Demand Shaping in Cellular Networks

Xinyang Zhou Lijun Chen

***Abstract*— Demand shaping is a promising way to mitigate the wireless cellular capacity shortfall in the presence of ever-increasing wireless data demand. In this paper, we formulate demand shaping as an optimization problem that minimizes the variation in aggregate traffic. We design a distributed and randomized offline demand shaping algorithm under complete traffic information and prove its almost surely convergence. We further consider a more realistic setting where the traffic information is incomplete but future traffic can be predicted to a certain accuracy. We design an online demand shaping algorithm that updates the schedules of deferrable applications each time when new information and updated prediction are available, based on solving at each timeslot an optimization problem over a shrinking horizon from the current time to the end of the day. We compare the performance of the online algorithm against the optimal offline algorithm, and provide numerical examples.**

***Index Terms*— Demand shaping, offline algorithm, online algorithm, supermartingale, deferrable applications, cellular networks.**

I. INTRODUCTION

The recent decade has witnessed rapid increase in demand on wireless data, driven by the proliferation of smartphones, tablets, and laptops with mobile broadband cards. The global mobile traffic in 2012 has reached 10,620 petabytes, almost 12 times greater than the global Internet traffic of 900 petabytes in 2000; yet, this number is expected to increase at a compound annual growth rate of 66%, i.e., a 13-fold growth, from 2012 to 2017 [11]. However, despite frequent upgrades of cellular networks from 2G to 3G and to 4G and beyond, wireless service providers fall short of keeping up with this increasing wireless data demand, which leads to congestion in the network and degraded quality of service (QoS) for the end users.

The capacity shortfall can be mitigated by allocating more wireless spectrum and deploying more wireless infrastructures including more and smaller cells and offload to WiFi networks, etc. However, spectrum allocation and infrastructure upgrading are not only costly but also time-consuming. A promising alternative is to improve spectrum and infrastructure efficiency through managing wireless data traffic (i.e., demand). Notice that wireless traffic or demand usually fluctuates with a large peak-to-valley ratio throughout a day; e.g., the traffic in peak hours can be as much as 10 times more than that in off-peak hours [10], and see also Fig. 1 for a trace of smartphone web browsing over a day. However, wireless capacity needs to be provisioned to meet the peak demand rather than the average. This means that the cellular network is stressed in peak hours while underutilized at other times. If the demand can be shaped to reduce the peak and smooth the variation, not only can more traffic be accommodated under limited capacity constraints, but also additional spectrum allocation and infrastructure upgrades can be slowed down, which greatly improves wireless network efficiency and yields huge savings.

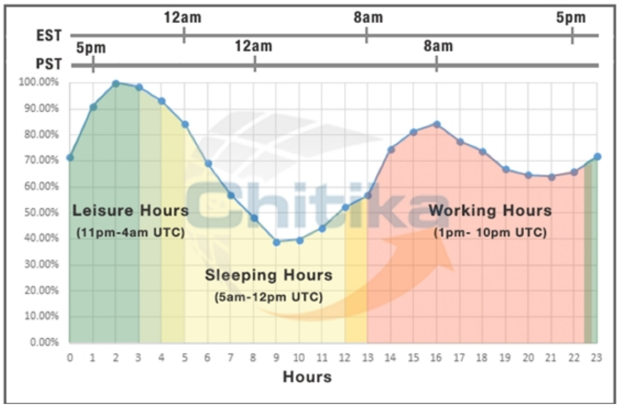

Fig. 1. North America smartphone web browsing activity by time of day [12].

In this paper, we focus on designing demand shaping algorithms for cellular networks. We divide wireless traffic into two categories: non-deferrable traffic and deferrable traffic. Non-deferrable traffic refers to the traffic of applications such as online gaming that have no or low delay tolerance, and constitute the base traffic that cannot be shaped. Deferrable traffic refers to the traffic of applications such as file downloading that are flexible in time and only require being served by a designated deadline. Deferrable applications are further divided into two major types: continuous-rate interruptible applications such as file downloading that allow any data rates, and discrete-rate noninterruptible applications such as online movie watching that usually require certain constant data rate and should not be interrupted once they are started. We seek to schedule the deferrable applications to flatten the aggregate traffic profile over a day.

Specifically, we formulate demand shaping as an optimization problem that minimizes the (time) variation in aggregate traffic subject to the specification on each deferrable application. We first assume complete traffic information and design an offline demand shaping algorithm. There are two challenging issues in the offline algorithm design. First, the resulting optimization problem is non-convex because of discrete-rate noninterruptible applications. We instead solve its convex relaxation and design a randomized scheme based on the solution for the relaxed problem. Second, demand shaping involves potentially a huge number of applications and users. A centralized algorithm is not scalable. We instead design

X. Zhou and L. Chen are with College of Engineering and Applied Science, University of Colorado, Boulder, CO 80309, USA (emails: {xinyang.zhou, lijun.chen}@colorado.edu).

an iterative and distributed algorithm based on the steepest descent method. We establish the almost surely convergence of the algorithm based on supermartingale theory.

We then consider a more realistic setting with incomplete information where we can only predict future traffic to a certain accuracy, and design an online and distributed demand shaping algorithm that updates the schedules of deferrable applications each time when new information and updated prediction are available, based on the above offline algorithm for an optimization problem over a shrinking horizon from the current time to the end of the day. We compare the performance of the online algorithm against the optimal offline algorithm, and provide numerical examples.

Demand shaping in cellular networks is similar to demand response in power networks, in terms of design objectives, problem formulations, and the associated algorithmic challenges. Indeed, we borrow insights from demand response in power networks, see, e.g., [6]–[8], [15]. In particular, our online demand shaping algorithm is motivated by the solution approach in [8], and mathematically can be seen as its extension to include discrete decision variables.

Remarks: In this paper we focus on designing demand shaping algorithms based on a general and simplified system model. We do not investigate the important practical issues such as the timescale and granularity at which we schedule and reschedule the deferrable applications. We plan to develop a platform to enable automatic demand shaping in the future, and will investigate various practical issues then. Also, demand shaping involves not only the design of control algorithms but also the design of right mechanisms to incentivize the users to move out of their "comfortable zone" in wireless applications and data usage. Incentive design for demand shaping is currently an active research area; see, e.g., the smart data pricing [10], [17] and the references therein.

Remarks: Some discussion on the practicality of demand shaping is in place. People tend to use mobile data services whenever they want, regardless of whether it is at peak time or valley time for the cellular network. However, a survey [16] conducted in India and USA in 2012 shows that, given proper monetary incentive, many people are willing to postpone their mobile data usage, with acceptable postponement varying from minutes to hours, depending on different types of services and different individual preferences [10]. For example, wireless service providers can motivate the users to shift their demand by implementing the time-dependent pricing (TDP) strategy. TDP is now applied as a simple two-period plan by many wireless service providers around the world, in voice services and data services; e.g., Verizon [4] and Sprint [2] in the US have "happy hours" in the night and weekend for voice service, TelCom [3] in South Africa has "Night Surfer" plans giving free data from 11pm to 5am, and Airtel [1] in India provides unlimited data in the night. More refined TDP strategies can be applied, to maximize benefits for both wireless service providers and users, by dynamically adjusting prices according to the data usage of the current time and predicted future. For instance, Ha *et al* [10] have been working on a TDP based application named TUBE. Trials in cooperation with local wireless service provider shows its effectiveness in shaping the traffic profile [13].

II. System Model and Problem Formulation

Consider a wireless cellular network that serves users for different applications such as web browsing, file sharing, and real-time entertainment. The applications can be broadly divided into two categories: deferrable and non-deferrable. The deferrable applications (DAs) refer to those that are flexible in the starting time and/or data rate, while the non-deferrable applications refer to those that should be served immediately and often have stringent rate requirement.

Our goal is to schedule the DAs so as to flatten the traffic profile over a day, subject to the time constraints and rate constraints of each application. We use a discrete-time model where a day is divided into T timeslots of equal duration, indexed by $t \in \mathcal{T} = \{1, 2, \cdots, T\}$. The duration of a timeslot can be, e.g., 30 minutes, corresponding to the time resolution at which the scheduling decisions are made.

A. Non-deferrable applications

Non-deferrable applications include web browsing, online gaming, and real-time chatting, etc. The latency tolerated by these applications vary from hundreds of milliseconds to seconds. Since they should be served immediately upon request, the traffic from those applications is inelastic and constitutes the base traffic that cannot be shaped. Denote the base traffic profile by $b = \{b(t); t \in \mathcal{T}\}$. As we can only predict the base traffic to a certain accuracy, we model it as a random variable with mean $\bar{b} = \{\bar{b}(t); t \in \mathcal{T}\}$ and random derivation $\delta b = \{\delta b(t); t \in \mathcal{T}\}$ from the mean, i.e., $b = \bar{b} + \delta b$. We assume that $\delta b(t)$ has a mean of 0 and variance of $\delta^2(t)$ respectively, and may be temporally correlated. We further assume that as time goes by, better prediction of base traffic is possible, modeled by a time-dependent deviation from the mean, i.e., the base traffic at time $\tau \in \mathcal{T}$ is predicted at time t as

$$b_t(\tau) = \bar{b}(\tau) + \delta b_t(\tau), \tag{1}$$

with $\delta b_t(\tau)$ having a decreasing variance $\delta_t^2(\tau)$ as time t goes by. The parameters $\bar{b}$ and δ_t will be specified exogenously, and can be estimated from the historical traffic records.

B. Deferrable applications

Assume that there are N deferrable applications (DAs), indexed by $n \in \mathcal{N} = \{1, \cdots, N\}$. Each application n is characterized by an arrival time t_n^a when it is requested or after which it can be started, a deadline t_n^d by which it must be done, and certain requirement or constraint on data rate $p_n = \{p_n(t); t \in \mathcal{T}\}$. Let P_n denote the total traffic required for DA n, i.e., $\sum_{t \in \mathcal{T}} p_n(t) = P_n$. We can classify DAs into two main categories: *continuous-rate interruptible applications* that allow variable data rate between certain upper and lower bounds and can be interrupted and resumed at any times before the deadline, and *discrete-rate noninterruptible applications* that require certain (roughly) constant data rate and cannot be interrupted once they are started.[1] For example, file-sharing is usually interruptible and allows any continuous data rates; while online streaming is usually non-interruptible and runs at a constant, discrete data rate once it is started.

We assume that there are N' continuous-rate interruptible DAs, denoted by $\mathcal{N}' = \{1, \cdots, N'\}$. For each DA $n \in N'$, denote by $\underline{p}_n(t)$ and $\overline{p}_n(t)$ the lower and upper bounds on the data rate at time $t \in \mathcal{T}$, i.e.,

$$\underline{p}_n(t) \leq p_n(t) \leq \overline{p}_n(t), t \in \mathcal{T}. \tag{2}$$

The lower bounds $\underline{p}_n(t)$ are actually zero, and the upper bounds $\overline{p}_n(t)$ can be set according to, e.g., the available bandwidth. The arrival time t_n^a and the deadline t_n^d can be represented by the rate constraint (2) by setting $\overline{p}_n(t) = 0$ for $t < t_n^a$ and $t > t_n^d$.

Denote those $N - N'$ discrete-rate noninterruptible DAs by $\mathcal{N}'' = \{N' + 1, \cdots, N\}$. For each DA $n \in \mathcal{N}''$, we assume for simplicity that it runs at a single, discrete rate $r_n > 0$. For discrete-rate applications, which are dominantly streaming applications such as online movie watching, a constant bit rate corresponds to a certain graphic quality; e.g., $r_n = 2$ Mb/second for a 720P video on Youtube. As the graphic quality usually does not change during those applications, this seemingly over-simplified assumption of a single discrete rate is reasonable.

For each DA $n \in \mathcal{N}''$ with the total traffic P_n and the rate r_n, it will take $l_n = P_n/r_n$ consecutive timeslots. Therefore, the number of its feasible traffic profiles $A_n = t_n^d - t_n^a - l_n + 1$, and the a-th feasible profile $f_{n,a} = \{p_n |\ p_n(t) = r_n,\ \text{if } t_n^a + a - 1 \leq t \leq t_n^a + a + l_n \ \&\ p_n(t) = 0, \text{otherwise}\}$. We denote the set of all feasible traffic profiles by $\mathcal{F}_n = \{f_{n,a};\ 1 \leq a \leq A_n\}$, i.e., for DA $n \in \mathcal{N}''$ the traffic profile

$$p_n \in \mathcal{F}_n. \tag{3}$$

C. Problem Formulation

We aim to schedule the DAs, so as to flatten the aggregate traffic profile as much as possible. Denote the "average" traffic profile by $d = \{d(t);\ t \in \mathcal{T}\} = \frac{1}{N}\left(b + \sum_{n \in \mathcal{N}} p_n\right)$. Traffic flattening can be achieved by minimizing the variance $V(d)$ of d, formulated as the following optimal demand shaping (ODS) problem.

ODS:

$$\min_{p,d} \quad V(d) = \frac{1}{T}\sum_{t \in \mathcal{T}} \left(d(t) - \frac{1}{T}\sum_{\tau \in \mathcal{T}} d(\tau) \right)^2 \tag{4}$$

$$\text{s.t.} \quad d(t) = \frac{1}{N}\left(b(t) + \sum_{n \in \mathcal{N}} p_n(t) \right), t \in \mathcal{T}, \tag{5}$$

$$0 \leq p_n(t) \leq \overline{p}_n(t),\ t \in \mathcal{T},\ n \in \mathcal{N}', \tag{6}$$

$$\sum_{t \in \mathcal{T}} p_n(t) = P_n, n \in \mathcal{N}', \tag{7}$$

$$p_n \in \mathcal{F}_n, n \in \mathcal{N}'', \tag{8}$$

where $p = \{p_n; n \in \mathcal{N}\}$. In the next section, we will investigate *offline* algorithm for solving the ODS problem under the assumption of complete information about the base traffic and deferrable applications. In the section next, we will study *online* algorithm for demand shaping under the more realistic setting of incomplete information where we can only predict the future traffic to a certain accuracy. The offline ODS problem and algorithm will serve as a benchmark to characterize the performance of the online algorithm.

III. Offline Demand Shaping Algorithm

In this section, we assume complete traffic information, i.e., there is no uncertainty in either base traffic or DAs, and study how to solve the resulting **offline ODS** problem.

A. Convex relaxation

The offline ODS problem is nonconvex, as each discrete-rate noninterruptible DA has to pick a traffic profile from a discrete set; see the constraint (8). Consider the convex hull of $\mathcal{F}_n$

$$\begin{aligned} &\text{conv}(\mathcal{F}_n) \\ =\ &\{\tilde{p}_n |\ \tilde{p}_n = \sum_{a=1}^{A_n} u_{n,a} f_{n,a},\ u_{a,n} \geq 0 \ \&\sum_{a=1}^{A_n} u_{n,a} = 1\}. \end{aligned}$$

We will instead solve the convex relaxation of the ODS problem by replacing (8) with the following constraint:

$$p_n \in \text{conv}(\mathcal{F}_n),\ n \in \mathcal{N}''. \tag{9}$$

We call this relaxed problem the **R-ODS** problem. Since a solution $p_n^*,\ n \in \mathcal{N}''$ to the R-ODS problem might not be feasible, i.e., $p_n^* \notin \mathcal{F}_n$, suppose that p_n^* can be written as the convex combination $\sum_{a=1}^{A_n} u_{n,a} f_{n,a}$, we will randomly pick a traffic profile $p_n = f_{n,a}$ with probability $u_{n,a}$. That said, we will design a randomized algorithm for the offline ODS problem, based on the solution for the R-ODS problem.

B. Distributed algorithm

Solving the R-ODS problem (and the offline ODS problem) in a centralized way requires collecting information on all DAs, which may incur too much communication overhead. We seek to solve it in a distributed way. Notice that R-ODS problem has decoupled constraints, so we may design an *iterative and distributed* algorithm based on the steepest decent method [5].

Before deriving the algorithm, let us first establish a useful result. At k-th iteration, let $p^k = \{p_n^k;\ n \in \mathcal{N}\}$ be the traffic profile, $d^k = \frac{1}{N}(b + \sum_{n \in \mathcal{N}} p_n^k)$ the average traffic profile, and $x_n = p_n^{k+1} - p_n^k,\ n \in \mathcal{N}$ the change in traffic profile between two consecutive iterations. We have

$$E\left[\| \sum_{n \in \mathcal{N}} x_n \|_2^2 \right] = \sum_{n \in \mathcal{N}} Var(x_n) + \| \sum_{n \in \mathcal{N}} E[x_n] \|_2^2,$$

[1] There are applications, e.g., short messages, that are hard to characterize in terms of interruptibility and/or rate. But they only contribute a small portion of traffic, and may incur a large complexity to shape while not help much. We will not seek to control them and will treat them as non-deferrable.

where the variance $Var(x) = E[\|x\|_2^2] - \|E\left[x\right]\|_2^2$, and $E[\cdot]$ denotes the average.[2] By Jensen's inequality, $\|\sum_{n\in\mathcal{N}} E[x_n]\|_2^2 \le N\sum_{n\in\mathcal{N}} \|E[x_n]\|_2^2$. Thus,

$$E\left[\|\sum_{n\in\mathcal{N}} x_n\|_2^2\right] \le \sum_{n\in\mathcal{N}} Var\left(x_n\right) + N\sum_{n\in\mathcal{N}} \|E[x_n]\|_2^2. \quad (10)$$

Now, let $V^k = V(d^k)$, we have

$$E[V^{k+1}|p^k] - V^k = \frac{1}{TN^2}E\left[\|\sum_{n\in\mathcal{N}} x_n\|_2^2 + 2\langle Nd^k, \sum_{n\in\mathcal{N}} x_n\rangle\right].$$

By equation (10),

$$\begin{aligned} & E[V^{k+1}|p^k] - V^k \\ \le\ & \sum_{n\in\mathcal{N}} Var\left(x_n\right) + N\sum_{n\in\mathcal{N}}\|E[x_n]\|_2^2 + 2\sum_{n\in\mathcal{N}} E\left[\langle Nd^k, x_n\rangle\right] \\ =\ & \sum_{n\in\mathcal{N}'}\left(2\langle Nd^k, x_n\rangle + N\left\|x_n\right\|_2^2\right) \\ & + \sum_{n\in\mathcal{N}''}\Big(2\langle Nd^k, E[x_n]\rangle + N\|E[x_n]\|_2^2 + Var\left(x_n\right)\Big). \end{aligned} \quad (11)$$

Denote by W_1 the first term in (11) and W_2 the second term. For $n \in \mathcal{N}'$, we choose p_n^{k+1} so as to minimize W_1, i.e., to solve

$$\begin{aligned} \min_{p_n} \quad & 2\langle d^k, p_n - p_n^k\rangle + \left\|p_n - p_n^k\right\|_2^2 & (12) \\ \text{s.t.} \quad & (6) - (7). & (13) \end{aligned}$$

Also, after some mathematical manipulations, we have

$$\begin{aligned} W_2 \;=\; & \sum_{n\in\mathcal{N}''}(2N\langle d^k - p_n^k, E[p_n^{k+1}]\rangle \\ & +(N-1)\|E[p_n^{k+1}]\|_2^2) + C, \end{aligned}$$

where C is a certain constant that depends on p_n^k. For $n \in \mathcal{N}''$, we choose $p_n^{*k+1} = E[p_n^{k+1}]$ so as to minimize W_2, i.e., to solve

$$\min_{p_n\in\text{conv}(\mathcal{F}_n)} 2\langle d^k - p_n^k, p_n\rangle + \frac{N-1}{N}\|p_n\|_2^2. \quad (14)$$

In essence, what we have done is to maximize the average incremental decrease in objective value at each iteration (i.e., steepest descent). This motivates a distributed demand shaping algorithm with the help of a coordinator; see Algorithm 1. The wireless service provider can implement a logical coordinator at the base station.

Remarks: Notice that, if there is no continuous-rate interruptible DA, Algorithm 1 reduces to the stochastic algorithm in [7]. We expect that the solution approach – randomized algorithm based on the "steepest" descent method for the convex relaxed problem – that we lay out in Sections III-A and III-B will find broad application in designing efficient algorithms for optimization problems that involve both continuous and discrete decision variables.

Algorithm 1 Offline Demand Shaping (Off-DS) Algorithm

```
At $k$-th iteration:
1) Upon gathering traffic profiles $p_n^k$ from DAs, the coordinator calculates the average traffic profile $d^k = \frac{1}{N}\left(b + \sum_{n\in\mathcal{N}} p_n^k\right)$ and announces it to DAs (or the end users) over a signaling or control channel.
2) Upon receiving the average traffic profile $d^k$,
   • Each DA $n \in \mathcal{N}'$ updates its traffic profile according to
        $p_n^{k+1} = \arg \min_{p_n} \left\|p_n - p_n^k + d^k\right\|_2^2$
        $\text{s.t.} \ (6) - (7),$
     and submits it to the coordinator.
   • Each DA $n \in \mathcal{N}''$ calculates the average traffic profile according to
        $p_n^{*k+1} = \arg \min_{p_n\in\text{conv}(\mathcal{F}_n)} \left\|p_n - \frac{N}{N-1}\left(p_n^k - d^k\right)\right\|_2^2,$
     represents it as a convex combination $p_n^{*k+1} = \sum_{a=1}^{A_n} u_{n,a}^{k+1} f_{n,a}$, and then randomly chooses a traffic profile $p_n^{k+1} = f_{n,a}$ with probability $u_{n,a}^{k+1}$ and submits it to the coordinator.
```

C. Convergence

Before showing the convergence of Algorithm 1, we first establish two relations that will be needed. For each DA $n \in \mathcal{N}'$, since p_n^{k+1} solves the problem (12)-(13), we have first-order optimality condition

$$\langle p_n^{k+1} - p_n^k + d^k, p_n - p_n^{k+1}\rangle \ge 0$$

for any feasible p_n. Set $p_n = p_n^k$, we obtain

$$\langle d^k, p_n^{k+1} - p_n^k\rangle \le -\left\|p_n^{k+1} - p_n^k\right\|_2^2. \quad (15)$$

For each DA $n \in \mathcal{N}''$, let $p_n^{*k+1} = E\left[p_n^{k+1}\right]$, i.e., the optimum of the problem (14). By the first oder optimality condition, we have

$$\langle\frac{N}{N-1}\left(d^k - p_n^k\right) + p_n^{*k+1}, p_n - p_n^{*k+1}\rangle \ge 0$$

for any feasible p_n. Set $p_n = p_n^k$, we obtain

$$\begin{aligned} \langle Nd^k, p_n^{*k+1} - p_n^k\rangle \;\le\; & -\left(N-1\right)\left\|p_n^{*k+1} - p_n^k\right\|_2^2 \\ & + \langle p_n^k, p_n^{*k+1} - p_n^k\rangle. \end{aligned} \quad (16)$$

Now, construct a filtration Σ^* of the probability space $\{\Omega, \Sigma, \mathcal{P}\}$, where the sample space Ω is the feasible set specified by the constraints (6)-(8), the σ-algebra $\Sigma_k = \Omega,\ k \ge 0$, and $\mathcal{P}(\Sigma_k) = \{\delta(p_n - p_n^k), n \in \mathcal{N}';\ u_{n,a}^k, 1 \le a \le A_n, n \in \mathcal{N}''\}$, i.e., determined by the k-th iteration of the Off-DS algorithm.

Theorem 1: The pair $(V(d),\ \Sigma^*)$ is a supermartingale.

Proof: First, notice that $V(d)$ is bounded from below, so $E[-\min\{0, V(d)\}] < \infty$. Second, applying relations (15)-

[2] Notice that we consider a randomized scheme only for discrete-rate noninterruptible applications. That said, for continuous-rate interruptible applications there is no randomness and the variance is zero.

(16) to equation (11), we have

$$\begin{aligned}
& E[V^{k+1}|p^k] - V^k \\
\leq\ & \sum_{n\in\mathcal{N}'} -N\left\|x_n\right\|_2^2 + \sum_{n\in\mathcal{N}''}\Big(Var\left(x_n\right) \\
& + \left(-N+2\right)\left\|E[x_n]\right\|_2^2 + 2\langle p_n^k, p_n^{*k+1} - p_n^k\rangle\Big) \\
=\ & \sum_{n\in\mathcal{N}'} -N\left\|x_n\right\|_2^2 + \sum_{n\in\mathcal{N}''}(-N+1)\left\|E[x_n]\right\|_2^2 \\
\leq\ & 0,
\end{aligned}$$

i.e., $E[V^{k+1}|p^k] \leq V^k$. By definition, $(V(d),\ \Sigma^*)$ is a supermartingale [9]. ■

Notice that $(V(d),\ \Sigma^*)$ is a non-negative supermartingale. By the martingale convergence theorem [9], the following result is immediate.

Corollary 1: $V^\infty = \lim_{k\to\infty} V^k$ exists almost surely.

Theorem 2: Denote by $\mathcal{P}^\infty$ an "equilibrium" distribution over traffic profiles that $(V(d),\ \Sigma^*)$ converges to. The support of $\mathcal{P}^\infty$ is a singleton.

Proof: When $(V(d),\ \Sigma^*)$ converges, $E[V^{k+1}|p^k] = V^k$. This requires $E[x_n] = E[x_{n'}],\ n, n' \in \mathcal{N},\ p_n^{k+1} = p_n^k,\ n \in \mathcal{N}'$, and $p_n^{*k+1} = p_n^k,\ n \in \mathcal{N}''$ for (10), (15), and (16) to hold with equality. Notice that $p_n^{*k+1} = p_n^k$ implies $p_n^{k+1} = p_n^k$, as different feasible traffic profiles of DA $n \in \mathcal{N}''$ are linearly independent. Thus, $p_n^{k+1} = p_n^k,\ n \in \mathcal{N}$. So, The support of $\mathcal{P}^\infty$ contains only one point. ■

Denote by p^∞ an "equilibrium" traffic profile of the Off-DS algorithm, i.e., if $p^k = p^\infty$, then $p^{k+1} = p^\infty$. Obviously the set of equilibrium profiles is not empty, as an optimum of the offline ODS problem is an equilibrium. The following result follows immediately from Theorem 2 and Corllary 1.

Theorem 3: The Off-DS algorithm converges almost surely to an equilibrium traffic profile.

IV. Online Demand Shaping Algorithm

In this section, we consider a realistic setting with incomplete information where we can only predict future traffic to a certain accuracy, and study online demand shaping that makes decision based on the prediction of future traffic and updates the decision as new information becomes available.

A typical algorithm used in this setting is the receding horizon control; see, e.g., [14]. However, as the objective function (4) does not have a nice additive structure, receding horizon control algorithm does not admit an easy analysis. We will instead extend a shrinking horizon control algorithm, which is used in [8] that studies mathematically the same problem with only continuous-rate interruptible applications, to include discrete-rate noninterruptible applications, and apply it to our online demand shaping (**Online DS**) problem.

A. Online algorithm

We assume that the number m_t of DAs arriving at time t is randomly distributed with a mean λ_t and variance $(\delta\lambda_t)^2$, and the total amount of traffic of each DA X is randomly distributed with a mean P and variance $(\delta P)^2$. We further assume for simplicity that each DA is equally likely continuos-rate interruptible or discrete-rate noninterruptible. Denote by $\mathcal{N}_t' = \{1, \cdots, N_t'\}$ the set of continuous-rate interruptible DAs and $\mathcal{N}_t'' = \{N'+1, \cdots, N_t''\}$ the set of discrete-rate noninterruptible DAs that have arrived by time $t \in \mathcal{T}$, and let $\mathcal{N}_t = \mathcal{N}_t' \cup \mathcal{N}_t''$ and $N_t = N_t' + N_t''$. Notice that we cannot reschedule the remaining traffic of a discrete-rate noninterruptible DA that has been started. Denote by $\tilde{\mathcal{N}}_t'' \subseteq \mathcal{N}_t''$ the set of discrete-rate noninterruptible DAs that have not been started by time t, and let $\tilde{\mathcal{N}}_t = \mathcal{N}_t' \cup \tilde{\mathcal{N}}_t''$. For each DA $n \in \tilde{\mathcal{N}}_t''$, denote by $\mathcal{F}_n(t) = \{f_{n,a};\ 1 \leq a \leq A_n(t)\}$ the set of feasible traffic profiles at time t.

At time time t, we assume that a prediction b_t of base traffic is available, and the information on DA $n \in \mathcal{N}_t$ and the expected total future deferrable traffic $\sum_{\tau=t+1}^T P\lambda_\tau$ are known. Following [8], we introduce a virtual deferrable traffic profile $q(t:T) = \{q(\tau); t \leq \tau \leq T\}$ with $q(t) = 0$ and $\sum_{\tau=t}^T q(\tau) = \sum_{\tau=t+1}^T P\lambda_\tau$, and use it to emulate the impact of the future deferrable traffic on the current demand shaping decision. With the afore setup, we aim to schedule and reschedule the DAs, so as to solve the following problem at each timeslot $t \in \mathcal{T}$.

ODS$_t$:

$$\min\ V\left(d\right) = \frac{1}{T-t+1}\sum_{\tau=t}^{T}\left(d(\tau) - \frac{\sum_{s=t}^T d(s)}{T-t+1}\right)^2 \tag{17}$$

$$\text{over}\ \ p(t:T), d(t:T), q(t:T)$$

$$\text{s.t.}\ \ d\left(\tau\right) = \frac{b_t(\tau) + q(\tau) + \sum_{n\in\mathcal{N}_t} p_n(\tau)}{N_t}, \tau \geq t, \tag{18}$$

$$0 \leq p_n\left(\tau\right) \leq \overline{p}_n\left(\tau\right), \tau \geq t, n \in \mathcal{N}_t', \tag{19}$$

$$\sum_{\tau=t}^{T} p_n\left(\tau\right) = P_n\left(t\right), n \in \mathcal{N}_t', \tag{20}$$

$$p_n \in \mathcal{F}_n(t), n \in \tilde{\mathcal{N}}_t'', \tag{21}$$

$$\sum_{\tau=t}^{T} q\left(\tau\right) = \sum_{\tau=t+1}^{T} P\lambda_\tau, \tag{22}$$

where $p(t:T) = \{p_n(\tau); t \leq \tau \leq T, n \in \tilde{\mathcal{N}}_t\}$, $d(t:T) = \{d(\tau); t \leq \tau \leq T\}$, and $P_n(t) = P_n - \sum_{\tau=1}^{t-1} p_n(\tau), n \in \mathcal{N}_t'$ is the amount of traffic to be served at or after time t.

We can solve the ODS$_t$ problem the same way as we solve the offline ODS problem (4)-(8), which gives an online demand shaping algorithm; see Algorithm 2. The convergence of Step 2) can be established in the same way as Algorithm 1.

B. Performance analysis

We have characterized the performance of the On-DS algorithm with respect to the optimal offline problem under certain specific assumption; see the extended version [18] for the detail.

V. Numerical Examples

In this section, we provide numerical experiments to evaluate the performance of the On-DS algorithm. We use certain composite traffic traces to drive simulations, to show the impact of base traffic prediction, deferrable traffic prediction, and deferrable traffic penetration level. We expect the conclusions obtained to be hold for real traffic.

```
Algorithm 2 Online Demand Shaping (On-DS) Algorithm
At each time slot $t \in \mathcal{T}$:
  1) Denote by $p_n^{(t-1)}, n \in \mathcal{N}_{t-1}$ the schedules determined
     by time $t-1$, and by $\hat{\mathcal{N}}_t'' \subseteq \mathcal{N}_t''$ the set of discrete-rate
     noninterruptible DAs that has been started before time
     $t$. For each DA $n \in \hat{\mathcal{N}}_t''$, set its schedule $p_n(t;T) =$
     $\{p_n(\tau); t \le \tau \le T\}$ as $p_n(\tau) = p_n^{(t-1)}(\tau),\ t \le \tau \le T$.
  2) Solve the ODS$_t$ problem iteratively
     At $k$-th iteration:
     a) Upon gathering traffic profiles $p_n^k(t:T) =$
        $\{p_n^k(\tau); t \le \tau \le T\}$ from DAs $n \in \tilde{\mathcal{N}}_t$, the
        coordinator solves the following problem
        $$\min_{q(t+1:T)} \sum_{\tau=t+1}^{T} \left( b_t(\tau) + q(\tau) + \sum_{n\in\hat{\mathcal{N}}_t''} p_n(\tau) + \sum_{n\in\tilde{\mathcal{N}}_t} p_n^k(\tau) \right)^2$$
        s.t. (22)
        to obtain a virtual deferrable traffic
        $\{q^k(\tau); t+1 \le \tau \le T\}$, and then
        calculates the average traffic $d^k(\tau) =$
        $\frac{1}{N_t}\left(b_t(\tau) + q^k(\tau) + \sum_{n\in\hat{\mathcal{N}}_t''} p_n(\tau) + \sum_{n\in\tilde{\mathcal{N}}_t} p_n^k(\tau)\right)$
        for $\tau \ge t$ and announces it to DA $n \in \tilde{\mathcal{N}}_t$ over a
        signaling or control channel.
     b) Upon receiving the average traffic profile $d^k$,
        • Each DA $n \in \mathcal{N}_t'$ obtains a new traffic profile
          $p_n^{k+1}(t:T)$ by solving
          $$\min_{p_n(t:T)} \left\| p_n(t:T) - p_n^k(t:T) + d^k(t:T) \right\|_2^2$$
          s.t. (19) – (20),
          and submits it to the coordinator.
        • Each DA $n \in \tilde{\mathcal{N}}_t''$ calculates the average traffic
          profile $p_n^{*k+1}(t:T)$ by solving
          $$\min_{p_n(t:T)} \left\| p_n(t:T) - \frac{N_t}{N_t-1}\left(p_n^k(t:T) - d^k(t:T)\right) \right\|_2^2$$
          s.t (21),
          represents it as a convex combination $p_n^{*k+1} =$
          $\sum_{a=1}^{A_n(t)} u_{n,a}^{k+1} f_{n,a}$, and then randomly chooses a
          traffic profile $p_n^{k+1} = f_{n,a}$ with probability $u_{n,a}^{k+1}$
          and submits it to the coordinator.
```

A. Experimental setup

Consider 24-hour period starting from 4:00pm to 4:00pm on the next day. The duration of one timeslot is set to be 30 minutes, making totally 48 timeslots.

1) Non-deferrable traffic: The "real" trace we use for base traffic, is shown in Fig. 2 (red line), which consists of an average trace (blue line) and randomly generated deviation. The average trace is composed based on the North American mobile web browsing activity by time of day in 2013 [12], shown in Fig. 1. As modeled in Section II-A, the prediction of base traffic follows (1), consisting of average base traffic $\bar{b}(\tau)$ and random deviation from the average value $\delta b_t(\tau)$.

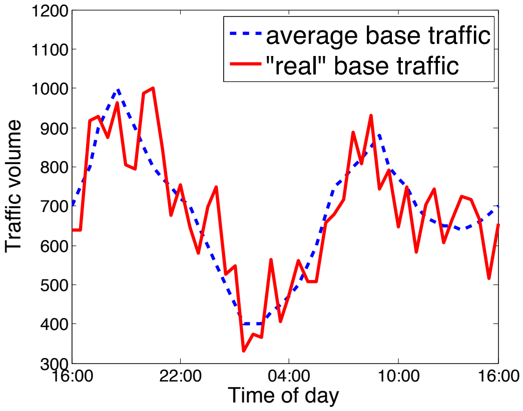

Fig. 2. Base traffic: the average (blue/dotted) and a "real" trace (red/solid).

Following [8], at time t, $\delta b_t(\tau)$ is modeled as

$$\delta b_t(\tau) = \sum_{s=t+1}^{\tau} n_s(\tau), t < \tau \le T,$$

where $n_s(\tau)$ are random variables with Gaussian distribution, with 0 mean and variances

$$E[n_s^2(\tau)] = \frac{\sigma^2}{\tau - s + 1}, 1 \le s \le \tau \le T. \tag{23}$$

In this way, we have $\delta b_t(\tau)$ with decreasing variance as t approaches τ, simulating a gradually improving prediction of future timeslot τ. In simulation, we take the values of σ in (23) from 0 to 100 with increment of 10, corresponding to a root-mean-square prediction error (RMSE) ranging from 0% to 32%, looking T timeslots ahead.

2) Deferrable traffic: We assume that the number of DAs arriving at each time slot follows a "shifted" Poisson process $m + \text{poissrnd}(\lambda_p)$, with $m \ge 0$ and $\text{poissrnd}(\lambda_p)$ denoting a Poisson process with rate λ. The total traffic P_n of each DA is uniformly distributed in $[\underline{P}, \overline{P}]$, and we set $\underline{P} = 12$ and $\overline{P} = 24$ in the numerical examples reported here. The deadline for each DA n is uniformly distributed in $[t_n^a + l_n + \underline{D}, t_n^a + l_n + \overline{D}]$, with $l_n = \ulcorner P_n/\overline{p}_n \urcorner$ the minimum number of timeslots required by the DA, and we set $\underline{D} = 6$ and $\overline{D} = 14$, and the bit rate upper bound $\overline{p}_n = 3$ in the numerical examples reported here. Further, as we have a fixed ending of the time horizon, for those DAs that arrive "close" to the end, in numerical experiments we adjust the amount of traffic (for discrete-rate DAs) or the upper bound on data rate (for continuous-rate DAs) such that they can be finished within the time horizon.

3) Benchmarks for comparison: We compare the performance of the On-DS algorithm with a few typical benchmarks, in order to evaluate (1) the impact of base traffic prediction error, and (2) the impact of DAs' penetration level. We thus consider four cases in our experiments:

(a) *Online demand shaping with On-DS algorithm.* We apply On-DS algorithm to schedule deferrable traffic, based on the prediction of future DAs and the updated prediction of base traffic at each timeslot.

(b) *Offline demand shaping with Off-DS algorithm.* We use complete information including real trace of future base traffic and all DAs' arrival information recorded from (a). By applying Off-DS algorithm, this gives the optimal performance for a given realization of traffic. It is used as a benchmark to characterize the sub-optimality of other cases.

(c) *Online demand shaping with exact information on base traffic and without exact information on DAs.* We apply On-DS algorithm, based on the real trace of base traffic and the prediction of DAs. Since this case assumes perfect information on base traffic, it is used to examine the impact of prediction error of base traffic for cases (a) and (d).

(d) *Demand shaping without updating prediction of base traffic and with exact information on DAs.* We use initial prediction made at $t = 0$ for base traffic and without any further updating, and use arrival information recorded from (a) for DAs. Since this case is only influenced by the prediction error of base traffic, it is used to show the impact of uncertain future arrivals of DAs in cases (a) and (c). Also, comparison between cases (a) and (d) demonstrates the benefit of using updated prediction of base traffic.

Denote by V^{opt} the objective value achieved by the optimal offline algorithm, i.e., case (b). We use the metric $G = \frac{V - V^{opt}}{V^{opt}}$ to measure the "sub-optimality" in performance of other scenarios or algorithms.

B. Experiment results

Given the stochastic nature of deferrable traffic arrival, base traffic prediction, and the decision of traffic profiles of DAs with discrete rate, we run simulation for ten times and show the average over these ten realizations.

1) The impact of base traffic prediction error: We fix the penetration level of deferrable traffic at 10.5%, and tune the variance σ^2 to emulate different levels of prediction error in base traffic (as described in Section V-A.1). As shown in Fig. 3, compared with case (c) that has complete base traffic information, case (a) has a rather good performance despite of the increase in prediction error of base traffic and maintains a sub-optimality of under 10%. This is because our online algorithm keeps improving its prediction of base traffic as time goes by. In contrast, case (d), which does not update base traffic prediction, has poor performance when σ^2 is large even though it has complete information on deferrable traffic. This is because case (d) always uses the initial prediction of base traffic, which can be very different from real value if prediction error is large. We conclude that one of the key features in our online algorithm – updating the prediction – helps ensure the performance of demand shaping.

2) The impact of penetration level of deferrable traffic: We fix the prediction error of base traffic with $\sigma^2 = 50$, and choose different average numbers of DA arrivals at each timeslot to set different deferrable traffic penetration levels. As shown in Fig. 4, case (d) has an improving performance as the penetration level of deferrable traffic increases. This is because it has exact information on deferrable traffic while the impact from the prediction error of base traffic decreases with the increased deferrable traffic penetration. In contrast, we see an increase and then a decrease of sub-optimality for cases (a) and (c) as the penetration level increases. Taking a closer look at the data, however, we find that this does not really mean their performance starts to improve when the deferrable traffic penetration level is large enough; instead, it is because the optimal result from case (b), the benchmark that they are compared against, is getting worse as a result from putting all the unfinished deferrable traffic to the end of the day. We conclude that, as expected, the higher the deferrable traffic penetration is, the worse the online demand shaping is. However, we do not have to worry much about the high penetration level of deferrable traffic, since in practice a penetration of 10-20% is already a large penetration and the online demand shaping has a reasonably good performance within this penetration range.

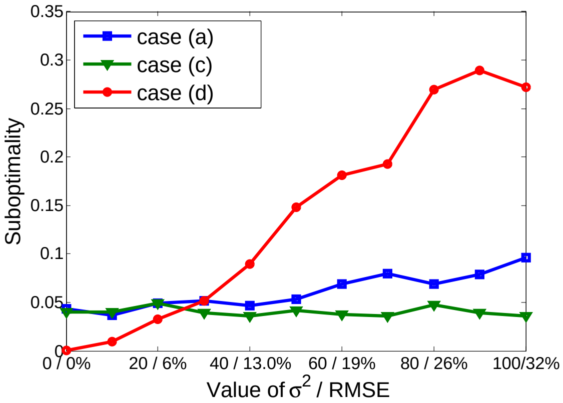

Fig. 3. The impact of base traffic prediction error.

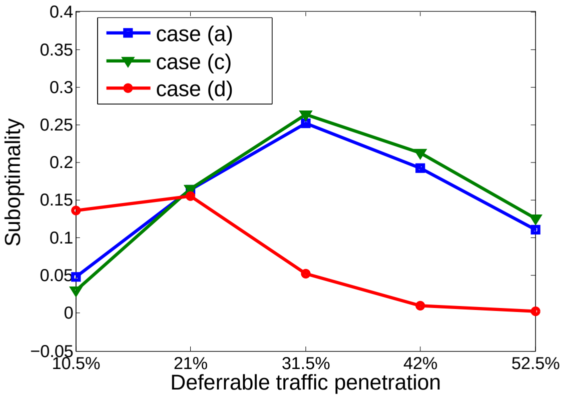

Fig. 4. The impact of penetration level of deferrable traffic.

VI. Conclusion

We have formulated demand shaping in cellular networks as an optimization problem that minimizes the variation in aggregate traffic. We design a distributed and randomized offline demand shaping algorithm under complete traffic information and prove its almost surely convergence. We then consider a more realistic setting with incomplete information where we can only predict future traffic to a certain accuracy, and design an online demand shaping algorithm that updates the schedules of deferrable applications each time new information is available, based on solving at each timeslot an optimization problem over a shrinking horizon from the current time to the end of the day. We compare the performance of the online algorithm against the optimal offline algorithm, and provide numerical examples. As future work, we are investigating to integrate the incentive mechanisms such as the smart data pricing into the demand shaping algorithm design. We also plan to develop a platform to enable automatic demand shaping in cellular networks and investigate the related practical issues.

Acknowledgement

We would like to thank Seungil You for helping with simulations and Lingwen Gan for careful comments.

References

[1] Airtel launches unlimited-usage night plans for calls, internet. http://businesstoday.intoday.in/story/airtel-night-plans-unlimited-usage-for-calls-internet/1/205272.html.

[2] Sprint night and weekend minutes. http://shop2.sprint.com/en/stores/popups/voice_nights_weekends_7pm_popup.shtml.

[3] Telkom night surfer plan. http://www.telkommobile.co.za/plans/prepaid-data/60gbpromo/.

[4] Verizon nationwide for business plans. http://business.verizonwireless.com/content/b2b/en/shop-business-products/business-plans/nationwide-for-business.html.

[5] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[6] L. Chen, L. Jiang, N. Li, and S. H. Low. Optimal Demand Response: Problem Formulation and Deterministic Case. In A. Chakrabortty and M. Ilic, editors, *Control and Optimization Theory for Electric Smart Grids*. Springer, 2012.

[7] Lingwen Gan, Ufuk Topcu, and Steven H Low. Stochastic distributed protocol for electric vehicle charging with discrete charging rate. In *Power and Energy Society General Meeting, 2012 IEEE*, pages 1–8. IEEE, 2012.

[8] Lingwen Gan, Adam Wierman, Ufuk Topcu, Niangjun Chen, and Steven H Low. Real-time deferrable load control: handling the uncertainties of renewable generation. In *Proceedings of the fourth international conference on Future energy systems*, pages 113–124. ACM, 2013.

[9] G. R. Grimmett and D. R. Stirzaker. *Probability and Random Process*. Oxford University Press, third edition, 2001.

[10] Sangtae Ha, Soumya Sen, Carlee Joe-Wong, Youngbin Im, and Mung Chiang. Tube: time-dependent pricing for mobile data. *ACM SIGCOMM Computer Communication Review*, 42(4):247–258, 2012.

[11] Cisco Visual Networking Index. Global mobile data traffic forecast update, 2012–2017. 2013.

[12] Chitika Insights. Hour-by-hour examination: Smartphone, tablet, and desktop usage rates.

[13] Carlee Joe-Wong, Sangtae Ha, and Mung Chiang. Time-dependent broadband pricing: Feasibility and benefits. In *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*, pages 288–298. IEEE, 2011.

[14] W. H. Kwon and A.E. Pearson. A modified quadratic cost problem and feedback stabilization of a linear system. *Automatic Control, IEEE Transactions on*, 22(5):838–842, Oct 1977.

[15] N. Li, L. Chen, and S. H. Low. Optimal demand response based on utility maximization in power networks. In *Proceedings of IEEE Power Engineering Society General Meeting*, July 2011.

[16] Carlee Joe-Wong Youngbin Im Sangtae Ha, Soumya Sen and Mung Chiang. Tube survey questions and demographics. http://www.princeton.edu/~cjoe/TUBE_Survey.pdf, Jan 2012.

[17] Soumya Sen, Carlee Joe-Wong, Sangtae Ha, and Mung Chiang. Smart data pricing (sdp): Economic solutions to network congestion. *SIGCOMM eBook on Recent Advances in Networking*, 2013.

[18] X. Zhou and L. Chen. Demand shaping in cellular networks. *Technical Report*, 2014. http://spot.colorado.edu/∼lich1539/papers/DS.pdf.