

The Learning and Prediction of Application-Level Traffic Data in Cellular Networks

Rongpeng Li, Zhifeng Zhao, Jianchao Zheng, Chengli Mei, Yueming Cai, and Honggang Zhang

Abstract—Traffic learning and prediction is at the heart of the evaluation of the performance of telecommunications networks and attracts a lot of attention in wired broadband networks. Now, benefiting from the big data in cellular networks, it becomes possible to make the analyses one step further into the application level. In this paper, we first collect a significant amount of application-level traffic data from cellular network operators. Afterward, with the aid of the traffic “big data,” we make a comprehensive study over the modeling and prediction framework of cellular network traffic. Our results solidly demonstrate that there universally exist some traffic statistical modeling characteristics at a service or application granularity, including α -stable modeled property in the temporal domain and the sparsity in the spatial domain. But, different service types of applications possess distinct parameter settings. Furthermore, we propose a new traffic prediction framework to encompass and explore these aforementioned characteristics and then develop a dictionary learning-based alternating direction method to solve it. Finally, we examine the effectiveness and robustness of the proposed framework for different types of application-level traffic. Our simulation results prove that the proposed framework could offer a unified solution for application-level traffic learning and prediction and significantly contribute to solve the modeling and forecasting issues.

Index Terms—Big data, cellular networks, traffic prediction, α -stable models, dictionary learning, alternative direction method, sparse signal recovery.

I. INTRODUCTION

TRAFFIC learning and prediction in cellular networks, which is a classical yet still appealing field, yields a significant number of meaningful results. From a macroscopic perspective, it provides the commonly believed result that mobile Internet will witness a 1000-folded traffic growth in the next 10 years [1], which is acting as a crucial anchor for the design of next-generation cellular network architecture

Manuscript received April 28, 2016; revised September 14, 2016 and January 16, 2017; accepted March 26, 2017. Date of publication March 30, 2017; date of current version June 8, 2017. This work was supported by the Program for Zhejiang Leading Team of Science and Technology Innovation under Grant 2013TD20, in part by the Zhejiang Provincial Technology Plan of China under Grant 2015C01075, and in part by the the National Postdoctoral Program for Innovative Talents of China under Grant BX201600133. The associate editor coordinating the review of this paper and approving it for publication was J. Tang. (*Corresponding author: Zhifeng Zhao.*)

R. Li, Z. Zhao, and H. Zhang are with Zhejiang University, Hangzhou 310027, China (e-mail: lirongpeng@zju.edu.cn; zhaozf@zju.edu.cn; honggangzhang@zju.edu.cn).

J. Zheng and Y. Cai are with PLA University of Science and Technology, Nanjing 210007, China (e-mail: longxingren.zjc.s@163.com; caiym@vip.sina.com).

C. Mei is with the China Telecom Technology Innovation Center, Beijing 100045, China (e-mail: meichl@ctbri.com.cn).

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

Digital Object Identifier 10.1109/TWC.2017.2689772

and embedded algorithms. On the other hand, the fine traffic prediction on a daily, hourly or even minutely basis could contribute to the optimization and management of cellular networks like energy savings [2], opportunistic scheduling [3], and network anomaly detection [4]. In other words, a precisely predicted future traffic load knowledge, which contributes to improving the network energy efficiency by dynamically configuring network resources according to the practical traffic demand [5], [6], plays an important role in designing greener traffic-aware cellular networks.

Our previous research [7] has demonstrated the microscopic traffic predictability in cellular networks for circuit switching’s voice and short message service and packet switching’s data service. However, compared to the more accurate prediction performance for voice and text service in circuit switching domain, the state-of-the-art research in packet switching’s data service is still not satisfactory enough. Furthermore, the fifth-generation (5G) cellular networks, which is under the standardization and assumed to be the key enabler and infrastructure provider in the information communication technology industry, aim to cater different types of services like enhanced mobile broadband (eMBB) with bandwidth-consuming and throughput-driving requirements, ultra-reliable low latency service (URLLC), etc. Hence, if we can detect the coming of the service with higher priority, we can timely reserve and specifically configure the resources (e.g., shorter transmission time interval) to guarantee the service provisioning. In a word, a learning and prediction study over application-level data traffic might contribute to understanding data service’s characteristics and performing finer resource management in the 5G era [8]. But, as listed in Table I, the applications (i.e., instantaneous message (IM), web browsing, video) in cellular networks are impacted by different factors and also significantly differ from those in wired networks. Hence, instead of directly applying the results generated from wired network traffic, we need to re-examine the related traffic characteristics in cellular networks and check the prediction accuracy of the application-level traffic. In order to obtain general results, we firstly collect a significant amount of practical traffic records from China Mobile.¹ By taking advantage of the traffic “big data”, we then confirm the preciseness of fitting α -stable models to these typical types of traffic and demonstrate α -stable models’ universal existence in cellular network traffic. We later show that α -stable models

¹It is worthwhile to note here that we also collect another dataset from China Telecom to further verify the effectiveness of the thoughts inside in this paper. Due to the space limitation, we put the results related to China Telecom dataset in a separate file available at http://www.rongpeng.info/files/sup_file_twc.pdf.

TABLE I
THE DIFFERENCES FOR APPLICATIONS IN CELLULAR AND WIRED NETWORKS

Key Difference	Typical Service	Remarks
Protocol	Instantaneous Messaging (IM)	Compared to their counterparts (Skype) in wired networks, mobile IM and social networking applications depend on the keep-alive mechanism to timely receive newly arrived information. In other words, user devices periodically wake up and consume certain signaling resources to build connections, when the keep-alive timer exceeds a predefined threshold.
User Mobility	Web Browsing	In wired networks, users usually rely on almost immobile personal computers to connect the Internet. However, users in cellular networks might travel in cars or trains, thus requiring the network to hand over the signaling and data information from one cell to another.
Billing Policy	Video	Wired network operators charge subscribers by bandwidth while most cellular network operators charge subscribers by consumed data volume. Hence, users in cellular networks are reluctant to watch expensive movies.

can be used to leverage the temporally long range dependence and guide linear algorithms to conduct the traffic prediction. Besides, we find that spatial sparsity is also applicable for the application-level traffic and propose that the predicted traffic should be able to be mapped to some sparse signals. In this regard, benefiting from the latest progress in compressive sensing [9]–[12], we could calibrate the traffic prediction results with the transform matrix unknown a priori. Finally, in order to forecast the traffic with the aforementioned characteristics, we formulate the prediction problem by a new framework and then develop a dictionary learning based alternating direction method (ADM) [12] to solve it.

A. Related Work

Due to its apparent significance, there have already existed two research streams toward the fine traffic prediction issue in wired broadband networks and cellular networks [7]. One is based on fitting models (e.g., ON-OFF model [13], ARIMA model [14], FARIMA model [15], mobility model [16], [17], network traffic model [17], and α -stable model [18], [19]) to explore the traffic characteristics, such as spatial and temporal relevancies [20] or self-similarity [21], [22], and obtain the future traffic by appropriate prediction methods. The other is based on modern signal processing techniques (e.g., principal components analysis method [23], [24], Kalman filtering method [24], [25] or compressive sensing method [2], [12], [23], [26]) to capture the evolution of traffic. However, it is useful to first model large-scale traffic vectors as sparse linear combinations of basis elements. Therefore, some dictionary learning method [27] is necessary to learn and construct the basis sets or dictionaries.

However, the existing traffic prediction methods in this microscopic case still lag behind the diverse requirements of various application scenarios. Firstly, most of them still focus on the traffic of all data services [28] and seldom shed light on a specific type of services (e.g., video, web browsing, IM, etc). Secondly, the existing prediction methods usually follow the analysis results in wired broadband networks like the α -stable models² [29], [30] or the often accompanied self-similarity [21] to forecast future traffic values [15], [19], [22].

²In this paper, the term “ α -stable models” is interchangeable with α -stable distributions.

TABLE II
DATASET 1 UNDER STUDY

	IM (Weixin)	Web Browsing (HTTP)	Video (QQLive)
Traffic Resolution (Collection Interval)	5 min	5 min	5 min
Duration	1 day	1 day	1 day
No. of Active Cells	2292	4507	4472
Location Info. (Latitude & Longitude)	Yes	Yes	Yes

But the corresponding results need to be validated before being directly applied to cellular networks [7], since cellular networks have more stringent constraints on radio resources [31], relatively expensive billing policies and different user behaviors due to the mobility [32] and thus exhibit distinct traffic characteristics.

B. Contribution

Compared to the previous works, this paper aims to answer how to accurately model, effectively profile, and efficiently predict mobile traffic at an *application* or *service* granularity. Belonging to one of the pioneering works toward application-level traffic analyses, we take advantage of a large amount of practical records (as summarized in Table II and Table III) and provide the following key insights:

- Firstly, this paper visits α -stable models and confirms their accuracy to model the application-level cellular network traffic for all three service types (i.e., IM, web browsing, video). To our best knowledge, it is the first in the literature to find an appropriate model for application-level traffic in cellular networks and show the modeling accuracy of α -stable models. Moreover, this paper shows the application-level traffic obeys the sparse property and demonstrates the distinct characteristics among different service types. Therefore, the paper contributes to a general understanding of the cellular network traffic.
- Secondly, in order to encompass and explore these aforementioned characteristics, this paper provides a traffic prediction framework in Fig. 1. Specifically,

TABLE III
DATASET 2 UNDER STUDY

	IM (QQ)	Web Browsing (HTTP)	Video (QQLive)
Traffic Resolution (Collection Interval) Δt	30 min	30 min	30 min
Duration	2 weeks	2 weeks	2 weeks
No. of Active Cells	5868	5984	5906
Location Info. (Latitude & Longitude)	Yes	Yes	Yes

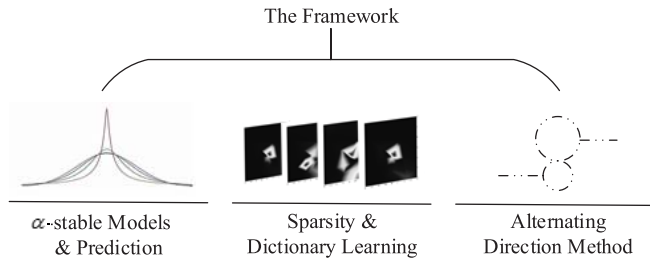


Fig. 1. Illustration of application-level traffic prediction framework.

the proposed framework consists of an “ α -Stable Model & Prediction” module to generate coarse prediction results, a “Sparsity & Dictionary Learning” module to impose a sparse constraint and refine the prediction results, and an “Alternating Direction Method” module to provide the algorithmic details and obtain the final results.

- Thirdly, this paper further demonstrates appealing prediction performance by extensive simulation results. In other words, this paper proves the existence and effectiveness of a unified solution for application-level mobile traffic. Hence, it could simplify the modeling, analyses and prediction for application-level traffic and contribute to the building of service-aware networks in the 5G era.

The remainder of the paper is organized as follows. In Section II, we first present some necessary background of required mathematical tools. In Section III, we introduce the dataset for traffic prediction analyses and later talk about the characteristics (i.e., α -stable models and spatial sparsity) of the application-level dataset. In Section IV, we propose a new traffic prediction framework and its corresponding solution. Section V evaluates the proposed schemes and presents the validity and effectiveness. Finally, we conclude this paper in Section VI.

Notation: In the sequel, bold lowercase and uppercase letters (e.g., \mathbf{x} and \mathbf{X}) denote a vector and a matrix, respectively. $(\cdot)^T$ denotes a transpose operation of a matrix or vector. $\|\mathbf{x}\|_0$ is an l_0 -norm, counting the number of non-zero entries in \mathbf{x} , while an l_p -norm $\|\mathbf{x}\|_p$, $p \geq 1$ of a $1 \times n$ vector $\mathbf{x} = (x_1, \dots, x_n)$ is defined by $\sqrt[p]{\sum_i |x_i|^p}$. The operation $\langle \mathbf{x}, \mathbf{y} \rangle$ denotes the summation operation of element-wise multiplication in \mathbf{x} and \mathbf{y} with the same size. $\text{sgn}(x)$ with respect to $x \in \mathcal{R}$ is defined as $\text{sgn}(x) = x/|x|$ when $x \neq 0$; and $\text{sgn}(x) = 0$ when $x = 0$.

II. MATHEMATICAL BACKGROUND

A. α -Stable Models

Following the generalized central limit theorem, α -stable models manifest themselves in the capability to approximate the distribution of normalized sums of a relatively large number of independent identically distributed random variables [33] and lead to the accumulative property. Besides, α -stable models produce strong bursty results with properties of heavy tailed distributions and long range dependence. Therefore, they arise in a natural way to characterize the traffic in wired broadband networks [34], [35] and have been exploited in resource management analyses [36], [37].

α -stable models, with few exceptions, lack a closed-form expression of the probability density function (PDF) and are generally specified by their characteristic functions.

Definition 1: A random variable T is said to obey α -stable models if there are parameters $0 < \alpha \leq 2$, $\sigma \geq 0$, $-1 \leq \beta \leq 1$, and $\mu \in \mathcal{R}$ such that its characteristic function is of the following form:

$$\begin{aligned} \Phi(\omega) &= E(\exp j\omega T) \\ &= \begin{cases} \exp \left\{ -\sigma^\alpha |\omega|^\alpha \left(1 - j\beta \text{sgn}(\omega) \tan \frac{\pi\alpha}{2} \right) + j\mu\omega \right\}, & \alpha \neq 1; \\ \exp \left\{ -\sigma |\omega| \left(1 + j\frac{2\beta}{\pi} \text{sgn}(\omega) \ln |\omega| \right) + j\mu\omega \right\}, & \alpha = 1. \end{cases} \end{aligned} \quad (1)$$

Here, the function $E(\cdot)$ represents the expectation operation with respect to a random variable. α is called the characteristic exponent and indicates the index of stability, while β is identified as the skewness parameter. α and β together determine the shape of the models. Moreover, σ and μ are called scale and shift parameters, respectively. In particular, if $\alpha = 2$, α -Stable models reduce to Gaussian distributions.

Furthermore, for an α -stable modeled random variable T , there exists a linear relationship between the parameter α and the function $\Psi(\omega) = \ln \{-\text{Re}[\ln(\Phi(\omega))]\}$ as

$$\Psi(\omega) = \ln \{-\text{Re}[\ln(\Phi(\omega))]\} = \alpha \ln(\omega) + \alpha \ln(\sigma), \quad (2)$$

where the function $\text{Re}(\cdot)$ calculates the real part of the input variable.

Usually, it's challenging to prove whether a dataset follows a specific distribution, especially for α -stable models without a closed-form expression for the PDF. Therefore, when a dataset is said to satisfy α -stable models, it usually means the dataset is consistent with the hypothetical distribution and the corresponding properties. In other words, the validation needs to firstly estimate parameters of α -stable models from the given dataset and then compare the real distribution of the dataset with the estimated α -stable model [35]. Specifically, the corresponding parameters in α -stable models can be determined by maximum likelihood methods, quantile methods, or sample characteristic function methods [34], [35].

B. Sparse Representation and Dictionary Learning

In recent years, sparsity methods or the related compressive sensing (CS) methods have been significantly investigated [9]–[12]. Mathematically, sparsity methods aim to tackle this sparse signal recovery problem in the form of

$$\begin{aligned} \min \quad & \|s\|_0, \\ \text{s.t.} \quad & \mathbf{y} = \mathbf{D}s, \end{aligned} \quad (3)$$

or

$$\begin{aligned} \min \quad & \|s\|_0, \\ \text{s.t.} \quad & \|\mathbf{y} - \mathbf{D}s\| \leq \iota. \end{aligned} \quad (4)$$

Here, s denotes a sparse signal vector while \mathbf{y} denotes a measurement vector based on a transform matrix or dictionary \mathbf{D} . Besides, ι is a predefined integer indicating the sparsity. By leveraging the embedded sparsity in the signals, sparsity methods could successfully recover the sparse signal with a high probability, depending on a small number of measurements fewer than that required in Nyquist sampling theorem. Basis pursuit (BP) [38], one of typical sparsity methods, solves the problem in terms of maximizing a posterior (MAP) criterion by relaxing the l_0 -norm to an l_1 -norm. On the other hand, orthogonal matching pursuit (OMP) [39] greedily achieves the final outcome in a sequential manner, by computing inner products between the signal and dictionary columns and possibly solving them using the least square criterion.

For sparsity methods above, there usually exists an assumption that the transform matrix or dictionary \mathbf{D} is already known or fixed. However, in spite of their computation simplicity, such pre-specified transform matrices like Fourier transforms and overcomplete wavelets might not be suitable to lead to a sparse signal [40]. Consequently, some researchers proposed to design \mathbf{D} based on learning [27], [40]. In other words, during the sparse signal recovery procedure, machine learning and statistics are leveraged to compute the vectors in \mathbf{D} from the measurement vector \mathbf{y} , so as to grant more flexibility to get a sparse representation s from \mathbf{y} . Mathematically, dictionary learning methods would yield a final transform matrix by alternating between a sparse computation process based on the dictionary estimated at the current stage and a dictionary update process to approach the measurement vector.

III. APPLICATION-LEVEL TRAFFIC DATASET AND ITS CHARACTERISTICS

A. Traffic Dataset Description

In this paper, our datasets are based on a significant number of practical traffic records from China Mobile in Hangzhou, an eastern provincial capital in China via the Gb interface of 2G/3G cellular networks or S1 interface of 4G cellular networks [41]. Specifically, the datasets encompass nearly 6000 cells' location records³ with more than 7 million subscribers involved. The datasets also contain the information

³Indeed, at one specific location, there might exist several cells operating on different frequencies or modes. For simplicity of representation, in the following analyses, we merge the information for different cells at the same location into one.

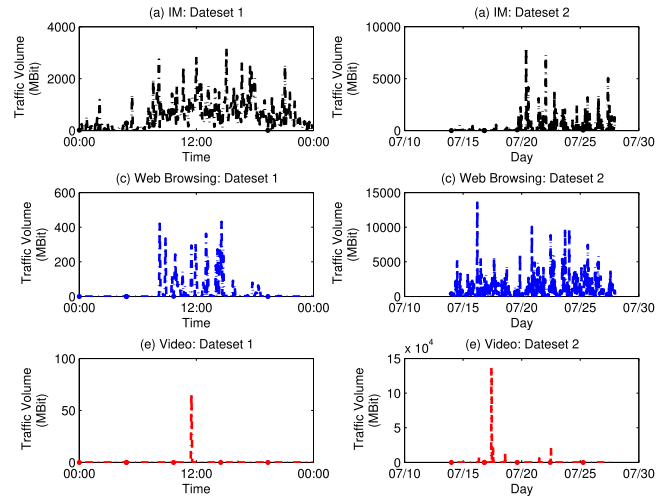


Fig. 2. Traffic variations of applications in different service types in the randomly selected (single) cells.

like timestamp, corresponding cell ID, and traffic-related application-layer information, thus being possible for us to differentiate applications. In particular, we can determine web browsing service and video service by the applied HTTP protocol and streaming protocol respectively, while we assume traffic records to belong to the IM service, after fitting them to the learning results of regular IM packet pattern (e.g., port number, Internet protocol address, and explicit application-layer information in the header). Moreover, the traffic volume could be calculated after aggregating packets to each influx base station. Notably, the paper aims to predict the traffic volume in each BS instead of the entire cellular network.

According to the traffic resolution (e.g., the traffic collection interval, namely 5 minutes and 30 minutes), the collected data can be sorted into two categories. Table II summarizes the information of per 5-minute traffic records collected on September 9th, 2014 with Weixin/Wechat,⁴ HTTP Web Browsing, and QQLive Video⁵ selected as the representatives of these three service types. Here, the term “no. of active cells” refers to the number of cells where a specific type of service happened. Similarly, Table III lists the corresponding details of per 30-minute traffic records from July 14th, 2014 to July 27th, 2014 with QQ,⁶ HTTP Web Browsing, and QQLive Video as the representatives, respectively.

Based on the datasets in Table II and Table III, Fig. 2 illustrates the traffic variations generated by these applications in the randomly selected cells. Indeed, the phenomena in Fig. 2 universally exist in other individual cells and lead to the following insight.

Remark 1: Different services exhibit distinct traffic characteristics. IM and HTTP web browsing services frequently

⁴Weixin/Wechat provides a Whatsapp-like instant messaging service developed by Tencent Inc. and is one of the most popular mobile social applications in China with more than 400 million active users.

⁵QQLive Video is a popular live streaming video platform in China.

⁶QQ is another instant messaging service developed by Tencent Inc. with more than 800 million active users. Due to some practical reasons, per 30-minute Weixin traffic records are unavailable. Therefore, Table III includes QQ's traffic records.

TABLE IV
THE PARAMETER FITTING RESULTS IN THE α -STABLE
MODELS BASED ON DATASET 1

Name	Parameters				K-S Test	
	α	β	σ	μ	GoF	95% Thres.
IM	1.61	1	188.67	221.83	0.0576	0.0800
Web Browsing	1.60	1	32.33	42.75	0.0434	0.0800
Video	0.51	1	1×10^{-10}	0	0.0382	0.0800

TABLE V
THE PARAMETER FITTING RESULTS IN THE α -STABLE
MODELS BASED ON DATASET 2

Name	Parameters				K-S Test	
	α	β	σ	μ	GoF	95% Thres.
IM	0.70	1	26.32	-100.69	0.0483	0.0524
Web Browsing	2	1	2.03×10^3	2.01×10^3	0.0504	0.0524
Video	0.51	1	136.52	-341.15	0.0237	0.0524

produce traffic loads; while distinct from them, video service with more sporadic activities may generate more significant traffic loads.

For simplicity of representation, we introduce a traffic vector \mathbf{x} , whose entries archive the volume of traffic in one given cell at different moments. Furthermore, by augmenting the traffic vectors for different cells, we refer to a traffic matrix \mathbf{X} to denote the traffic records in an area of interest. Then, every row vector of traffic matrix indicates traffic loads at one specific cell with respect to the time while every column vector reflects volumes of traffic of several adjacent cells at one specific moment. Specifically, for a traffic resolution Δt , $X(i, t)$ in a traffic matrix \mathbf{X} denotes traffic loads of cell i from t to $t + \Delta t$.

Remark 2: Traffic prediction can be regarded as the procedure to obtain a column vector $\hat{\mathbf{x}}_p = \hat{\mathbf{X}}(:, t)^T$ at a future moment t , based on the already known traffic records. Each entry $\hat{x}_p^{(i)}$ in $\hat{\mathbf{x}}_p$ corresponds to the future traffic for cell i .

B. The α -Stable Modeling and Sparse Properties

In this section, we examine the results of fitting the application-level dataset to α -stable models. Firstly, in Table IV and Table V, we list the parameter fitting results using quantile methods [42], when we take into consideration the traffic records in three randomly selected cells (each for one service type) of Table II and Table III and quantize the volume of each traffic vector into 100 parts.

Afterwards, we use the α -stable models, produced by the aforementioned estimated parameters, to generate some random variable and compare the induced quantized cumulative distribution function (CDF) with the real quantized one. Table IV and Table V summarize the related fitting parameters. Notably, as the parameter μ indicates the shift of the probability density function (PDF) and equals the mean of the variable when $\alpha \in (0, 1)$, the PDF for some negative interval is non-zero and then it makes no sense for the cellular network traffic when $\mu < 0$. Consequently, in the stricter sense, when

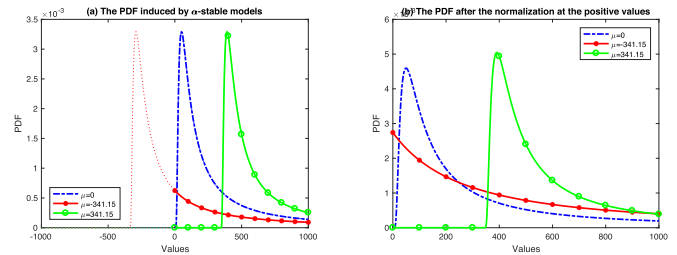


Fig. 3. Illustration of the PDF induced by α -stable models.

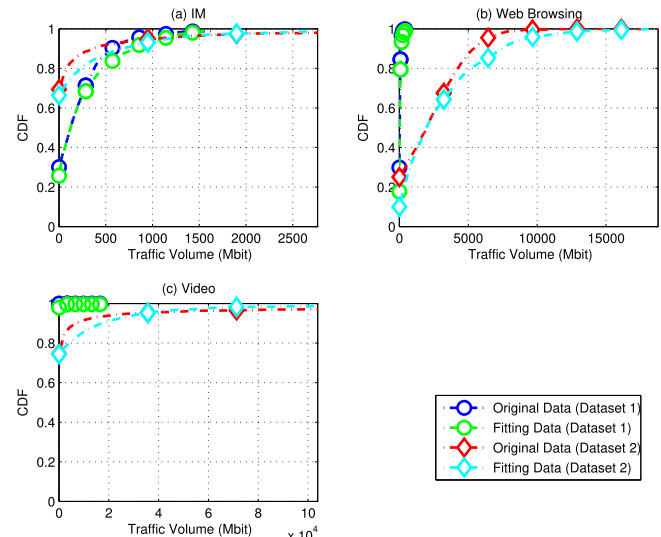


Fig. 4. For different service types, α -stable model fitting results versus the real (empirical) ones in terms of the cumulative distribution function (CDF).

we discuss the α -stable modeling, we only consider the non-negative interval of the variable and normalize the PDF during such an interval with practical meaning. Mathematically, for a variable X with the PDF $\mathcal{P}(X)$, the normalized PDF $\bar{\mathcal{P}}(X)$ could be expressed as

$$\bar{\mathcal{P}}(X = x) = \begin{cases} \frac{\mathcal{P}(x)}{\int_{y \geq 0} \mathcal{P}(y) dy}, & x \geq 0; \\ 0, & x < 0. \end{cases} \quad (5)$$

For example, Fig. 3(a) illustrates the PDF of an α -stable modeled video service with $\alpha = 0.51$, $\beta = 1$, $\sigma = 136.52$ and different μ . From Fig. 3(a), when μ varies, the PDF shifts accordingly. For the video service with $\mu = -341.15$, we actually talk about the normalized PDF in Fig. 3(b). Fig. 4 presents the corresponding comparison between the simulated results and the real ones. Notably, due to the quantization in the real and estimated CDF, if the PDF at the first quantized value does not equal to zero (e.g., 0.3 for Fig. 4(a) and Fig. 4(b)), the corresponding CDF will start from a positive value. As stated in Section II-A, if the simulated dataset has the same or approximately same distribution as the real one, the empirical dataset could be deemed as α -stable modeled. Therefore, Fig. 4 indicates the traffic records in these selected areas could be simulated by α -stable models. We also perform the Kolmogorov-Smirnov (K-S) goodness-of-fit (GoF) test [43] and compare the K-S GoF values and the 95%-confidence

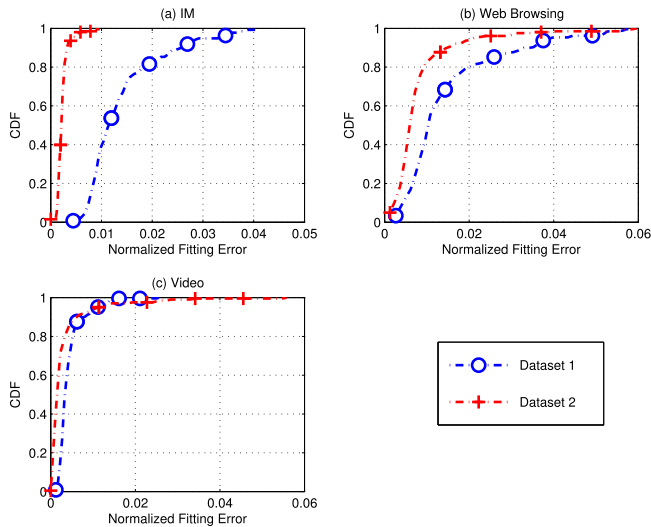


Fig. 5. Precision error CDF for all the cells after fitting $\Psi(\omega)$ with respect to $\ln(\omega)$ to a linear function.

thresholds in Table IV and Table V. From the tables, the GoF values are smaller than the thresholds, which further validates the conclusion drawn from Fig. 4.

On the other hand, recalling the statements in Section II, for an α -stable modeled random variable X , there exists a linear relationship between the parameter α and the function $\Psi(\omega) = \ln\{-\text{Re}[\ln(\Phi(\omega))]\}$. Thus, we fit the estimated parameter α with the computing function $\Psi(\omega)$ and provide the preciseness error CDF for all the cells in Fig. 5. According to Fig. 5, the normalized fitting errors for 80% cells in both datasets are less than 0.02. Therefore, the practical application-level traffic records follow the property of α -stable models (in Eq. (2)) and further enhance the validation results by Fig. 4. Moreover, different application-level traffic exhibits different fitting accuracy. In that regard, the video traffic in Fig. 5(c) has the minimal fitting error, while the fitting error of the web browsing traffic in Fig. 5(b) is the largest. But, the fitting error quickly decreases along with the increase in traffic resolution, since a larger traffic resolution means a confluence of more application-level traffic packets and could better demonstrate the accumulative property of α -stable models.

Remark 3: Due to their generality, α -stable models are suitable to characterize the application-level traffic loads in cellular networks, even though it might not be the most accurate one.

Indeed, the universal existence of α -stable models also implies the self-similarity of application-level traffic [21]. Hence, in the following sections, it is sufficient to only present and discuss the results from Dataset 1 in Table II. On the other hand, the phenomena that application-level traffic universally obeys α -stable models can be explained as follows. Our previous study [44] unveiled that the message length of one individual IM activity follows a power-law distribution. Moreover, according to the generalized central limit theorem [45], the sum of a number of random variables with power-law distributions decreasing as $|x|^{-\alpha-1}$ where $0 < \alpha < 2$ (and therefore having infinite variance) will tend to an α -stable model as the number of summands grows.

Hence, the application-level traffic within one cell follows α -stable models, as the traffic distribution within one cell can be regarded as the accumulation of lots of IM activities.

Additionally, data traffic in wired broadband networks [23] and voice and text traffic in circuit switching domain of cellular networks [2] prove to possess the spatio-temporal sparsity characteristic. Indeed, the application-level traffic spatially possesses this sparse property as well. Fig. 6 depicts the traffic density in 10AM and 4PM in randomly selected dense urban areas. Here, the traffic density is achieved by dividing the cell traffic of each BS by the corresponding Voronoi cell area [46]. When the derived traffic density in one cell is comparatively larger than that in others, it is depicted as a red “hot spot”. As shown in Fig. 6, there appear a limited number of traffic hotspots and the number of “hot spots” change in both temporal and spatial domain. This spatially clustering property is also consistent with the findings in [20] and proves the traffic’s spatial sparsity. It can also be observed that the locations of “hot spots” are also service-specific. In other words, different services have distinct requirements on bandwidth, thus leading to various types of user behavior. For example, video service, which usually consumes huge traffic budget and only is affordable for few subscribers, yield only the smallest number of “hot spots”.

Remark 4: The application-level traffic dataset further validates that the traffic for different service types of applications follows a spatially sparse property. Besides, compared to IM and web browsing service, video service exhibits the strongest sparsity.

IV. APPLICATION-LEVEL TRAFFIC PREDICTION FRAMEWORK

Section III unveils that the application-level cellular network traffic could be characterized by α -stable models and obey the sparse property. In this section, we aim to fully take advantage of these results and propose a new framework in Fig. 1 to predict the traffic. The proposed framework consists of three modules. Among them, the “ α -Stable Model & Prediction” module would take advantage of the already known traffic knowledge to learn and distill the parameters in α -stable models and provide a coarse prediction result. Meanwhile, the “Sparsity & Dictionary Learning” module imposes constraints to make the final prediction results satisfy the spatial sparsity. But, these two modules inevitably add multiple parameters unknown a priori and thus need specific mathematical operations to obtain a solution. Hence, the proposed framework also contain an “Alternating Direction Method” module to iteratively process the other modules and yield the final result.

A. Problem Formulation

Previous sections unearth several important characteristics in application-level traffic in cellular networks, including spatial sparsity and temporally α -stable modeling. All these factors could be leveraged for forecasting the future traffic vector $\hat{\mathbf{x}}_p$.

- Temporal modeling component. As Section III-B states, the application-level traffic loads follow α -stable models.

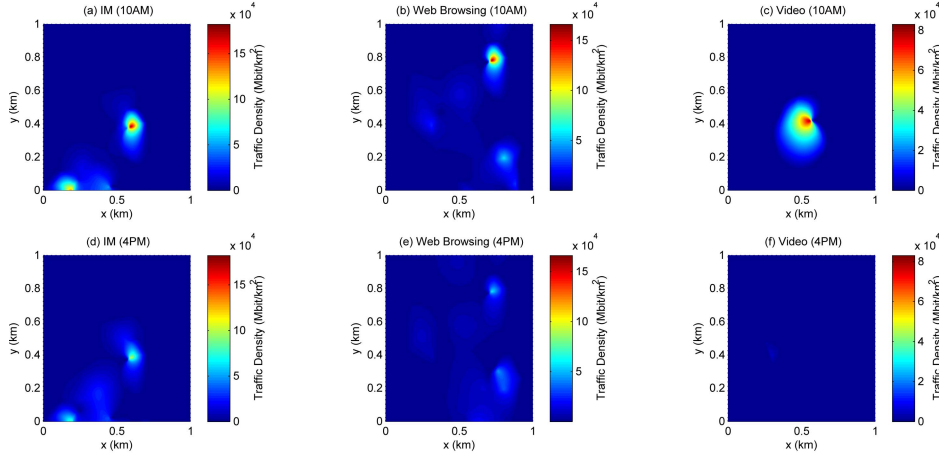


Fig. 6. Application-level cellular network traffic density in 10AM and 4PM in randomly selected dense urban areas for three service types of applications. The area for IM, Web Browsing, and Video contains 23, 39, 35 active cells, respectively.

Therefore, benefiting from the substantial body of works towards α -stable model based linear prediction [29], [35], coarse prediction results can be achieved by computing linear prediction coefficients in terms of the least mean square error criterion, the minimum dispersion criterion, or the covariation orthogonal criterion [18]. Due to its simplicity and comparatively low variability, the covariation orthogonal criterion [18], [19] is chosen in this paper to demonstrate the α -stable based linear prediction performance.

Without loss of generality, assume that there exist N cells in the area of interest. For a cell $i \in N$ with a known n -length traffic vector $\mathbf{x}^{(i)} = (x^{(i)}(1), \dots, x^{(i)}(n))$ in α -stable models-based predicted traffic vector $\hat{\mathbf{x}}_\alpha = (\hat{x}_\alpha^{(1)}, \dots)$ is approximated by

$$\tilde{x}_\alpha^{(i)} = \sum_{j=1}^m a^{(i)}(j)x^{(i)}(n+1-j), \quad (6)$$

with $1 < m \leq n$, where $\mathbf{a}^{(i)} = (a^{(i)}(1), \dots, a^{(i)}(m))$ denotes the prediction coefficients by α -stable models-based linear prediction algorithms. For example, in order to make the 1-step-ahead linear prediction $\tilde{x}_\alpha^{(i)}$ covariation orthogonal to $x^{(i)}(t), \forall t \in \{1, \dots, n\}$, coefficient $a^{(i)}(h), \forall h \in \{1, \dots, m\}$ should be given as [35]

$$\begin{aligned} a^{(i)}(h) &= \sum_{l=1}^m \left[\sum_{j=\max(h,l)}^n x^{(i)}(j-l+1)x^{(i)}(j-h+1) \right]^{<\alpha-1>} \\ &\quad \times \sum_{j=l+k}^n x^{(i)}(j) \left(x^{(i)}(j-k-l+1) \right)^{<\alpha-1>}. \end{aligned}$$

Here, the signed-power $v^{<\alpha-1>} = |v|^{\alpha-1} \text{sgn}(v)$. For simplicity of representation, the terminology “ $(n = 36, m = 10, k = 1)$ -linear prediction” is used to denote a prediction method, which firstly utilizes $n = 36$ consecutive traffic records in one randomly selected cell, then calculates $m = 10$ prediction coefficients, and finally predicts the traffic value at the next (i.e., $k = 1$) moment.

- Noise component. For any prediction algorithm, there dooms to exist some prediction error. Therefore, final traffic prediction vector $\hat{\mathbf{x}}_p$ is approximated by $\hat{\mathbf{x}}_\alpha$ plus Gaussian noise \mathbf{z} .⁷ Combining the temporal modeling and noise components, $\hat{\mathbf{x}}_p$ could be achieved by

$$\begin{aligned} \min_{\hat{\mathbf{x}}_p, \tilde{\mathbf{x}}_\alpha, \mathbf{z}} \quad & \|\hat{\mathbf{x}}_\alpha - \tilde{\mathbf{x}}_\alpha\|_2^2 + \lambda_1 \|\mathbf{z}\|_2^2, \\ \text{s.t.} \quad & \hat{\mathbf{x}}_p = \hat{\mathbf{x}}_\alpha + \mathbf{z}, \end{aligned} \quad (7)$$

$$\tilde{\mathbf{x}}_\alpha = \left(\tilde{x}_\alpha^{(1)}, \dots, \tilde{x}_\alpha^{(N)} \right), \quad (8)$$

$$\begin{aligned} \tilde{x}_\alpha^{(i)} &= \sum_{j=1}^m a^{(i)}(j)x^{(i)}(n+1-j), \\ \forall i \in \{1, \dots, N\}. \end{aligned} \quad (9)$$

For simplicity of representation, we omit constraints in Eq. (8) and Eq. (9) in the following statements.

- Spatial sparse component. In Section III-B, application-level traffic is shown to exhibit the spatial sparsity. Therefore, $\hat{\mathbf{x}}_p$ could be further refined by minimizing the gap between $\hat{\mathbf{x}}_p$ and a sparse linear combination (i.e., $\mathbf{s} \in \mathcal{R}^{K \times 1}$) of a dictionary $\mathbf{D} \in \mathcal{R}^{N \times K}$, namely

$$\min_{\hat{\mathbf{x}}_p, \mathbf{D}, \mathbf{s}} \|\hat{\mathbf{x}}_p - \mathbf{D}\mathbf{s}\|_2^2, \quad \text{s.t.} \quad \|\mathbf{s}\|_0 \leq \epsilon. \quad (10)$$

Notably, in Fig. 6, we observe sparse application-level cellular network traffic density. In other words, there merely exist few traffic spots with significantly large traffic volume. On the other hand, in the area of sparse representation, a l_0 -norm, which counts the number of

⁷There are two reasons leading to the assumption that noise is Gaussian distributed. Firstly, Gaussian distributed noise is widely used to characterize the fitting error between models and practical data. Secondly, we have conducted an experiment to examine the prediction performance of a simple $(n = 36, m = 10, k = 1)$ -linear prediction procedure and found that the prediction procedure could well predict the traffic trend. However, there would exist some gap between the real traffic trace and the predicted one. But, Fig. 7 indicates that the prediction error can be approximated by the Gaussian distribution. The K-S test further shows the GoF statistics for the IM, web browsing and video services are 0.0319, 0.0657 and 0.0437, respectively, smaller than the 95%-confidence threshold (i.e., 0.3507). So, we model the prediction error by Gaussian distribution.

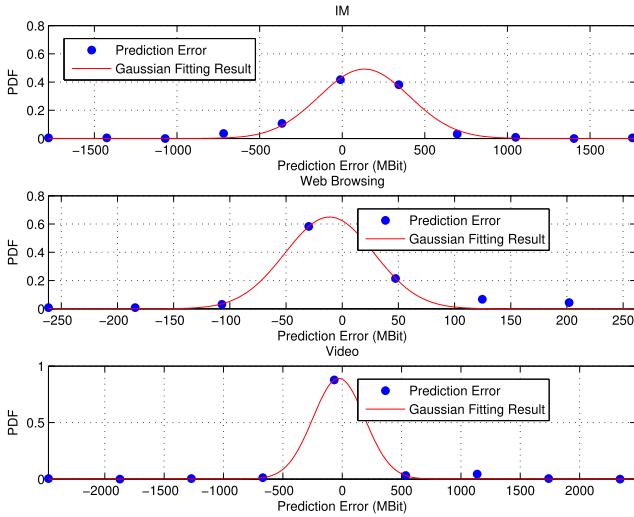


Fig. 7. Result by fitting the prediction error to a Gaussian distribution, after an α -stable model-based (36, 10, 1)-linear prediction method.

nonzero elements in the vector, is often used to characterize the sparse property. Therefore, in Eq. (10), we use a l_0 -norm to add the sparse constraint to the final optimization problem. Moreover, the exact representation of the dictionary, which the previous sparsity analyses do not mention, remains a problem and would be solved later.

Therefore, it is natural to consider the original dataset as a mixture of these effects and propose a new framework to combine these two components together to get a superior forecasting performance.

In order to capture the temporal α -stable modeled variations while keeping the spatial sparsity, a new framework is proposed as follows:

$$\begin{aligned} \min_{\hat{\mathbf{x}}_p, \hat{\mathbf{x}}_\alpha, \mathbf{z}, \mathbf{D}, \mathbf{s}} \quad & \|\hat{\mathbf{x}}_\alpha - \tilde{\mathbf{x}}_\alpha\|_2^2 + \lambda_1 \|\mathbf{z}\|_2^2 + \lambda_2 \|\hat{\mathbf{x}}_p - \mathbf{D}\mathbf{s}\|_2^2, \\ \text{s.t.} \quad & \hat{\mathbf{x}}_p = \hat{\mathbf{x}}_\alpha + \mathbf{z}, \quad \|\mathbf{s}\|_0 \leq \epsilon. \end{aligned}$$

Due to the nonconvexity of l_0 -norm, the constraints in Eq. (11) are not directly tractable. Thanks to the sparsity methods discussed in Section II-B, an l_1 -norm relaxation is employed to make the problem convex while still preserving the sparsity property [47]. Therefore, Eq. (11) can be reformulated as

$$\begin{aligned} \min_{\hat{\mathbf{x}}_p, \hat{\mathbf{x}}_\alpha, \mathbf{z}, \mathbf{D}, \mathbf{s}} \quad & \|\hat{\mathbf{x}}_\alpha - \tilde{\mathbf{x}}_\alpha\|_2^2 + \lambda_1 \|\mathbf{z}\|_2^2 + \lambda_2 \|\hat{\mathbf{x}}_p - \mathbf{D}\mathbf{s}\|_2^2, \\ \text{s.t.} \quad & \hat{\mathbf{x}}_p = \hat{\mathbf{x}}_\alpha + \mathbf{z}, \quad \|\mathbf{s}\|_1 \leq \epsilon. \end{aligned}$$

where ϵ is a predefined constraint, similar to ϵ .

Remark 5: This proposed framework integrates the temporal modeling and spatial correlation together. Moreover, by adjusting λ_1 and λ_2 to some extreme values, it's easy to show that the framework in Eq. (11) is closely tied to some typical methods in other references.

- If λ_1 and λ_2 are extremely small, the framework is simplified to a simple α -stable linear prediction method [18], [19].
- If λ_2 is extremely large, the spatial sparsity factor dominates in the framework [23].

B. Optimization Algorithm

In order to optimize the generalized framework, we first reformulate Eq. (11) by taking advantage of the augmented Lagrangian function [48] and then develop an alternating direction method (ADM) [12] to solve it. Specifically, the corresponding augmented Lagrangian function can be formulated as

$$\begin{aligned} \mathcal{L}(\hat{\mathbf{x}}_p, \hat{\mathbf{x}}_\alpha, \mathbf{z}, \mathbf{D}, \mathbf{s}, \mathbf{m}, \gamma, \eta) \\ \triangleq \|\hat{\mathbf{x}}_\alpha - \tilde{\mathbf{x}}_\alpha\|_2^2 + \lambda_1 \|\mathbf{z}\|_2^2 + \lambda_2 \|\hat{\mathbf{x}}_p - \mathbf{D}\mathbf{s}\|_2^2 \\ + \langle \mathbf{m}, \hat{\mathbf{x}}_p - \hat{\mathbf{x}}_\alpha - \mathbf{z} \rangle \end{aligned} \quad (11)$$

$$+ \gamma \cdot \|\mathbf{s}\|_1 \quad (12)$$

$$+ \eta \cdot \|\hat{\mathbf{x}}_p - \hat{\mathbf{x}}_\alpha - \mathbf{z}\|_2^2. \quad (13)$$

Besides, \mathbf{m} and γ are the Lagrangian multipliers, while η is a factor for the penalty term. Essentially, the augmented Lagrangian function includes the original objective, two Lagrange multiplier terms (i.e., Eq. (11) and Eq. (12)), and one penalty term converted from the equality constraint (i.e., Eq. (13)). Specifically, introducing Lagrange multipliers conveniently converts an optimization problem with equality constraints into an unconstrained one. Moreover, for any optimal solution that minimizes the (augmented) Lagrangian function, the partial derivatives with respect to the Lagrange multipliers must be zero [49]. Additionally, the penalty terms enforce the original equality constraints. Consequently, the original equality constraints are satisfied. Besides, by including Lagrange multiplier terms as well as the penalty terms, it's not necessary to iteratively increase η to ∞ to solve the original constrained problem, thereby avoiding ill-conditioning [48].

The ADM algorithm progresses in an iterative manner. During each iteration, we alternate among the optimization of the augmented function by varying each one of $(\hat{\mathbf{x}}_p, \hat{\mathbf{x}}_\alpha, \mathbf{z}, \mathbf{D}, \mathbf{s}, \mathbf{m}, \gamma, \eta)$ while fixing the other variables. Specifically, the ADM algorithm involves the following steps:

- 1) Find $\hat{\mathbf{x}}_\alpha$ to minimize the augmented Lagrangian function $\mathcal{L}(\hat{\mathbf{x}}_p, \hat{\mathbf{x}}_\alpha, \mathbf{z}, \mathbf{D}, \mathbf{s}, \mathbf{m}, \gamma, \eta)$ with other variables fixed. Removing the fixed items, the objective turns into

$$\begin{aligned} \arg \min_{\hat{\mathbf{x}}_\alpha} \quad & \|\hat{\mathbf{x}}_\alpha - \tilde{\mathbf{x}}_\alpha\|_2^2 \\ & + \langle \mathbf{m}, \hat{\mathbf{x}}_p - \hat{\mathbf{x}}_\alpha - \mathbf{z} \rangle + \eta \cdot \|\hat{\mathbf{x}}_p - \hat{\mathbf{x}}_\alpha - \mathbf{z}\|_2^2, \end{aligned}$$

which can be further reformulated as

$$\arg \min_{\hat{\mathbf{x}}_\alpha} \frac{1}{\eta} \cdot \|\hat{\mathbf{x}}_\alpha - \tilde{\mathbf{x}}_\alpha\|_2^2 + \|\hat{\mathbf{x}}_\alpha - (\hat{\mathbf{x}}_p - \mathbf{z} + \frac{\mathbf{m}}{2\eta})\|_2^2. \quad (14)$$

Letting $\mathbf{J}_{\hat{\mathbf{x}}_\alpha} = \hat{\mathbf{x}}_p - \mathbf{z} + \frac{\mathbf{m}}{2\eta}$ and setting the gradient of the objective function in Eq. (14) to be zero, it yields

$$\hat{\mathbf{x}}_\alpha = \frac{1}{\eta + 1} \cdot (\tilde{\mathbf{x}}_\alpha + \eta \cdot \mathbf{J}_{\hat{\mathbf{x}}_\alpha}). \quad (15)$$

- 2) Find \mathbf{z} to minimize the augmented Lagrangian function $\mathcal{L}(\hat{\mathbf{x}}_p, \hat{\mathbf{x}}_\alpha, \mathbf{z}, \mathbf{D}, \mathbf{s}, \mathbf{m}, \gamma, \eta)$ with other variables fixed. The corresponding mathematical formula is

$$\begin{aligned} \arg \min_{\mathbf{z}} \quad & \lambda_1 \|\mathbf{z}\|_2^2 + \langle \mathbf{m}, \hat{\mathbf{x}}_p - \hat{\mathbf{x}}_\alpha - \mathbf{z} \rangle \\ & + \eta \cdot \|\hat{\mathbf{x}}_p - \hat{\mathbf{x}}_\alpha - \mathbf{z}\|_2^2. \end{aligned}$$

Similarly, it can be reformulated as

$$\arg \min_{\hat{\mathbf{x}}_a} \frac{\lambda_1}{\eta} \cdot \|\mathbf{z}\|_2^2 + \|\mathbf{z} - (\hat{\mathbf{x}}_p - \hat{\mathbf{x}}_a + \frac{\mathbf{m}}{2\eta})\|_2^2. \quad (16)$$

Letting $\mathbf{J}_z = \hat{\mathbf{x}}_p - \hat{\mathbf{x}}_a + \frac{\mathbf{m}}{2\eta}$ and setting the gradient of the objective function in Eq. (16) to be zero, it yields

$$\mathbf{z} = \frac{1}{\lambda_1/\eta + 1} \cdot \mathbf{J}_z. \quad (17)$$

- 3) Find $\hat{\mathbf{x}}_p$ to minimize the augmented Lagrangian function $\mathcal{L}(\hat{\mathbf{x}}_p, \hat{\mathbf{x}}_a, \mathbf{z}, \mathbf{D}, \mathbf{s}, \mathbf{m}, \gamma, \eta)$ with other variables fixed. It gives

$$\arg \min_{\hat{\mathbf{x}}_p} \lambda_2 \|\hat{\mathbf{x}}_p - \mathbf{D}\mathbf{s}\|_2^2 + \langle \mathbf{m}, \hat{\mathbf{x}}_p - \hat{\mathbf{x}}_a - \mathbf{z} \rangle + \eta \cdot \|\hat{\mathbf{x}}_p - \hat{\mathbf{x}}_a - \mathbf{z}\|_2^2.$$

That is

$$\arg \min_{\hat{\mathbf{x}}_p} \frac{\lambda_2}{\eta} \cdot \|\hat{\mathbf{x}}_p - \mathbf{D}\mathbf{s}\|_2^2 + \|\hat{\mathbf{x}}_p - (\hat{\mathbf{x}}_a + \mathbf{z} - \frac{\mathbf{m}}{2\eta})\|_2^2. \quad (18)$$

Define $\mathbf{J}_{\hat{\mathbf{x}}_p} = \hat{\mathbf{x}}_a + \mathbf{z} - \frac{\mathbf{m}}{2\eta}$ and set the corresponding gradient in Eq. (18) to be zero. It becomes

$$\hat{\mathbf{x}}_p = 1 / \left(\frac{\lambda_2}{\eta} + 1 \right) \cdot \left(\frac{\lambda_2}{\eta} \mathbf{D}\mathbf{s} + \mathbf{J}_{\hat{\mathbf{x}}_p} \right). \quad (19)$$

- 4) Find \mathbf{D} and \mathbf{s} to minimize the augmented Lagrangian function $\mathcal{L}(\hat{\mathbf{x}}_p, \hat{\mathbf{x}}_a, \mathbf{z}, \mathbf{D}, \mathbf{s}, \mathbf{m}, \gamma, \eta)$ with other variables fixed. In fact, the objective function turns into

$$\arg \min_{\mathbf{D}, \mathbf{s}} \lambda_2 \|\hat{\mathbf{x}}_p - \mathbf{D}\mathbf{s}\|_2^2 + \gamma \cdot \|\mathbf{s}\|_1. \quad (20)$$

Obviously, this optimization problem in Eq. (20) is exactly the sparse signal recovery problem without the dictionary a priori in Section II-B. Inspired by the dictionary learning methodology (namely the means to learn the dictionary or basis sets of large-scale data) in [27], the corresponding solution alternatively determines \mathbf{D} and \mathbf{s} and thus involves two sub-procedures, namely online learning algorithm [27] and LARS-lasso algorithm [50]. Algorithm 1 provides the skeleton of this solution.

In order to update the dictionary in Eq. (22), the proposed sparse signal recovery algorithm utilizes the concept of stochastic approximation, which is firstly introduced and mathematically proved convergent to a stationary point in [27].

On the other hand, based on the learned dictionary, the concerted effort to recover a sparse signal could be exploited. As mentioned above, the well known LARS-lasso algorithm [50], which is a forward stagewise regression algorithm and gradually finds the most suitable solution along a equiangular path among the already known predictors, is used here to solve the problem in Eq. (21). Meanwhile, it is worthwhile to note that other compressive sensing algorithms [39] could also be used here.

- 5) Update estimate for the Lagrangian multiplier \mathbf{m} according to steepest gradient descent method [51], namely

Algorithm 1 The Sparse Signal Recovery Algorithm Without a Predetermined Dictionary

Require: the dictionary \mathbf{D} as an input dictionary $\mathbf{D}^{(0)}$ (which could be the dictionary learned in last calling this Algorithm), the number of iterations for learning a dictionary as T , two auxiliary matrices $\mathbf{A}^{(0)} \in \mathcal{R}^{K \times K}$ and $\mathbf{B}^{(0)} \in \mathcal{R}^{K \times K}$ with all elements therein equaling zero.

1: **for** $t = 1$ to T **do**

- 2: Sparse coding: computing $\mathbf{s}^{(t)}$ using LARS-Lasso algorithm [50] to obtain

$$\mathbf{s}^{(t)} = \arg \min_{\mathbf{s}} \lambda_2 \|\hat{\mathbf{x}}_p - \mathbf{D}^{(t-1)}\mathbf{s}\|_2^2 + \gamma \cdot \|\mathbf{s}\|_1. \quad (21)$$

- 3: Update $\mathbf{A}^{(t)}$ according to

$$\mathbf{A}^{(t)} \leftarrow \mathbf{A}^{(t-1)} + \mathbf{s}^{(t)}(\mathbf{s}^{(t)})^T.$$

- 4: Update $\mathbf{B}^{(t)}$ according to

$$\mathbf{B}^{(t)} \leftarrow \mathbf{B}^{(t-1)} + \hat{\mathbf{x}}_p(\mathbf{s}^{(t)})^T.$$

- 5: Dictionary Update: computing $\mathbf{D}^{(t)}$ online learning algorithm [27] to obtain

$$\begin{aligned} \mathbf{D}^{(t)} &= \arg \min_{\mathbf{D}} \lambda_2 \|\hat{\mathbf{x}}_p - \mathbf{D}\mathbf{s}^{(t)}\|_2^2 + \gamma \cdot \|\mathbf{s}^{(t)}\|_1 \\ &= \arg \min_{\mathbf{D}} \text{Tr}(\mathbf{D}^T \mathbf{D} \mathbf{A}^{(t)}) - 2\text{Tr}(\mathbf{D}^T \mathbf{B}^{(t)}). \end{aligned} \quad (22)$$

6: **end for**

- 7: **return** the learned dictionary $\mathbf{D}^{(t)}$ and the sparse coding vector $\mathbf{s}^{(t)}$.
-

$\mathbf{m} \leftarrow \mathbf{m} + \eta \cdot (\hat{\mathbf{x}}_p - \hat{\mathbf{x}}_a - \mathbf{z})$. Similarly, update estimate γ by $\gamma \leftarrow \gamma + \eta \cdot \|\mathbf{s}\|_1$.

- 6) Update $\eta \leftarrow \eta \cdot \rho$.

In Algorithm 2, we summarize the steps during each iteration. Notably, without loss of generality, consider a known traffic vector $\mathbf{x}(0, \dots, t)$ of a given cell at different moments $(0, \dots, t)$. Then, we could estimate the α -stable related parameters according to maximum likelihood methods, quantile methods, or sample characteristic function methods in [34], [35]. Afterwards, we could conduct Algorithm 2 to predict the traffic volume at moment $t + 1$. Similarly, we need to estimate the α -stable related parameters according to methods in [34] and [35], in terms of the traffic vector $\mathbf{x}(0, \dots, t + 1)$, and perform Algorithm 2 to predict the traffic volume at moment $t + 2$. It can be observed that, compared to Algorithm 1, which is an application of the lines in [27], Algorithm 2 consists of some additional iterative procedures to procure the parameters unknown a priori. Besides, most steps involved in Algorithm 2 are deterministic vector computations and thus computationally efficient. Therefore, the whole framework could effectively yield the traffic forecasting results.

V. PERFORMANCE EVALUATION

We validate the prediction accuracy improvement of our proposed framework in Algorithm 2 relying on the practical traffic dataset. Specifically, we choose the traffic load records of these three service types of applications generated in

Algorithm 2 The Dictionary Learning-Based Alternating Direction Method

Require: $\hat{x}_p, \hat{x}_a, z, D, s, m, \gamma, \eta$ according to $\hat{x}_p^{(0)}, \hat{x}_a^{(0)}, z^{(0)}, D^{(0)}, s^{(0)}, m^{(0)}, \gamma^{(0)}, \eta^{(0)}$, and the number of iterations T . Compute \tilde{x}_a according to α -stable model based linear prediction algorithms [29], [35].

- 1: **for** $t = 1$ to T **do**
- 2: Update \hat{x}_a according to $\hat{x}_a^{(t)} \leftarrow \frac{1}{\eta^{(t-1)}+1} \cdot \left(\tilde{x}_a + \eta^{(t-1)} \cdot \left(\hat{x}_p^{(t-1)} - z^{(t-1)} + \frac{m^{(t-1)}}{2\eta^{(t-1)}} \right) \right)$.
- 3: Update z according to $z^{(t)} \leftarrow \frac{1}{\lambda_1/\eta^{(t-1)}+1} \cdot \left(\hat{x}_p^{(t-1)} - \hat{x}_a^{(t)} + \frac{m^{(t-1)}}{2\eta^{(t-1)}} \right)$.
- 4: Update \hat{x}_p according to $\hat{x}_p^{(t)} \leftarrow \frac{1}{\left(\frac{\lambda_2}{\eta^{(t-1)}} + 1 \right)} \cdot \left(\frac{\lambda_2}{\eta^{(t-1)}} D^{(t-1)} s^{(t-1)} + \hat{x}_a^{(t)} + z^{(t)} - \frac{m^{(t-1)}}{2\eta^{(t-1)}} \right)$.
- 5: Update D and s according to sparse signal recovery algorithm (i.e., Algorithm 1). In particular, use two sub-procedures namely online learning algorithm [27] and LARS-lasso algorithm [50] to update D and s , respectively.
- 6: Update m according to $m^{(t)} \leftarrow m^{(t-1)} + \eta^{(t-1)} \cdot \left(\hat{x}_p^{(t)} - \hat{x}_a^{(t)} - z^{(t)} \right)$.
- 7: Update γ by $\gamma^{(t)} \leftarrow \gamma^{(t-1)} + \eta^{(t-1)} \cdot \|s^{(t)}\|_1$.
- 8: Update η by $\eta^{(t)} \leftarrow \eta^{(t-1)} \cdot \rho$, here ρ is an iteration ratio.
- 9: **end for**
- 10: **return** the predicted traffic vector \hat{x}_p .

113 cells within a randomly selected region from Dataset 1. Moreover, we intentionally divide the traffic dataset into two part. One is used to learn and distill the parameters related to traffic characteristics, and the other part is to conduct the experiments to verify and validate the accuracy of the proposed framework in Algorithm 2. Specifically, we compare our prediction \hat{x}_p with the ground truth x in terms of the normalized mean absolute error (NMAE) [12], which is defined as

$$\text{NMAE} = \frac{\sum_{i=1}^N |\hat{x}_p(i) - x(i)|}{\sum_{i=1}^N |x(i)|}. \quad (23)$$

As described in Algorithm 2, most of the parameters could be set easily and tuned dynamically within the framework. Therefore, we can benefit from this advantage and only need to examine the performance impact of few parameters, namely $\lambda_1, \lambda_2, \gamma$ and η , by dynamically adjusting them. By default, we set $\lambda_1 = 10, \lambda_2 = 1, \gamma = 1$ and $\eta = 10^{-4}$, and the number of iterations in Algorithm 2 and sparse signal recovery algorithm (i.e., Algorithm 1) to be 20 and 3, respectively. Besides, we impose no prior constraints on D, s , and z , and set them as zero vectors.

Fig. 8 gives the performance of our proposed framework in terms of NMAE, by taking advantage of the (36,10,1)-linear prediction algorithm in Section IV-A to provide the ‘‘coarse’’ prediction results \tilde{x}_a . In other words, we would exploit traffic records in the last three hours to train the parameters of α -stable models and predict traffic loads in the next 5 minutes. In order to provide a more comprehensive comparison,

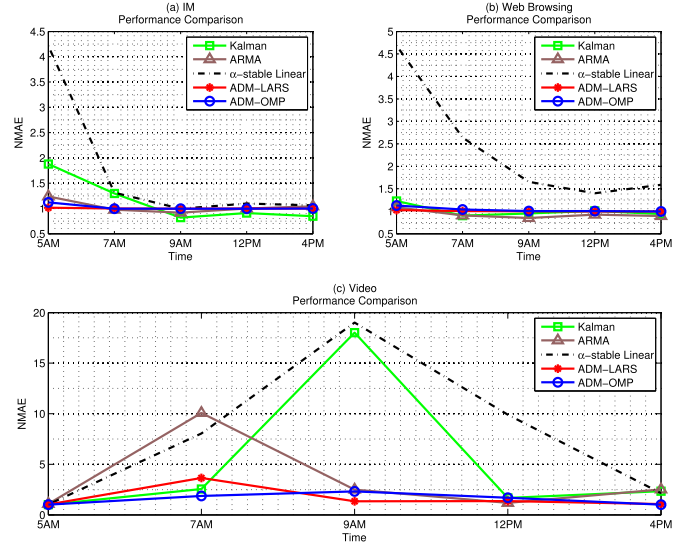


Fig. 8. Performance comparison between the proposed ADM framework with different sparse signal recovery algorithms (i.e., LARS-Lasso and OMP), and the α -stable model based (36,10,1)-linear prediction algorithm.

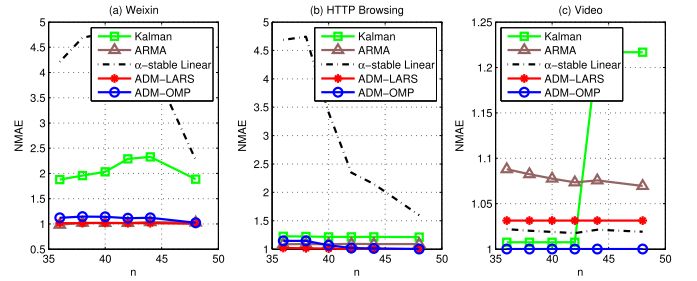


Fig. 9. Performance variations with respect to the training data length n for the proposed ADM framework with LARS-Lasso algorithm.

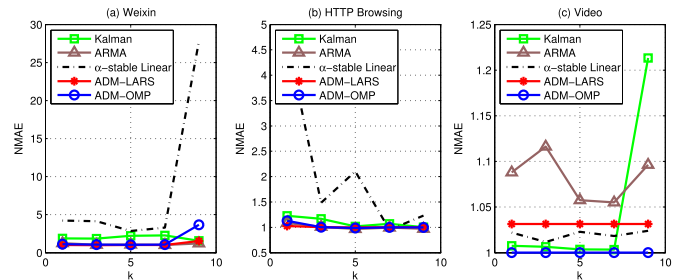


Fig. 10. Performance variations with respect to the forecasting time lag k for the proposed ADM framework with LARS-Lasso algorithm.

the simulations run in both busy moments (i.e., 9AM, 12PM, and 4PM) and idle ones (i.e., 7AM and 9PM) of one day. We first examine the corresponding performance improvement of the proposed ADM framework with different sparse signal recovery algorithm (i.e., LARS-Lasso algorithm [50] and OMP algorithm [39]). It can be observed that in most cases, different sparse signal recovery algorithm has little impact on the prediction accuracy. Therefore, the applications of the proposed framework could pay little attention to the involved sparsity methods. Afterwards, we can find that the proposed framework significantly outperforms the classical α -stable model based (36,10,1)-linear prediction algorithm

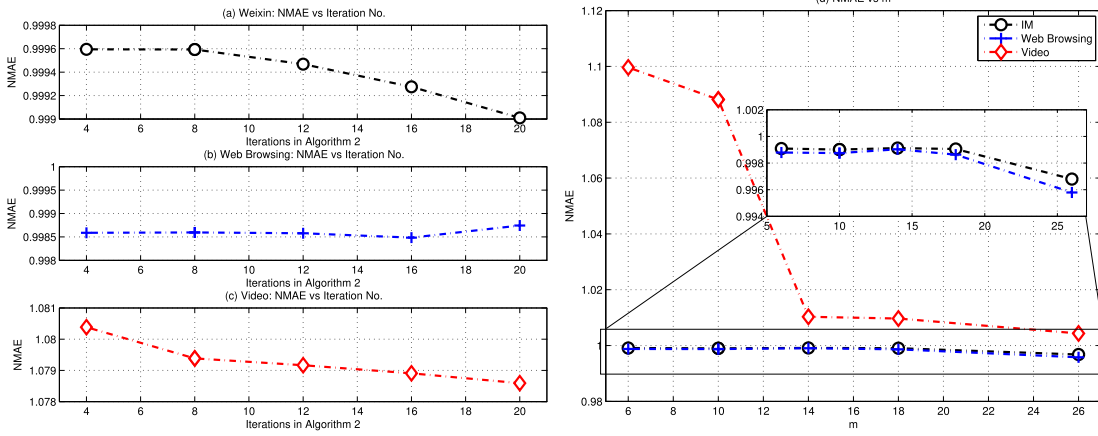


Fig. 11. Performance variations with respect to (a)~(c) the number of iterations and (d) the number of prediction coefficients m in Algorithm 2 for the proposed ADM framework with LARS-Lasso algorithm.

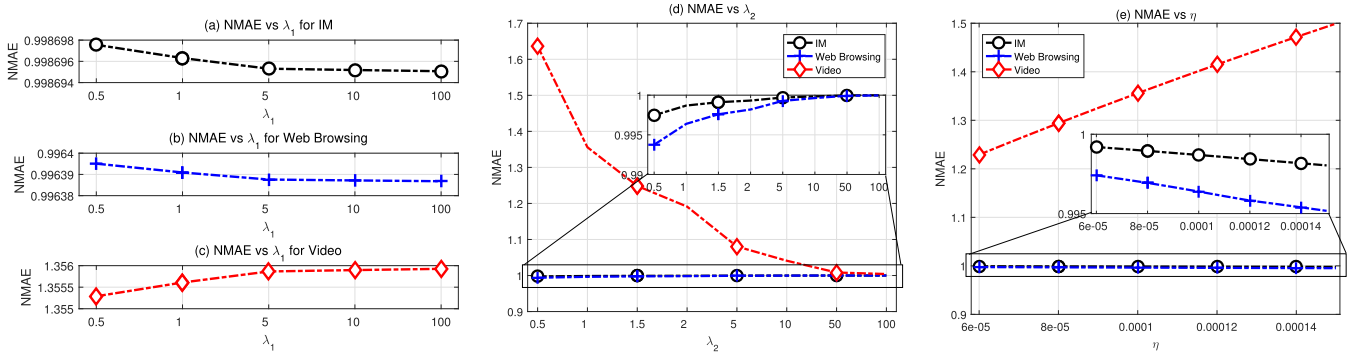


Fig. 12. Performance variations with respect to λ_1 , λ_2 , and η , for the proposed ADM framework with LARS-Lasso algorithm.

(the “ α -stable linear” curve in Fig. 8). In particular, the NMAE of the proposed framework can be as 12% small (e.g., prediction for 12PM video traffic) as that for the classical linear algorithm. This performance improvement can be interpreted as the gain by exploiting the embedded sparsity in traffic and taking account of the originally existing prediction error of linear prediction. Furthermore, we also compare the proposed framework with ARMA and Kalman filtering algorithms and show that our solution can achieve competitive performance for IM and web browsing services and yield far more stable and superior performance for the video service. As shown in Table IV, the α value for the video service is different from those of the other services and less than 1, so the video traffic with distinct characteristics makes ARMA and Kalman filtering algorithms less effective. We can confidently reach the conclusion that our proposed framework offers a unified solution for the application-level traffic modeling and prediction with appealing accuracy.

Afterwards, we further evaluate the impact of the training length n and the forecasting time lag k on the prediction accuracy, and give the related results in Fig. 9 and Fig. 10, respectively. Fig. 9 shows the increase of n contributes to improving the prediction accuracy for all types of applications especially the video service, which is consistent with our intuition. Besides, when n varies, the proposed framework

with LARS and OMP algorithms demonstrates more robust prediction accuracy while the other algorithms might yield inferior performance for some values of n . Fig. 10 presents that with the increase of the forecasting time lag k , the prediction accuracy demonstrates a increasing trend, which also matches our intuition. Again, for different types of services and different k , the proposed framework possesses the strongest robustness.

Next, we further evaluate the performance of our proposed ADM framework with LARS-Lasso algorithm and provide more detailed sensitivity analyses. Fig. 11 depicts the performance variations with respect to the number of iterations in Algorithm 2 and the number of prediction coefficients m , respectively. From Fig. 11(a)~(c), the loss in prediction accuracy is rather small when the number of iterations decreases from 20 to 4. Hence, if we initialized the prediction process with 20 iterations, we can stop the iterative process whenever the results between two consecutive iterations become sufficiently small, so as to reduce the computational complexity. Fig. 11(d) shows that similar to the case in Fig. 9, the increase of m also contributes to improving the prediction accuracy for all types of applications especially the video service. Fig. 12(a), Fig. 12(b) and Fig. 12(c) show that the prediction accuracy nearly stays the same irrespective of λ_1 . This means that the noise component has limited contribution

to the corresponding performance. It also implies that the choice of λ_1 could be flexible when we apply the framework in practice. Fig. 12(d) demonstrates that the influence of λ_2 is comparatively more obvious and even diverges for different service types. Specifically, a larger λ_2 has a slightly negative impact on predicting the traffic loads for IM and web browsing service, but it contributes to the prediction of video service. As stated in Section III-B, video service demonstrates the strongest sparsity. Hence, by increasing λ_2 , it implies to put more emphasis on the importance of sparsity and results in a better performance for video service. It's worthwhile to note here that, in Eq. (12), λ_2 and γ are coupled together as well and should have inverse performance impact. Therefore, due to the space limitation, the performance impact of γ is omitted here. Fig. 12(e) depicts the performance variation with respect to η , which is similar to that with respect to λ_2 . But, a larger η has a positive impact on predicting the traffic loads for IM and web browsing service, but it degrades the prediction performance of video service. This phenomenon is potentially originated from the very distinct characteristics of these three services types (e.g., different α -stable models' parameters and different sparsity representation) and needs a further careful investigation. However, it safely comes to the conclusion that the proposed framework provides a superior and robust performance than the classical linear algorithm.

VI. CONCLUSION AND FUTURE DIRECTION

In this paper, we collected the application-level traffic data from one operator in China. With the aid of this practical traffic data, we confirmed several important statistical characteristics like temporally α -stable modeled property and spatial sparsity. Afterwards, we proposed a traffic prediction framework, which takes advantage of the already known traffic knowledge to distill the parameters related to aforementioned traffic characteristics and forecasts future traffic results bearing the same characteristics. We also developed a dictionary learning-based alternating direction method to solve the framework and manifested the effectiveness and robustness of our algorithm through extensive simulation results.

On the other hand, there still exist some issues to be addressed. The biggest challenge for the application-level traffic modeling prediction lies in that as new types of applications or services continually emerge and blossom, whether the unveiled characteristics still hold? Furthermore, it is still interesting to investigate how to leverage the additional information (e.g., inter-service relevancy) to further optimize the proposed framework and improve the prediction accuracy.

REFERENCES

- [1] *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2012–2017*, Cisco, San José, CA, USA, Feb. 2013. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html
- [2] R. Li, Z. Zhao, X. Zhou, and H. Zhang, "Energy savings scheme in radio access networks via compressive sensing-based traffic load prediction," *Trans. Emerg. Telecommun. Technol.*, vol. 25, no. 4, pp. 468–478, Apr. 2014.
- [3] U. Paul, M. Buddhikot, and S. Das, "Opportunistic traffic scheduling in cellular data networks," in *Proc. IEEE DySPAN*, Bellevue, WA, USA, Oct. 2012, pp. 339–348.
- [4] P. Romirer-Maierhofer, M. Schiavone, and A. D'Alconzo, "Device-specific traffic characterization for root cause analysis in cellular networks," in *Traffic Monitoring and Analysis* (Lecture Notes in Computer Science), vol. 9053, M. Steiner, P. Barlet-Ros, and O. Bonaventure, Eds. Cham, Switzerland: Springer, Apr. 2015, pp. 64–78. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-319-17172-2_5
- [5] Z. Niu, Y. Wu, J. Gong, and Z. Yang, "Cell zooming for cost-efficient green cellular networks," *IEEE Commun. Mag.*, vol. 48, no. 11, pp. 74–79, Nov. 2010.
- [6] Z. Niu, "TANGO: Traffic-aware network planning and green operation," *IEEE Wireless Commun.*, vol. 18, no. 5, pp. 25–29, Oct. 2011.
- [7] R. Li, Z. Zhao, X. Zhou, J. Palicot, and H. Zhang, "The prediction analysis of cellular radio access network traffic: From entropy theory to networking practice," *IEEE Commun. Mag.*, vol. 52, no. 6, pp. 234–240, Jun. 2014.
- [8] N. Bui, M. Cesana, S. A. Hosseini, Q. Liao, I. Malanchini, and J. Widmer. (Jun. 2016). "Anticipatory networking in future generation mobile networks: A survey." [Online]. Available: <http://arxiv.org/abs/1606.00191>
- [9] R. G. Baraniuk, "Compressive sensing [lecture notes]," *IEEE Signal Process. Mag.*, vol. 24, no. 4, pp. 118–121, Jul. 2007. [Online]. Available: http://omni.isr.ist.utl.pt/~aguiar/CS_notes.pdf
- [10] J. Romberg and M. Wakin, "Compressed sensing: A tutorial," in *Proc. IEEE SSP Workshop*, Madison, WI, USA, Aug. 2007, pp. 1–128. [Online]. Available: <http://people.ee.duke.edu/~willett/SSP/Tutorials/ssp07-cs-tutorial.pdf>
- [11] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [12] Y.-C. Chen, L. Qiu, Y. Zhang, G. Xue, and Z. Hu, "Robust network compressive sensing," in *Proc. ACM Mobicom*, Maui, HI, USA, Sep. 2014, pp. 1–12.
- [13] *IEEE 802.16 Broadband Wireless Access Working Group*, IEEE Standard 802.16m Evaluation Methodology document, Jul. 2008. [Online]. Available: <http://ieee802.org/16>
- [14] B. Zhou, D. He, Z. Sun, and W. H. Ng, "Network traffic modeling and prediction with ARIMA/GARCH," in *Proc. HET-NETs Conf.*, Ilkley, U.K., Jul. 2005, pp. 1–10.
- [15] O. Cappe, E. Moulines, J. C. Pesquet, A. P. Petropulu, and X. Yang, "Long-range dependence and heavy-tail modeling for teletraffic data," *IEEE Signal Process. Mag.*, vol. 19, no. 3, pp. 14–27, May 2002.
- [16] F. Ashtiani, J. A. Salehi, and M. R. Aref, "Mobility modeling and analytical solution for spatial traffic distribution in wireless multimedia networks," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 10, pp. 1699–1709, Dec. 2003.
- [17] K. Tutschku and P. Tran-Gia, "Spatial traffic estimation and characterization for mobile communication network design," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 5, pp. 804–811, Jun. 1998.
- [18] L. Xiang, X. Ge, C. Liu, L. Shu, and C.-X. Wang, "A new hybrid network traffic prediction method," in *Proc. IEEE Globecom*, Miami, FL, USA, Dec. 2010, pp. 1–5.
- [19] X. Ge, S. Yu, W.-S. Yoon, and Y.-D. Kim, "A new prediction method of alpha-stable processes for self-similar traffic," in *Proc. IEEE Globecom*, Dallas, TX, USA, Nov./Dec. 2004, pp. 675–679.
- [20] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, and J. Wang, "Geospatial and temporal dynamics of application usage in cellular data networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 7, pp. 1369–1381, Jul. 2014. [Online]. Available: http://myweb.uiowa.edu/mshafiq/files/spatialApp_TMC.pdf
- [21] M. E. Crovella and A. Bestavros, "Self-similarity in world wide Web traffic: Evidence and possible causes," *IEEE/ACM Trans. Netw.*, vol. 5, no. 6, pp. 835–846, Dec. 1997.
- [22] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Trans. Netw.*, vol. 2, no. 1, pp. 1–15, Feb. 1994.
- [23] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and Internet traffic matrices," in *Proc. ACM SIGCOMM*, Barcelona, Spain, Aug. 2008, pp. 267–278.
- [24] A. Soule *et al.*, "Traffic matrices: Balancing measurements, inference and modeling," in *Proc. ACM SIGMETRICS*, Banff, AB, Canada, Jun. 2005, pp. 1–13.
- [25] M. C. Falvo, M. Gastaldi, A. Nardecchia, and A. Prudenzi, "Kalman filter for short-term load forecasting: An hourly predictor of municipal load," in *Proc. IASTED ASM*, Palma de Mallorca, Spain, Aug. 2007, pp. 364–369.

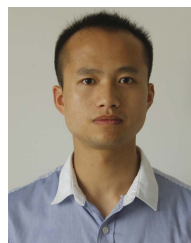
- [26] R. Li, Z. Zhao, Y. Wei, X. Zhou, and H. Zhang, "GM-PAB: A grid-based energy saving scheme with predicted traffic load guidance for cellular networks," in *Proc. IEEE ICC*, Ottawa, ON, Canada, Jun. 2012, pp. 1160–1164.
- [27] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, Mar. 2010.
- [28] U. Paul, L. Ortiz, S. R. Das, G. Fusco, and M. M. Buddhikot, "Learning probabilistic models of cellular network traffic with applications to resource management," in *Proc. IEEE DySPAN*, McLean, VA, USA, Apr. 2014, pp. 82–91.
- [29] J. B. Hill, "Minimum dispersion and unbiasedness: 'Best' linear predictors for stationary ARMA α -stable processes," Dept. Econ., Univ. Colorado Boulder, Boulder, CO, USA, Tech. Rep. 00-06, Sep. 2000.
- [30] A. Karasaris and D. Hatzinakos, "Network heavy traffic modeling using α -stable self-similar processes," *IEEE Trans. Commun.*, vol. 49, no. 7, pp. 1203–1214, Jul. 2001.
- [31] F. Qian, Z. Wang, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck, "Characterizing radio resource allocation for 3G networks," in *Proc. ACM SIGCOMM*, New York, NY, USA, May 2010, pp. 1–14.
- [32] F. P. Tso, J. Teng, W. Jia, and D. Xuan, "Mobility: A double-edged sword for HSPA networks: A large-scale test on hong kong mobile HSPA networks," in *Proc. ACM MobiHoc*, Sep. 2010, pp. 1–10.
- [33] G. Samorodnitsky, *Stable Non-Gaussian Random Processes: Stochastic Models With Infinite Variance*. New York, NY, USA: Chapman & Hall, Jun. 1994. [Online]. Available: <http://www.amazon.com/Stable-Non-Gaussian-Random-Processes-Stochastic/dp/0412051710>
- [34] J. R. Gallardo, D. Makrakis, and L. Orozco-Barbosa, "Use of alpha-stable self-similar stochastic processes for modeling traffic in broadband networks," *Proc. SPIE*, vol. 3530, pp. 281–296, Nov. 1998.
- [35] G. Xiaohu, Z. Guangxi, and Z. Yaoting, "On the testing for alpha-stable distributions of network traffic," *Comput. Commun.*, vol. 27, no. 5, pp. 447–457, Mar. 2004.
- [36] W. Song and W. Zhuang, "Resource reservation for self-similar data traffic in cellular/WLAN integrated mobile Hotspots," in *Proc. IEEE ICC*, Cape Town, South Africa, May 2010, pp. 1–5.
- [37] J. C.-I. Chuang and N. R. Sollenberger, "Spectrum resource allocation for wireless packet access with application to advanced cellular Internet service," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 6, pp. 820–829, Aug. 1998.
- [38] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, Aug. 1998.
- [39] Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. ACSSC*, Pacific Grove, CA, USA, Nov. 1993, pp. 40–44.
- [40] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
- [41] X. Zhou, Z. Zhao, R. Li, Y. Zhou, and H. Zhang, "The predictability of cellular networks traffic," in *Proc. IEEE ISCIT*, Gold Coast, QLD, Australia, Oct. 2012, pp. 973–978.
- [42] J. H. McCulloch, "Simple consistent estimators of stable distribution parameters," *Commun. Statist. Simul.*, vol. 15, no. 4, pp. 1109–1136, Jan. 1986.
- [43] M. Vidyasagar. (Sep. 2016). *Fitting Data to Distributions (Lecture 4)*. [Online]. Available: <http://www.utdallas.edu/~m.vidyasagar/Fall-2014/6303/Lect-4.pdf>
- [44] X. Zhou, Z. Zhao, R. Li, Y. Zhou, J. Palicot, and H. Zhang, "Understanding the nature of social mobile instant messaging in cellular networks," *IEEE Commun. Lett.*, vol. 18, no. 3, pp. 389–392, Mar. 2014.
- [45] A. N. Kolmogorov, K. L. Chung, and B. V. Gnedenko, *Limit Distributions for Sums of Independent Random Variables*, Reading, MA, USA: Addison-Wesley, 1968. [Online]. Available: https://openlibrary.org/books/OL19738039M/Limit_distributions_for_sums_of_independent_random_variables
- [46] D. Lee, S. Zhou, X. Zhong, Z. Niu, X. Zhou, and H. Zhang, "Spatial modeling of the traffic density in cellular networks," *IEEE Wireless Commun.*, vol. 21, no. 1, pp. 80–88, Feb. 2014.
- [47] R. Fang, T. Chen, and P. C. Sanelli, "Towards robust deconvolution of low-dose perfusion CT: Sparse perfusion deconvolution using online dictionary learning," *Med. Image Anal.*, vol. 17, no. 4, pp. 417–428, May 2013.
- [48] Wikipedia. (Oct. 2014). *Augmented Lagrangian Method*. [Online]. Available: http://en.wikipedia.org/w/index.php?title=Augmented_Lagrangian_method
- [49] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, Mar. 2004.
- [50] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Statist.*, vol. 32, no. 2, pp. 407–499, 2004.
- [51] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*. Cambridge, U.K.: Cambridge Univ. Press, 1998. [Online]. Available: <http://webdocs.cs.ualberta.ca/~sutton/book/ebook/>



Rongpeng Li received the B.E. degree (Hons.) from Xidian University, Xian, in 2010, and the Ph.D. degree from Zhejiang University, Hangzhou, China, in 2015. He was a Visiting Doctoral Student with Supélec, Rennes, France, in 2013, and an Intern Researcher with the China Mobile Research Institute, Beijing, China, in 2014. From 2015 to 2016, he was a Researcher with the Wireless Communication Laboratory, Huawei Technologies Co., Ltd., Shanghai, China. He is currently a Post-Doctoral Researcher with the College of Computer Science and Technologies and the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China. His research interests currently focus on resource allocation of cellular networks such as full duplex networks, applications of reinforcement learning, and analysis of cellular network data and he has authored/co-authored several papers in the related fields. He was granted by the National Post-Doctoral Program for Innovative Talents, which has a grant ratio of 13% in 2016. He serves as an Editor of the *China Communications*.



Zhifeng Zhao received the Ph.D. degree in communication and information system from the PLA University of Science and Technology, Nanjing, China, in 2002. He is currently an Associate Professor with the Department of Information Science and Electronic Engineering, Zhejiang University, China. His research area includes cognitive radio, wireless multi-hop networks such as *ad hoc*, mesh, WSN, wireless multimedia network, and green communications. He was the Symposium Co-Chair of the ChinaCom 2009 and 2010. He was the Technical Program Committee Co-Chair of the 10th IEEE International Symposium on Communication and Information Technology in 2010.



Jianchao Zheng received the B.S. degree in Communications Engineering, and the Ph.D. degree in communications and information systems from the of Communications Engineering, the PLA University of Science and Technology, Nanjing, China, in 2010 and 2016, respectively. He is currently an assistant professor in College of Communications Engineering, PLA University of Science and Technology. During 2015-2016, he was the visiting scholar at the Broadband Communications Research (BBCR) group, Department of Electrical and Computer Engineering, the University of Waterloo, Canada. His research interests focus on Interference mitigation techniques, learning theory, game theory, and optimization techniques. He has published several papers in international conferences and reputed journals in his research area.



Chengli Mei received the Ph.D. degree from Shanghai Jiaotong University, Shanghai, China. He is a Principal Research Engineer and the Director of the China Telecom Technology Innovation Center, Beijing, China. His research interests focus on the standardization and techniques of mobile communication networks, especially in the area of network evolution and service deployment.



Yueming Cai received the B.S. degree in physics from Xiamen University, Xiamen, China, in 1982, the M.S. degree in micro-electronics engineering and the Ph.D. degree in Communications and Information Systems from Southeast University, Nanjing, China, in 1988 and 1996 respectively. His current research interest includes cooperative communications, signal processing in communications, wireless sensor networks, and physical layer security.



Honggang Zhang is currently a Full Professor with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China. He is also an Honorary Visiting Professor with the University of York, York, U.K. He was the International Chair Professor of Excellence for the European University of Brittany and Supélec, France, from 2012 to 2014. He was a co-author and an editor of two books with the titles of *Cognitive Communications-Distributed Artificial Intelligence, Regulatory Policy and Economics, Implementation* (John Wiley & Sons) and *Green Communications: Theoretical Fundamentals, Algorithms and Applications* (CRC Press), respectively. He is currently active in the research on green communications. He served as the Chair of the Technical Committee on Cognitive Networks of the IEEE Communications Society from 2011 to 2012. He was the Leading Guest Editor of the *IEEE Communications Magazine* Special Issues on Green Communications. He is an Associate Editor-in-Chief of the *China Communications* and the Series Editor of the *IEEE Communications Magazine* for its Green Communications and Computing Networks Series.