smart Grid Communications (SmartGridComm), 2010.
[20] M.J. Neely, Stochasitic Network Optimization with Application to Communication and Queueing Systems, Morgan & Claypool, 2010.
[21] T. Pering, Y. Agarwal, R. Gupta, R. Want, Coolspots: reducing the power consumption of wireless mobile devices with multiple radio interfaces, in: Proceedings of the 4th International Conference on Mobile Systems, Applications and Services, New York, NY, USA, 2006.

[22] M.-R. Ra, J. Paek, A.B. Sharma, R. Govindan, M.H. Krieger, M.J. Neely, Energy-delay tradeoffs in smartphone applications, in: Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services, New York, NY, USA, 2010.

[23] A. Rahmati, L. Zhong, Context-for-wireless: context-sensitive energy-efficient wireless data transfer, in: Proceedings of the 5th International Conference on Mobile Systems, Applications and Services, New York, NY, USA, 2007.

[24] P. Shu, F. Liu, H. Jin, M. Chen, F. Wen, Y. Qu, B. Li, etime: Energy-efficient transmission between cloud and mobile devices, in: Proceedings IEEE INFOCOM 2013, Turin, Italy, 2013.

[25] M. Stemm, R.H. Katz, Measuring and reducing energy consumption of network interfaces in hand-held devices, IEICE Trans. Commun. E80-B (8) (1997) 1125–1131.

[26] Y.-C. Tian, X. Jiang, D. Levy, A. Agrawala, Local adjustment and global adaptation of control periods for QoC management of control systems, IEEE Trans. Control Syst. Technol. 20 (3) (2012) 846–854.

[27] R. Urgaonkar, B. Urgaonkar, M.J. Neely, A. Sivasubramaniam, Optimal power cost management using stored energy in data centers, in: Proceedings of the ACM SIGMETRICS Joint International Conference on Measurement and Modeling of Computer Systems, New York, NY, USA, 2011.

[28] N. Vallina-Rodriguez, P. Hui, J. Crowcroft, A. Rice, Exhausting battery statistics: Understanding the energy demands on mobile handsets, in: Proceedings of the Second ACM SIGCOMM Workshop on Networking, Systems, and Applications on Mobile Handhelds, MobiHeld '10, ACM, New York, NY, USA, 2010.

[29] L. Xiang, S. Ye, Y. Feng, B. Li, B. Li, Ready, set, go: Coalesced offloading from mobile devices to the cloud, in: INFOCOM, 2014 Proceedings IEEE, 2014.

[30] X. Xu, L. Zuo, Z. Huang, Reinforcement learning algorithms with function approximation: recent advances and applications, Inf. Sci. 261 (0) (2014) 1–31.

[31] D. Yang, G. Xue, X. Fang, J. Tang, Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing, in: Proceedings of the 18th Annual International Conference on Mobile Computing and Networking, Mobicom '12, ACM, New York, NY, USA, 2012.

[32] Y. Yao, L. Huang, A. Sharma, L. Golubchik, M. Neely, Power cost reduction in distributed data centers: a two-time-scale approach for delay tolerant workloads, IEEE Trans. Parallel Distrib. Syst. 25 (1) (2014) 200–211.

[33] J. Zhao, H. Li, C. Wu, Z. Li, Z. Zhang, F. Lau, Dynamic pricing and profit maximization for the cloud with geo-distributed data centers, in: INFOCOM, 2014 Proceedings IEEE, 2014.

[34] Y. Zhou, Y. Zhang, H. Liu, N. Xiong, A.V. Vasilakos, A bare-metal and asymmetric partitioning approach to client virtualization, IEEE Trans. Serv. Comput. 7 (1) (2014) 40–53.