Mobile Data Offloading: How Much Can WiFi Deliver?

Kyunghan Lee, *Associate Member, IEEE*, Joohyun Lee, *Student Member, IEEE*, Yung Yi, *Member, IEEE*, Injong Rhee, *Member, IEEE*, and Song Chong, *Member, IEEE*

***Abstract*—This paper presents a quantitative study on the performance of 3G mobile data offloading through WiFi networks. We recruited 97 iPhone users from metropolitan areas and collected statistics on their WiFi connectivity during a two-and-a-half-week period in February 2010. Our trace-driven simulation using the acquired whole-day traces indicates that WiFi already offloads about 65% of the total mobile data traffic and saves 55% of battery power without using any delayed transmission. If data transfers can be delayed with some deadline until users enter a WiFi zone, substantial gains can be achieved only when the deadline is fairly larger than tens of minutes. With 100-s delays, the achievable gain is less than only 2%–3%, whereas with 1 h or longer deadlines, traffic and energy saving gains increase beyond 29% and 20%, respectively. These results are in contrast to the substantial gain (20%–33%) reported by the existing work even for 100-s delayed transmission using traces taken from transit buses or war-driving. In addition, a distribution model-based simulator and a theoretical framework that enable analytical studies of the average performance of offloading are proposed. These tools are useful for network providers to obtain a rough estimate on the average performance of offloading for a given WiFi deployment condition.**

***Index Terms*—Delayed transmission, experimental networks, mobile data offloading, mobility.**

I. Introduction

MOBILE data traffic is growing at an unprecedented rate. Many researchers from networking and financial sectors [2], [3], [18], [27] forecast that by 2014, an average broadband mobile user will consume 7 GB of traffic per month, which is 5.4 times more than today's average user consumes per month, and the total mobile data traffic throughout the world will reach about 3.6 EB per month, 39 times increase from 2009 at a compound annual rate of 108%. It is also predicted by Cisco that about 66% of this traffic is mobile video data [2]. The main drive behind this explosive growth in traffic demand is rapid increase in the number of smart phones and tablets that offer ubiquitous Internet access and proliferation of traffic-intensive applications for such smart devices (e.g., applications providing cloud-based services).

There are several solutions to this explosive traffic growth problem. The first is to scale the network capacity by building out more cell towers and base stations of smaller cell sizes (e.g., picocell, femtocell) or upgrading the network to the next-generation networks such as Long Term Evolution (LTE) and WiMAX. However, this is not a winning strategy, especially under a flat price structure where revenue is independent of data usage. It is interesting to note that most of these data consumptions come from a small percentage of mobile users: While smartphone users constitute about 3% of the total users in AT&T, they consume about 40% of the network traffic as of the end of 2009 [18]. Besides, expanding the network capacity may even exacerbate the problem by encouraging more data usages since the first deployment of the 4G networks is likely targeting the densely populated metropolitan areas like Manhattan, NY, or San Francisco, CA. The second is to adopt a usage-based price plan that limits heavy data usages. While price restructuring is rather inevitable, pure usage-based plans are likely to backfire by singling out a particular sector of user groups, e.g., smartphone users, which have the highest potential for future revenue growth.

WiFi offloading seems the most viable solution at the moment. Building more WiFi hotspots is significantly cheaper than network upgrades and build-out. Many users are also installing their own WiFi access points (APs) at homes and work. If a majority of traffic is redirected through WiFi networks, carriers can accommodate the traffic growth only at a far lower cost. Given that there is already a widespread deployment of WiFi networks, WiFi offloading addresses the "time-to-capacity" issue for the currently pressing need of additional network capacity.

There are two types of offloading: *on-the-spot* and *delayed* [20]. On-the-spot offloading is to use spontaneous connectivity to WiFi and transfer data on the spot. Most of the current smartphones support *on-the-spot* offloading by default. In delayed offloading, each data transfer is associated with a deadline, and the data transfer is resumed whenever getting in the coverage of WiFi until the transfer is complete. If the transfer does not finish within its deadline, cellular networks

Manuscript received January 06, 2011; revised October 21, 2011 and March 29, 2012; accepted May 29, 2012; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor A. Feldmann. Date of publication November 16, 2012; date of current version April 12, 2013. This work was supported in part by the National Science Foundation under Grants CNS-0910868 and CNS-1016216 and the Korea Communications Commission (KCC), Korea, under the R&D Program KCA-2012-11913-05004 supervised by the Korea Communications Agency (KCA). *(Corresponding author: J. Lee)*

K. Lee is with the School of Electrical and Computer Engineering, Ulsan National Institute of Science and Technology (UNIST), Ulsan 689-798, Korea (e-mail: khlee@unist.ac.kr).

J. Lee, Y. Yi, and S. Chong are with the Department of Electrical Engineering, KAIST, Daejeon 305-701, Korea (e-mail: jhlee@netsys.kaist.ac.kr; yiyung@kaist.edu; songchong@kaist.edu).

I. Rhee is with the Department of Computer Science, North Carolina State University, Raleigh, NC 27695 USA (e-mail: rhee@ncsu.edu).

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

Digital Object Identifier 10.1109/TNET.2012.2218122

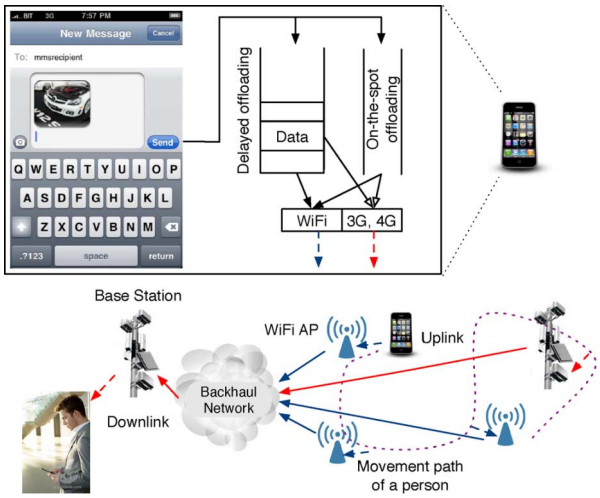


Fig. 1. Sketch of a system supporting delayed mobile data offloading. Our illustration focuses on uplink scenarios, but can be symmetrically extended to downlink cases.

finally complete the transfer. Fig. 1[1] illustrates a system that enables offloading.[2]

On-the-spot offloading is prevalent in current smartphones, but delayed offloading, whose notion is close to delay-tolerant networks, is relatively new. While users prefer to have data immediately, if network carriers provide more incentives in price for users to use transfer data with longer deadlines [35], users will be willing to tolerate delay for non-real-time traffic, e.g. down-and-play video/audio, software update, mobile backup, etc. For instance, Alice records videos and photos of her family outing at a park using her smartphone and wants to archive it in her cloud data storage. She can use delayed offloading since she does not need the data until she arrives home after a few hours. This scenario is in fact implemented in a project on Urban Tomography [1].

There is no doubt that both on-the-spot and delayed offloading reduce the load on 3G networks. However, an important yet underaddressed question is how much benefits offloading can bring to network providers and users. Network carriers are interested in knowing how much traffic load WiFi offloading takes away from cellular networks under a given or future WiFi network deployment. On-the-spot offloading is currently being offered through smartphones. Since carriers do not have control over WiFi networks that users connect to, they have no idea how much on-the-spot offloading helps them even now, let alone the future. How much does the new notion of delayed offloading help reduce their traffic given the projected amount of data growth in the future? The answers to these questions can provide clues on their price and cost restructuring strategies. Users are also interested in offloading because of economic reasons, e.g., a potential decrease of subscription fees or better service with the same fees. The average delays of offloaded data are also important to users. If they can predict in advance how long the actual data transfers will take on average based on their own mobility patterns, they can use that information in choosing the right price and deadlines for their transfer services. Users are also interested in actual energy saving that delayed offloading can achieve. All the above questions are fundamentally tied to the mobility patterns of users as users may come in and out of WiFi coverage. In this paper, we offer rough and rule-of-thumb answers to these questions.

[1]Note that each data associated with a deadline is served in a shortest remaining time first (SRTF) manner, not in a first-in–first-out (FIFO) manner.

[2]Although the figure shows an uplink scenario, the same can be symmetrically applied to downlink with a server supporting data queueing inside a carrier network.

A. Summary of Results

There have been several recent studies [6], [7], [23], [26] on the related topic. Some [7], [23], [26] have studied in the context of energy saving with the assumption that data can tolerate a delay of 1 min to a few hours, and the other [6], [23] in the context of on-the-spot or short delayed (up to 100 s) offloading. However, the benefits of the full-scale delayed offloading under daily WiFi usage traces have not been studied in detail. The data sets used in these studies are somewhat limited to be generalized. In [6], the authors use several traces of a war-driving around a city using their own vehicles and 20 city transit buses, which might incur more frequent WiFi contacts and shorter connection duration per contact. These data sets are not general enough to answer our questions as they do not account for the *temporal coverage* of normal users in their daily lives (i.e., their results are meaningful only when mobile data are generated in a city transit bus or in their war-driving scenarios) and their characteristics (e.g., how often and when they enter and leave WiFi zones, how long and when they stay in WiFi zones). The authors report about 10%–30% of the total traffic can be offloaded using on-the-spot offloading, and with up to 100-s delays, delayed offloading can achieve about 20%–33% additional gains over on-the-spot offloading. However, our results using whole-day traces show that on-the-spot offloading can offload about 65% of the total traffic load and delayed offloading with 100-s delay deadlines can get only 2%–3%. In [26], the authors study energy saving efficiency using a set of walk traces, each walk taking a few hours with an instrumented mobile device. For our study, these data are of limited use because each trace is too short to account for the daily-life patterns of users. More details on related work can be found in Section V.

We offer, to the best of our knowledge, the first quantitative answers to some of these questions by conducting an extensive measurement study in South Korea. For our measurement study, we first designed and implemented an iPhone application that tracks WiFi connectivity. We recruited 97 iPhone users from the Internet who downloaded our application to their phones and used it for about a two-and-a-half-week period in February 2010. About 55% of the users live in Seoul, and the others in the other major cities in Korea. None of the users, to our knowledge, are related to the authors. We briefed the users about the types of the measured data and their objectives. The phone is configured to connect to various WiFi networks as the users travel, including its carrier's WiFi network. The application runs in the background to record the locations of WiFi stations to which each user connects, the connection times and durations, and the data transfer rates between WiFi stations and smartphones, and then periodically uploads the recorded data to our server. These data are used to carry out trace-driven simulation of offloading with diverse data traffic and WiFi deployment scenarios.

From our data, we find that users are in a WiFi coverage zone for 70% of their time on average (63% during the active hours (9:00 ∼ 24:00). They stay in a coverage area for about 2 h on average, and after leaving the area, they return to an WiFi area within 40 min (this time interval is called *interconnection times*). The distributions of these statistics have a strong heavy-tail tendency. Data rates from the phone to our measurement server in the Internet are about 1.26 Mb/s on average during the active hours and 2.76 Mb/s during the nighttime. The full analysis is presented in Section II-B.

Using the data traces we obtained from the experiments, we run a trace-driven simulation to measure the efficiency of on-the-spot and delayed offloading. Our simulation uses the measured data rates from our traces, and each data transfer by a user in a WiFi zone is assumed to run at the actual transfer rate experienced by the user in our trace. This ignores the effect of changed load (e.g., contention) on the network bandwidth in the future. The same simulation strategy is used in [6]. The results below must be interpreted as upper bounds if the carriers can sustain the measured data rates through additional WiFi resource provisioning in the future.

The following are the key findings from our simulation.

1) On-the-spot offloading can offload about 65% of the total traffic load. This is achieved without using any delayed transfer. When delayed offloading is used with 100-s delay deadlines, the achievable gain over on-the-spot is very insignificant: 2%–3%. Our analysis indicates that in order for delayed offloading to get significant gains, the deadline must be much longer than 100 s because of long interconnection times. When data transfers are opted by users for delayed transfers with a deadline of 1 h and longer, the gain over on-the-spot becomes larger than about 29%.
2) On-the-spot offloading alone (without any delayed transfer) can achieve about 55% energy saving for mobile devices because WiFi offloading can reduce the transmission time of mobile devices substantially. However, for delayed transfers with very short deadlines like 100 s, the achievable energy saving gain over on-the-spot offloading is highly limited to about 3%. However, with 1-h delay, the achievable energy saving gain increases to around 20%.
3) For a prediction-based offloading strategy like Breadcrumbs [6], [23] to be useful, it has to predict over several tens of minutes since the interconnection time has a median of 10 min (90th percentile of 162 min). Because of the heavy-tail tendency of the interconnection times, this prediction will be even harder.
4) The average completion time of data transfers is much shorter than their delay deadlines. While on-the-spot offloading obviously achieves faster transfer than using 3G networks only, it is surprising that video file transfers of size larger than 30 MB with 1-h deadline are consistently faster than no offloading. Furthermore, the 3G network usage reduction gain of these transfers is more than 50% over on-the-spot offloading and more than 80% over no offloading, which implies 50% or more cost reduction for the carriers to deliver such transfers and translates directly into price reductions for users.

We develop a theoretical framework using a *queueing model with impatience of customers and service interruptions*. The model can be used to predict the average performance of offloading for a given WiFi deployment condition that can be expressed by statistics on the durations of users inside and outside a WiFi coverage area (i.e., temporal coverage). Simulating this queueing model can predict the performance of offloading with about 10% margin error.

Fig. 2. iPhone App, DTap, for measuring WiFi availability.

More detailed analysis of our simulation can be found in Sections III. We describe our theoretical framework in Section IV and summarize related works in Section V. We present some discussions on the limitations of our work and future work in Section VI.

II. Measurement Study

A. Experimental Setup

The performance of offloading highly depends on the patterns of WiFi coverage and user mobility. Accurate modeling of offloading performance calls for a measurement study. We first develop an Apple iPhone application, called *DTap* (Delay Tolerant APplication) that records the statistics of WiFi connectivity in the background and periodically sends the recorded statistics to a server (see Fig. 2 for a screenshot). Running in the background, DTap scans for WiFi connectivity at every 3-min interval. As scanning for WiFi, iPhone connects to the AP, if any, with the strongest signal strength among those to which it has a past history of connections. Note that the captured WiFi APs include the private APs at home and work and commercial APs installed by the carrier of the mobile phone and the third-party companies (e.g., Boingo). As our participants are actively using their iPhones, these APs are mostly included in their past histories. After connecting to a WiFi network, it measures data throughput and round-trip times by pinging the server with a 100-B packet 10 times and computing the average. This measures the end-to-end data rate between the client phone and a server we have. This is obviously not the most accurate measurement method, but it is reasonable under the constraint that DTap should minimally consume bandwidth and battery of a smartphone.[3] Note that, according to an extensive measurement

[3]By running DTap based on the ping test, participants' usage time had already been reduced significantly from about 1 day to 4–6 h. Also, to enable more users' participation in our experiment, we had to strictly avoid potential 3G data consumption of participants since data cost during the experiment period was very high in South Korea.

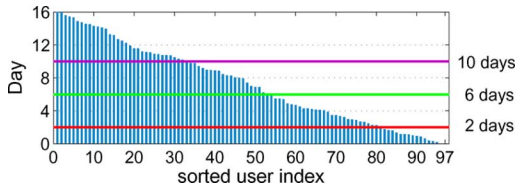


Fig. 3. Number of valid experimental days for each user. Users are sorted in the descending order of the total log duration.

study [21], the data rate estimated through ping has strong positive correlation to the measurement done by bulk data transfer using FTP. Also, it is shown in [31] that the bandwidth measurement through small packets in the size of a ping packet can be as accurate as measurement methods that generate much higher probe traffic. DTap regularly records in a log file the GPS location where the connection occurs, and the duration, data rate, and time of the connection in every 3 min. All the information recorded by DTap was released to the participants in advance. Mainly due to the privacy and energy concern of participants, DTap excluded recording user traffic generation behaviors.

DTap does not perform offloading. This is because performing offloading directly in participants' phones for arbitrarily generated data drains too much battery, which faces strong resistance from volunteers. Even with just the WiFi scanning and pinging tests, the phone drains the battery power very quickly. Instead, we take an approach of collecting detailed traces of WiFi connectivity and later using the traces to simulate offloading under diverse traffic patterns.

The log files are uploaded to our server using ftp connections daily between 4:00 AM to 4:30 AM. The daily log file size is typically less than 1 MB. DTap runs with customized parameters that are contained in an XML configuration file automatically updated to client phones daily. Each row of the log file contains the following tuple: (*device id, time stamp, event name, field 1*, . . . *field* n). The device id is the unique id of a phone. The time stamp is the time when the corresponding tuple is recorded. Multiple event names are used in our experiment, depending on which, the number, and the values of associated fields are decided. They are summarized in Table I. The AP list represents all the APs that the phone can currently detect. GPS location is associated with the location accuracy information provided by the phone.

We recruited 97 volunteers who own iPhone 3G/3GS from an iPhone user community in Korea and asked them to install DTap in their phones for a period of 18 days in January and February 2010. The volunteers come from diverse occupational backgrounds and various major cities in Korea (60% from Seoul). For data integrity, we have excluded a very small number of daily traces that show no movement (as users might have forgotten to carry their phones). Fig. 3 shows the number of experimental days for each participant. The total number of valid daily traces we collect is 705.

Note that our data can be biased in the sense that we collected traces of the participants from an iPhone user community in Korea. Korea is well known for its high-speed access networks [25]. WiFi networks may be also densely deployed than other countries. Also, the users collected from the iPhone community are likely to be tech-savvy users, who may have more

TABLE I
EVENT NAMES AND ASSOCIATED FIELDS IN THE LOG FILE

Event Name	Associated Fields
WiFi connectivity	0 or 1
AP Lists	SSID1, . . . , SSIDn
GPS	latitude, longitude, accuracy
Data rate	rate

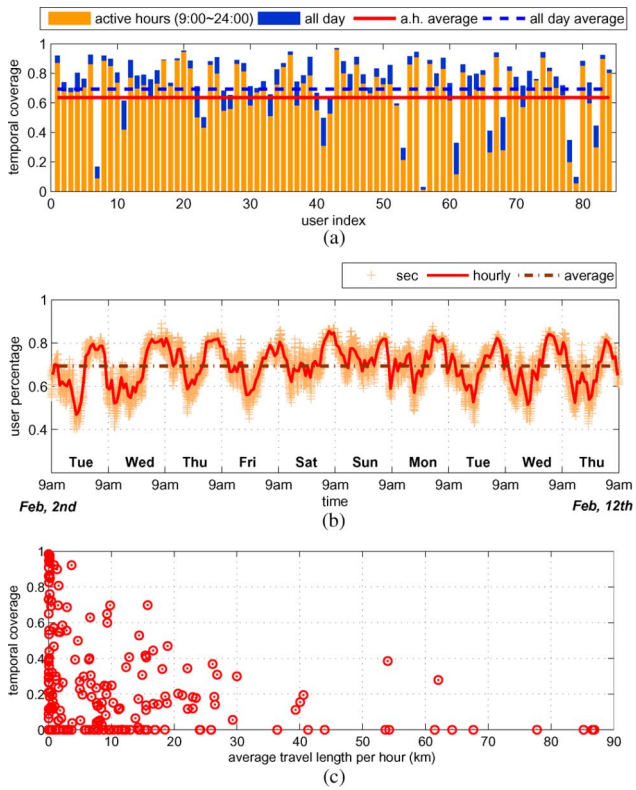


Fig. 4. Temporal coverages per user, time, and hourly mobility. (a) Temporal coverage of users. Users are ordered in the same order as in Fig. 9(a). (b) Percentage of users with WiFi access from 9:00 AM, Feb. 2, to 9:00 AM, Feb. 12. (c) Temporal coverage for hourly mobility.

opportunities to see WiFi networks in their vicinities than average users. We expect average countries or users will eventually get similar WiFi network environments in near future.

B. Key Observations

We measure the following statistics relevant to offloading: the total time duration of WiFi connectivity, the data rate during connections, the distributions of connection times and interconnection times, and the correlations of the total travel lengths with the data rate and time of WiFi connectivity time.

Temporal Coverage: The performance of offloading highly depends on the time portion that a user stays in a WiFi coverage area, which is defined as *temporal coverage*. Fig. 4(a) shows the daily average temporal coverage recorded by each participant. It also plots the coverage recorded during the active hours (9:00 ∼ 24:00).[4] The averages across all the users are 70% for all day and 63% for the active hours only. Difference between all day and active hours averages arises because most participants are likely to have WiFi connectivity at home. Fig. 4(b)

[4]Our traces reveal that participants are actively changing their locations during 9:00–24:00, probably because of their jobs and frequent bar-hopping.

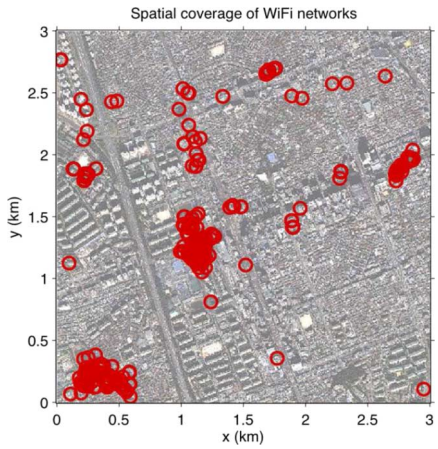


Fig. 5. Locations of WiFi APs detected by the participants in a 3×3-km^2 area with the most dense WiFi deployment inside Seoul (more visible in color).

shows the percentage of users that have WiFi connectivity at any given time averaged over 1-s and 1-h periods, respectively. It indicates that at any time, about 70% of users stay in a WiFi coverage area.

There is a substantial difference between the data from [6] that reports 11% temporal coverage. This difference comes from the fact that their measurements are done only when a user is on a vehicle. Typically, users spend most of their time in the office and home. This type of information is missing as average users are not likely to spend most of their time only inside a car or bus. To verify our conjecture, we also record the traveling distances of each user for each hour. This can be calculated as the log contains GPS data. We map the temporal coverage during each hour to the travel distance that the user makes during that time period. Fig. 4(c) shows the results. The results indicate that users with high mobility (i.e., including those moving in a car) have very low temporal coverage.

We measure *spatial coverage*, which is defined to be the fraction of an area that is under any WiFi coverage. Our traces give only a rough estimation of spatial coverage since they do not capture all possible WiFi APs located in the city because the walkabouts of participants do not cover the whole area. However, it certainly gives a lower bound. Fig. 5 shows the locations of WiFi APs that the users visit in a 3×3-km^2 area of the city where the users visit most. We measure the spatial coverage by drawing 50-m-radius circles, a typical WiFi range, around each WiFi-detected AP and totaling the areas of the drawn circles. Our analysis shows that the spatial coverage is about 8.3% (20.6% for 100-m-radius circles).

Our data show that the temporal coverage is about $3.5 \sim 8$ times larger than the spatial coverage for a given region, indicating that most users stay inside a WiFi network for a long time once they connect to a WiFi network. Fig. 6 shows the complementary cumulative distribution function (CCDF) of the stay time (called *connection times*). The average connection times is about 2 h for all day and 52 min for active hours only. Fig. 7 shows the CCDF of *interconnection times*, the time duration after a user leaves a coverage area until it returns to a coverage area. The average is about 40 min for all day and 25 min for active hours. Similarly, we measured the medians (and 90th percentiles) of connection and interconnection times. For whole-day traces, they are shown to be 7.4 (694) and 10.5 (162) min, respectively. They become 7.4 (280) and 10.4 (109) min when considering active hours only. An interesting observation from our trace is that both CCDFs show a heavy-tailed tendency and, in particular, fit best with *truncated Pareto distributions*[5] using maximum likelihood estimation (MLE). To find the best fit, we performed two well-known test methods: Cramer–Smirnov–Von–Mises (CSVM) statistical hypothesis test [12], [33] and Akaike test [28], where the tested distributions are exponential, log-normal, Weibull, and truncated Pareto distributions. The test results for aggregate interconnection and connection time distributions are summarized in Table II. In both tests, a smaller criterion measure (value) for a distribution indicates a better fit.

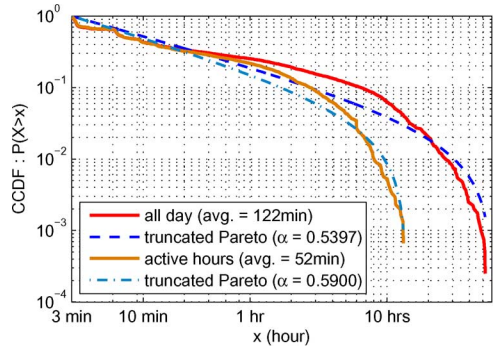


Fig. 6. CCDF of connection times. The median and 90th percentile are 7.4 and 694 min. The distribution fits best with truncated Pareto distribution with $\alpha = 0.54$ for all day and $\alpha = 0.59$ for active hours only.

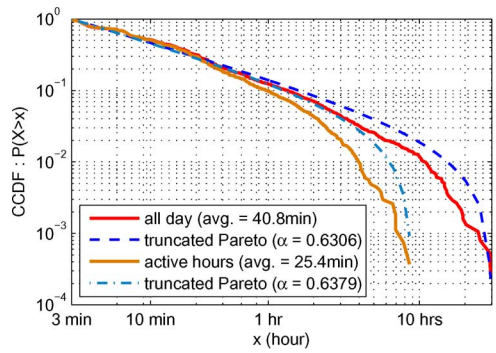


Fig. 7. CCDF of interconnection times. The median and 90th percentile are 10.5 and 162 min. The distribution fits best with truncated Pareto distribution with $\alpha = 0.63$ for all day and $\alpha = 0.64$ for active hours only.

The parameter α in truncated Pareto distribution controls the shape of the PDF function. Pareto distribution that has no truncation (i.e., $\nu = \infty$) is known to become heavy-tailed when $0 < \alpha < 2$. A smaller α gets a heavier tail. The measured statistics fit very well with $\alpha = 0.54$ and 0.63 for connection and interconnection times, as shown in Figs. 6 and 7. The minimum value of truncated Pareto distributions is set to be 3 min, which is our measurement interval. The maximum value of truncated Pareto distributions is set to be the maximum observed from all samples. The maximum of interconnection and connection times for all day are 29.5 and 58.3 h, respectively. The maximum of interconnection and connection times at active hours (9:00 $\sim$ 24:00)

[5]The probability density function (PDF) of the truncated Pareto distribution with the parameters α is $(\alpha\gamma^\alpha/(1-(\gamma/\nu)^\alpha))x^{-(\alpha+1)}$, where x is truncated as $0 < \gamma \leq x \leq \nu$.

TABLE II
CSVM CRITERION AND AKAIKE INFORMATION CRITERION (AIC) FOR EXPONENTIAL, LOG-NORMAL, WEIBULL, AND TRUNCATED PARETO DISTRIBUTIONS OVER AGGREGATE INTERCONNECTION TIME AND CONNECTION TIME. TRUNCATED PARETO DISTRIBUTION SHOWS THE BEST FITS FOR ALL CASES

	Exponential	Log-normal	Weibull	Truncated Pareto
All day				
ICTs; CSVM	0.0415	0.0032	0.0088	**0.0012**
ICTs; AIC	6.53e+4	6.20e+4	6.36e+4	**6.02e+4**
CTs; CSVM	0.0943	0.0105	0.0134	**0.0097**
CTs; AIC	7.86e+4	7.12e+4	7.26e+4	**6.72e+4**
Active hours				
ICTs; CSVM	0.0138	0.0022	0.0061	**0.0014**
ICTs; AIC	5.44e+4	5.15e+4	5.25e+4	**4.88e+4**
CTs; CSVM	0.0549	0.0084	0.0112	**0.0072**
CTs; AIC	4.41e+4	4.29e+4	4.39e+4	**4.20e+4**

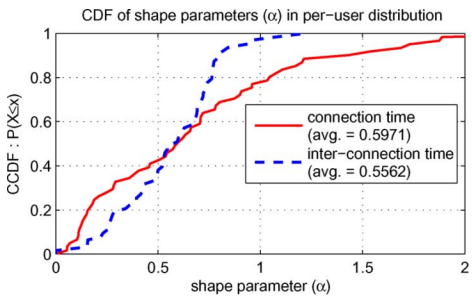


Fig. 8. CDFs of truncated Pareto fitting parameter α for per-user interconnection time and connection time. Most of the α values are less than 1.2, showing that per-user statistics also have heavy-tail tendency.

TABLE III
PROPORTION OF USERS SHOWING THEIR BEST FITS IN THEIR PER-USER INTERCONNECTION AND CONNECTION TIME DISTRIBUTIONS FOR FOUR DISTRIBUTIONS

	Exponential	Log-normal	Weibull	Truncated Pareto
ICTs; CSVM test	1.5%	28.8%	22.7%	**47.0%**
ICTs; Akaike test	1.5%	3.0%	3.0%	**92.4%**
CTs; CSVM test	0%	33.3%	27.3%	**39.4%**
CTs; Akaike test	3.0%	1.5%	0%	**95.5%**

are 10.3 and 13.2 h, respectively. It is interesting to see that the interconnection time distribution shows a similar pattern to the intercontact time distribution observed from human mobility, which is known to be heavy-tailed [10], [19], [28].

We further applied CSVM and Akaike tests to *per-user* interconnection and connection time statistics to figure out the best fitting distribution. For 66 users showing more than 10 measurement samples, we counted the number of best fits for the aforementioned four distributions, as shown in Table III. In both tests, truncated Pareto outperformed other distributions. We summarized the best fitting α parameters of truncated Pareto distribution for connection and interconnection time as CDFs in Fig. 8. Most α values are found to be less than 1.2, indicating that per-user statistics also have heavy-tail tendency.

The heavy-tail tendency of interconnection times with a large average (25–40 min) implies that the prediction-based offloading strategies like Breadcrumbs [23] may not be effective enough. These strategies use past history of user mobility

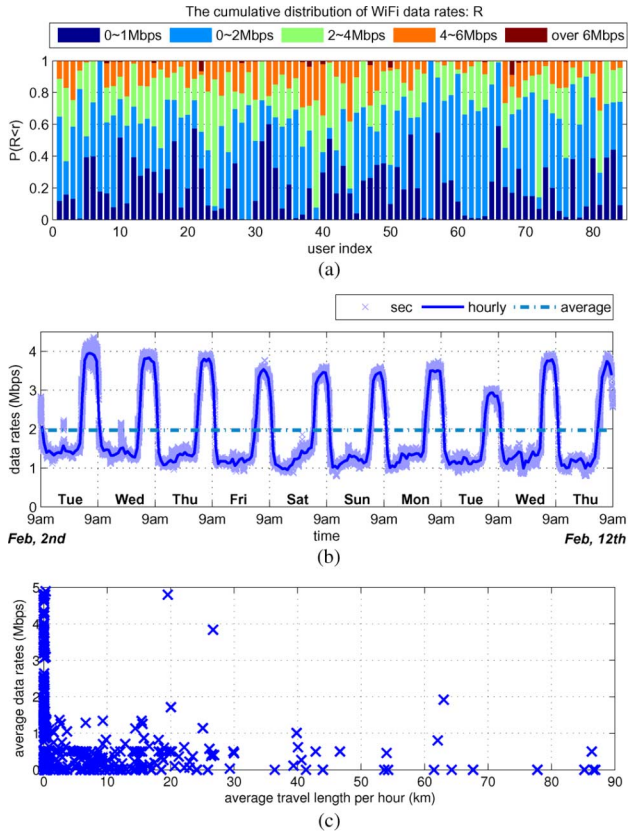


Fig. 9. End-to-end data rates per user, day and hourly mobility. (a) Cumulative distributions of the end-to-end data rates. Users are ordered in the same order as in Fig. 4(a). (b) User-averaged end-to-end data rates from 9:00 AM, Feb. 2, to 9:00 AM, Feb. 12. Data rates are high at the nighttime. (c) Per-hour mobility versus data rate.

and predict whether users will be entering a WiFi zone with fast transmission rates within a given deadline. Typically, these algorithms use short duration (e.g., 100 s) for look-ahead times. However, the large median (or 90th percentile) value of interconnection times requires these algorithms to look ahead much farther. Furthermore, because of the heavy-tail tendency of interconnection times, the efficiency of the prediction may not be high. We will see the detailed performance of offloading in Section III for various delay deadlines in the scale of median and average values along with arbitrarily small and large delay deadlines.

End-to-End Rates: Fig. 9(a) shows the cumulative distributions of end-to-end data rates reported by users. It shows a variety of experienced rates with their average being around 1.97 Mb/s. This average is highly skewed by the data rate during the nighttime. Fig. 9(b) shows the data rates averaged across users for each time period. It can be seen that the high data rates during the night time are around 2.76 Mb/s, and on average, users are experiencing around 1.26 Mb/s during the active hours. During the nighttime, users are likely connecting to their home APs. This data shows that offloading during the nighttime is going to be very effective if users can tolerate large delays. We also map the measured data rate per hour for each user to their hourly traveling distance [Fig. 9(c)]. It is shown that user mobility has weaker correlation with data rate than temporal coverage.

III. Offloading Efficiency

In this section, we report the simulation results using the traces we discussed in Section II. Since we have detailed records of user connectivity and data rates during the connectivity for every 3-min interval, they can be used to simulate the offloading of input traffic with diverse patterns.

A. Simulation Method

As it is impossible to accurately anticipate the mobile data usage patterns of users under the situation where the offloading is adopted, for each user, we exhaustively generate input data traffic with diverse arrival and size patterns. A data request for upload (or download) arrives during typical active hours (9:00 $\sim$ 24:00) to the phone of a user (or to an offloading server in a carrier's network) with a random interarrival time and a random size selected from input distributions (typically exponential or Weibull[6]) of a mean a for interarrival times and a mean b for file sizes. We say that b/a is *traffic intensity*. Generating traffic during active hours can be simplistic, given recent measurement studies [13], [14], [24] that characterize the smartphone usage patterns including temporal characteristics of aggregate traffic, traffic volume distribution over users, and popularity of applications. However, considering that the temporal correlation revealed so far has only rough information (e.g., aggregate traffic volume during the active hours is larger than nighttime), more detailed spatial and temporal correlations of individual mobile data traffic have not been clearly investigated. Thus, we provide extensive simulation results for various traffic intensities, which may give some insights of upper- or lower-bounded offloading efficiency in practice.

We assume that each data request is associated with a deadline typically assigned by its user or application program depending on the type of data. Upon arrival, each request is scheduled according to its deadline such that a request with smallest remaining time is served first (i.e., SRTF). The transmission time of a data transfer is determined by the measured data rate experienced by the user at the time of the transfer (this is taken from the user log trace). If the transfer cannot be finished before its deadline, the request is assumed to be uploaded (or downloaded) through 3G networks and is removed from the queue. We develop a MATLAB simulator that follows a simulation model depicted in Fig. 10.

We define *offloading efficiency* to be the total bytes transferred through WiFi divided by the total bytes generated.

B. Traffic Model

To understand the impact of offloading in relieving the future traffic demands, we use the projection data released from CISCO [2] on the amount of mobile traffic demands by year 2014. It is predicted that an average user consumes about 7 GB per month, and the contribution of various data types to this traffic is summarized in Table IV. We assign three different types of offloading deadlines to each data type from short to long deadlines. The short and medium deadlines are 10 min and 1 h, which represent the scales of median and average values of interconnection time. We also test long and very long deadlines corresponding to 6 and 12 h. Note that as we shall see later in Section III-D, most transfers finish well before their deadlines. We assume that the interarrival time distribution is exponential and the distributions of the arrival traffic volumes of video, text, and data are Weibull and that of the audio (VoIP) data is exponential. The means of all the distributions are deduced from the estimated monthly traffic of each type. Audio and P2P (e.g., data sharing in the proximity) data are assumed to be not delay-tolerant, so zero delay deadlines are assigned to them.

[6]The PDF of Weibull distribution with the parameters α and k is $(k/\alpha)(x/\alpha)^{k-1}\exp[-(x/\alpha)^k]$. When $k<1$, the distribution is heavy-tailed, and as k gets smaller, it becomes more heavy-tailed.

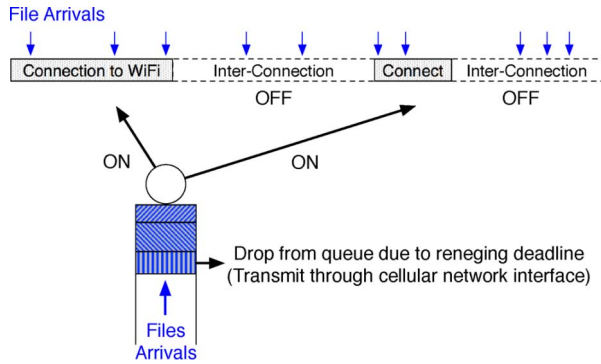


Fig. 10. Simulation model of a user. The data in the user queue are serviced only when a smartphone is connected to a WiFi network in an SRTF manner. When the delay deadline of the file in the queue expires, the file is removed from the queue and transmitted through 3G networks (the same queueing model can be used for downlink).

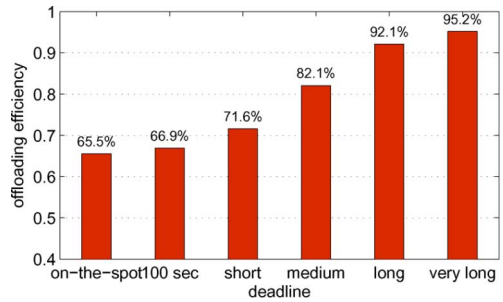


Fig. 11. Offloading efficiency of delayed transfers with various deadlines when all the transfers are opted for delayed transfers.

TABLE IV
Input Data to the Experiment for Fig. 11. We use the Projection From [2] on the Amount of Mobile Data Traffic and Their Constituent Types and Proportions in Year 2014. We Assign Various Deadlines to Different Types of Data From Short to Very Long Deadlines. The Mean InterArrival Times are Calculated From the Predicted Monthly Data Volume (DL Denotes Deadline)

	Video	Data	P2P	Audio (VoIP)	Total
Ratio [2]	64.0 %	18.3 %	10.6 %	7.1 %	100 %
Data/month	4.48 GB	1.28 GB	740 MB	500 MB	7 GB
Avg. IAT	1 hour	2 hours	2 hours	1 hour	-
Traffic vol.	10 MB	5.7 MB	3.3 MB	1.1 MB	-
Traffic dist.	Weibull (k=0.5)	←	←	Exponential	-
On-the-spot	0 sec.	0 sec.	0 sec.	0 sec.	-
DL:short	10 min.	10 min.	0 sec.	0 sec.	-
DL:medium	1 hour	1 hour	0 sec.	0 sec.	-
DL:long	6 hours	6 hours	0 sec.	0 sec.	-
DL:very long	12 hours	12 hours	0 sec.	0 sec.	-

In this work, we assumed that mobile data traffic are randomly generated at users' devices independent from their positions. Due to privacy concerns, we intentionally avoided collecting data generation patterns from users, but incorporating these patterns from real data is clearly of our future work.

C. 3G Network Traffic Reduction

In this section, we measure the amount of traffic offloading to WiFi from 3G networks. Fig. 11 shows the offloading efficiency of on-the-spot and delayed offloading. We added the results using 100-s delay deadlines to show the impact of very short delay mainly discussed in other research works. In this experiment, we assume that all transfers of video and data use delayed transfers. It is surprising that on-the-spot offloading (without any delays) can achieve extremely high offloading efficiency already. Note that on-the-spot offloading is what is currently being performed by smartphones today. If most of mobile data volume comes from smartphones, WiFi can offload more than 65% of traffic even today.

As we increase delay deadlines, offloading efficiency increases substantially. For long deadlines, the efficiency increases to 92.1%, indicating most mobile data can be offloaded to WiFi. However, with short and medium deadlines corresponding to the scale of median and average interconnection time, we get only 9.3% and 25.5% gains. This is because: 1) offloading efficiency is computed by offloaded data volume/total traffic volume; and 2) the amount of traffic volume generated during a short interconnection is mostly small. With 100-s-or-less deadlines, the additional gains of delayed transfers over on-the-spot become even smaller. Only 2%–3% gain is observable. That is substantially smaller than 20%–33% gain reported by [6]. This difference also comes from that their traces contain much shorter interconnection times as buses and cars travel much faster than average users on the street or offices. To have substantial gain using short delays, users must need to experience very short interconnection times (as if they are in a car).

The offloading efficiency of 92.1% for long deadlines is certainly unrealistic. It is not true that all transfers of video and data in Table IV are opted for delayed transfers with such a long delay (6 h). It is possible that despite pricing incentives, users may opt for on-the-spot offloading only. To see the effect of this, we measure the performance as the ratio of delayed transfers over the total data traffic (called *delayed transfer ratio*) is varied in Fig. 12. In this plot, we are interested in the gain achieved by delayed transfers over on-the-spot. Again, the gain achieved by 100-s deadlines is very minimal from 2% to 3% for 30%–50% delayed transfer ratios. The gain for 1-h deadline is about 13%–21% with 30% –50% delayed transfer. This result indicates that since on-the-spot offloading is already very good, for delayed transfers to achieve substantial gain, their deadlines must be fairly long (e.g., longer than several tens of minutes).

D. Completion Time

Deadlines of 30 min or 1 h seem very long for some applications. However, our results indicate that most transfers finish well before these deadlines. In Fig. 13, we measure the average completion times of transfers with various traffic types. For this experiment, we set the data rate of 3G networks to be

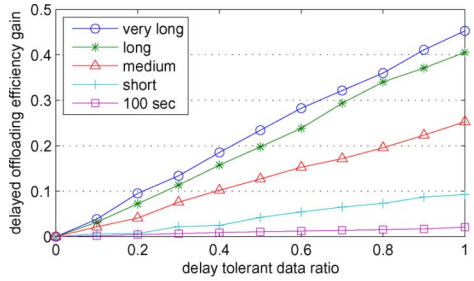


Fig. 12. Offloading efficiency gains over on-the-spot offloading achieved by delayed transfers as the delayed transfer ratio varies.

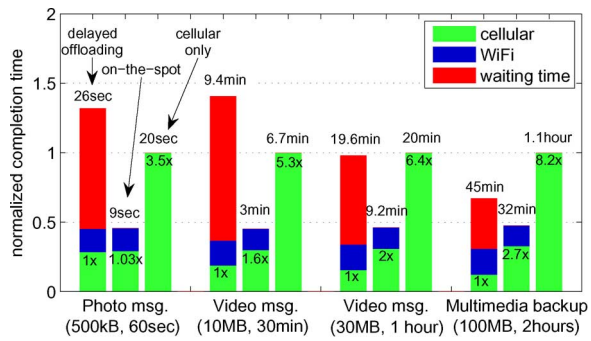


Fig. 13. Comparison of average completion times of offloading methods for various types of applications normalized to the time taken using only cellular networks. Parameters in the bracket show the size of files and applied deadlines (more visible in color).

200 kb/s, which we typically get through our iPhones for uplink [29]. For WiFi, we use the end-to-end data rate of WiFi measured by DTap, whose average is 1.97 Mb/s. We measure the completion times for: 1) delayed offloading; 2) on-the-spot offloading; and 3) no offloading (3G network only). The result in each traffic type is normalized by the completion time of no offloading. Photo messages with 60-s deadlines finish in 26 s on average, 6 s more than no offloading. The breakeven point where the completion time of delayed offloading becomes the same as that of no offloading occurs when video messages with 30 MB of 1-h deadline are transmitted. When that happens, the amount of 3G network usage of delayed offloading is half of that of on-the-spot. At this point, users using delayed offloading may experience the same delay as no offloading while the cost of delivery by the carriers is only half of that of on-the-spot and about 20% of no offloading. This happens because delayed offloading delays its transfer until it has a WiFi connectivity. Since WiFi offers higher data rate, more use of WiFi leads to shorter completion time. Although delayed offloading has a longer completion time than on-the-spot offloading, it uses 3G network far less, which is translated into cost reduction for carriers and price reduction for users. With larger file sizes or longer deadlines, delayed offloading achieves faster completion time and more cost reduction than no offloading. If 3G data rate changes, the 3G time portion in the graph (green bars) will change accordingly. This tells that when the tangible cellular data rate becomes slower, the gap in the completion time between delayed offloading and on-the-spot offloading will reduce. If the data rate increases by network upgrade, the gap will increase in the same manner. Overall, the cellular data rate affects the attractiveness of delayed offloading.

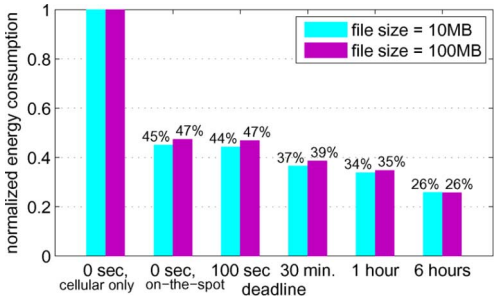


Fig. 14. Normalized energy consumption of delayed transfers of 10- and 100-MB files with 1-h deadline. File sizes and intervals are assumed to be exponentially distributed.

E. Energy Saving

There exists a fundamental tradeoff between energy consumption and delay in smartphone applications as smartphones have multiple radio interfaces with different transmission rates and availability [23], [26]. While 3G networks are more widely available than WiFi, their data rates are much less than WiFi. Therefore, by delaying transmission until WiFi is available, there are opportunities to reduce the transmission time as we have seen it in Section III-D. The reduced transmission times are directly translated into battery power saving for smartphones because energy consumption of WiFi per second is almost the same as 3G networks[30]. The transmission time of a transfer is different from its completion time as transmission times account for only the time that radio interfaces are used to complete the transfer. Thus, the transmission time is the time after subtracting the waiting time from the completion time in Fig. 13. We assume that power consumption during the waiting time is negligible using smart WiFi perception technology [5], [11], [32].

Fig. 14 examines power consumed for delayed transfers of 10- and 100-MB files with various delay deadlines, which is directly translated from their transmission times. The values are normalized to the energy consumption of no offloading. On-the-spot offloading already achieves 55% energy saving over no offloading because of reduced transmission time through use of WiFi. However, in order for delayed transfers to achieve substantial energy saving gain over on-the-spot offloading, the deadlines must be substantially long. With 100-s deadlines, the saving gain over on-the-spot is extremely limited. One-hour deadlines achieve about 20% additional gain.

F. Impact of Traffic Types

In this section, we further evaluate the detailed impact of varying input traffic characteristics to offloading efficiency. We especially focus on the interarrival time distributions of input files and file size distributions. To test diverse traffic types, we first vary traffic intensity. For instance, traffic intensities of text and video messages would be different. For the same traffic intensity, we also vary traffic burstness (i.e., 1/interarrival time or simply the number of files generated per unit time) and the file size distributions. We test deterministic, exponential, and heavy-tailed file-size distributions. In our simulation, we conducted simulations for the traffic intensities 0.1, 50, 500, and 5000 kB/min. For each traffic intensity, we test two different cases of traffic burstiness and file size.

Offloading efficiency for less bursty traffic, shown in the upper three plots (exponential interarrival time with 1-min. average) of Fig. 15, uniformly ranges from 0.7 to 1 depending on the stringency of delay deadline, but irrespective of average file sizes and file-size distributions. Specifically, for the traffic generated with the average rate of 5 MB per minute, even just 2 h of delay tolerance enables us to offload about 80% of data traffic from the current cellular network. This clearly shows a benefit of a combination of delay tolerance and user mobility, which increase the total system capacity significantly. It is intuitive that more bursty traffic induces lower offloading efficiency. The middle three plots (exponential interarrival time with 1-h average) of Fig. 15 show such a case that the files are generated every hour (thus, for the same traffic intensity, a file with larger size is generated), where a slight decrease of offloading efficiency is observed. However, such a decrease is visible only for short deadlines, and for long deadlines, the performance difference is not considerable. Note that heavy-tailed interarrival distributions are reported to appropriately model the time interval between consecutive e-mails [8]. The bottom plots (Weibull interarrival time with 1-h average) of Fig. 15 show the performance for the case.

It is known that applications often generate traffic whose file-size distributions are heavy-tailed in many cases. See [4] for the video file-size distributions in YouTube. Intuitively, more heavy-tailed traffic leads to lower offloading efficiency since file size far larger than the mean can be generated with non-negligible probability. Fig. 16 depicts the offloading efficiency for a varying heavy-tail degree in the file-size distribution controlled by the k value of Weibull distribution with the mean set to 100 MB. The interarrival times have an exponential distribution with 1-h average. Recall that smaller $k < 1$ generate more heavy-tailed traffic, and when $k = 1$, it boils down to the exponential distribution. We observe that even for very heavy-tailed traffic, the offloading efficiency is at least 20%, and over 40% of files with 2-h deadlines can be offloaded through WiFi except the extreme case, $k = 0.1$.

To get more realistic offloading efficiency, we set the input parameters of various application data considering the property of the data. We set the interarrival time of 0.1-kB text messages to 1 min constant, 500-kB photo messages to an exponential distribution with a mean 60 min, 10-MB video messages to a Weibull distribution with a mean 120 min and $k = 0.5$, and 100 MB of multimedia backup to a Weibull distribution with a mean 120 min and $k = 0.5$. Fig. 17 shows the result. Text and photo messages can be offloaded instantly at the rate of 70%. Video messages and multimedia backup can be offloaded around 70% with deadlines of 30 min and 2 h, respectively.

G. Impact of WiFi Deployment

We investigate the impact of WiFi density and deployment strategies on offloading efficiency. To test them, we use the current deployment observed in our traces as a baseline and thin out density by progressively eliminating WiFi APs according to two different strategies: *activity-based* and *random*. In the activity-based strategy, we measure the *connection time of an AP*, which is the sum of time duration that each user spends in the coverage of the AP. The activity-based strategy eliminates WiFi APs in the increasing order of their connection times until a target density is reached.

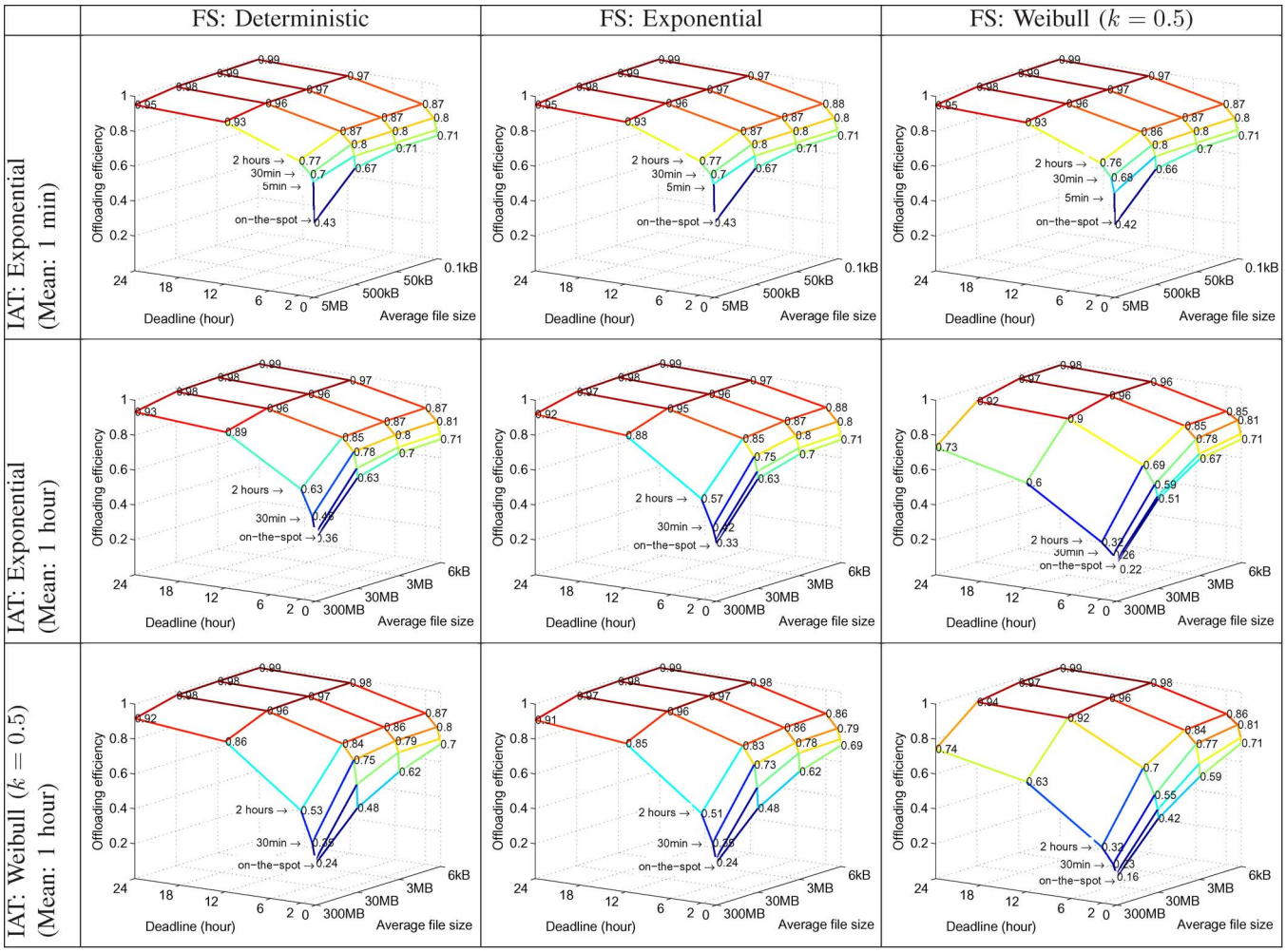


Fig. 15. Offloading efficiency for different traffic intensity, file-size distributions, and delay deadlines. Intersrrival time (IAT) follows exponential or Weibull whose mean in the bracket determines traffic burstiness. File size (FS) follows deterministic, exponential, or Weibull distribution whose mean is specified in the file-size axis in the figures. (More visible in color.)

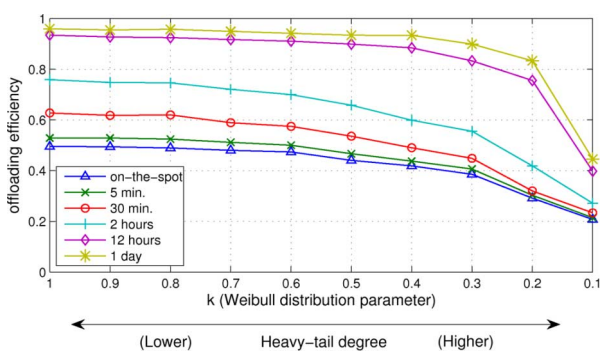


Fig. 16. Offloading efficiency varying k parameter of Weibull distribution. Mean file size is 100 MB, and the traffic generation interval is 1 h. When $k = 1$, Weibull distribution is exponential. As k decreases, it is more likely that a huge file arrives at the system.

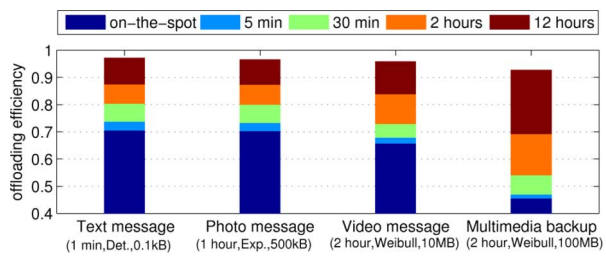


Fig. 17. Offloading efficiency of candidate applications (text, photo, video-clip messaging, and multimedia backup). Interarrival time, type of file-size distribution (Det: deterministic, Exp: exponential), and mean file size are denoted in the bracket.

We set the density of the current deployment measured from the trace to 1. The random strategy randomly eliminates APs with equal probability. Fig. 18 shows the offloading efficiency for two considerably heterogeneous traffic types, text messaging and multimedia data backup, whose traffic parameters are the same as those in Fig. 13. The activity-based strategy naturally outperforms random, but it is interesting to see that even after reducing its density by half, the activity-based strategy reduces offloading efficiency by only a small percentage, while the random strategy has about a 50% performance drop. It implies that careful deployment plans can yield substantial improvement in the capacity even with a small increase in density. We leave the investigation of the optimal strategy of WiFi deployment for delayed offloading as future work.

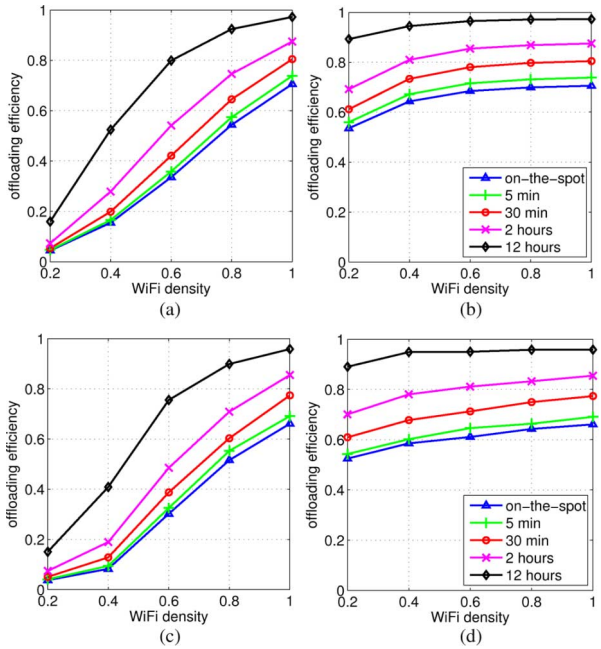


Fig. 18. Offloading efficiency for various amount of WiFi deployment and different deployment strategies. Irrespective of traffic type, activity-based deployment that might primarily lead to install WiFi APs to users' houses shows clearly higher offloading efficiency than random. (a) Random (text msg.). (b) Activity (text msg.). (c) Random (video msg.). (d) Activity (video msg.).

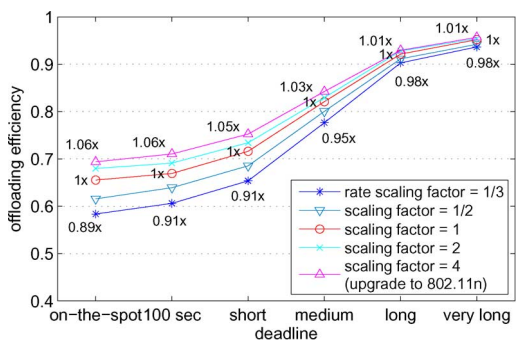


Fig. 19. Comparison of offloading efficiencies after scaling up or down data rates to mimic network conditions with physical-layer innovation (e.g., IEEE 802.11n) or with heavier congestion due to more smartphone users. The numbers (−x) represent the increase or decrease of the offloading efficiency compared to the offloading efficiency when the scaling factor is one. For example, the number "1.06×" at the on-the-spot deadline indicates that the offloading efficiency is increased by 6% when the scaling factor is four.

H. Impact of Throughput Scaling

It is relatively easy to forecast that the total WiFi network capacity increases in the future. However, predicting changes in WiFi network environment in the future for an individual user is of extreme difficulty since it is both possible for a user to experience higher data rates mainly due to more investment from carriers or physical-layer innovations (e.g., 802.11n, 802.11ac/ad) and lower data rates mainly due to excessive smartphone users compared to the present. To see the impact of the changes in both directions, we emulate the network environments by scaling up and down the current data rates as shown in Fig. 19. We performed simulations with scaled data rates from 33% to 400% under the same data traffic shown in Table IV. Interestingly, the performance gap in the offloading efficiencies under widely different data rates becomes reduced as longer deadlines are allowed. This implies that longer deadlines are more beneficial to the networks with limited capacity because the deadline allows temporal load balancing, which makes data traffic to be widely distributed over time and to leverage more unused WiFi connection opportunities.

IV. Analytical Framework

In this section, we develop a model-based simulation method as well as an analytic framework based on a queueing model to simply obtain offloading efficiency. The model-based simulation removes the necessity of detailed WiFi connectivity traces by abstracting the traces into closely matching distributions with few parameters. This method is helpful in predicting variation in offloading efficiency for the changes (e.g., additional deployment) in WiFi environment. Moreover, the framework referred to as a *queueing system with reneging and service interruptions*, models a user's data queue that switches the transmission interface between WiFi and cellular networks under a deadline. It provides a closed-form expression of offloading efficiency for some restricted cases. This theoretical framework is extremely helpful in predicting the offloading performance for diverse future WiFi deployments.

A. Model

Reneging: We consider a continuous queueing system with a single server (see Fig. 10). This queueing system models file generation and transfer from the phone of a user through cellular or WiFi networks. Files are generated and queued according to some stochastic process. The file size is assumed to follow a probabilistic distribution. A deadline, called a *reneging deadline*, is associated with each generated file. Files are serviced in the SRTF order depending on their remaining deadlines. A file can be serviced only via WiFi before its deadline. As the queueing system is continuous, it handles transmission at the bit level so that assigning a deadline to a file is equivalent to assigning the same deadline to each bit of the file. Any bits that have not finished transmission by its deadline will be removed from the queue and are assumed to be transmitted through cellular networks.

Service Interruption: Users move in and out of a WiFi coverage area. We model this time varying nature of the connection state by the ON and OFF states, where only in the ON state, the user is connected to a WiFi network. ON and OFF periods are random values selected from connection and interconnection time distributions in Figs. 6 and 7, respectively. We assume that the arrival process, file size distribution, and mobility process are pairwise independent. The "actual" service capacity of the server is determined by the length of ON and OFF states. For simplicity, we assume that the service capacity when the server is ON is constant. The queueing system introduced here is also referred to as a *queue with reneging and service interruptions* in this paper.

Challenges: To compute offloading efficiency $E_{\text{off}}(\tau)$ using our queueing model, we need to compute the stationary probability that data is removed from the queue before service when its deadline is τ, i.e.,

$$E_{\text{off}}(\tau) = 1 - \mathbb{P}[\text{waiting time in the queue} > \tau].$$

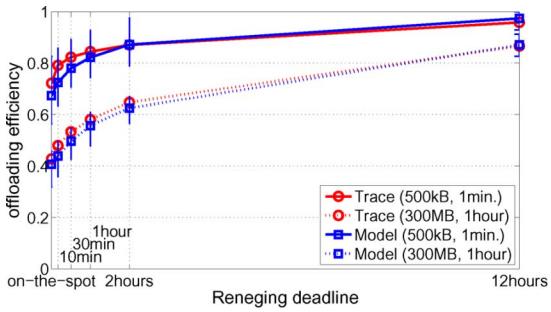


Fig. 20. Comparison of offloading efficiencies between trace-based and model-based simulations.

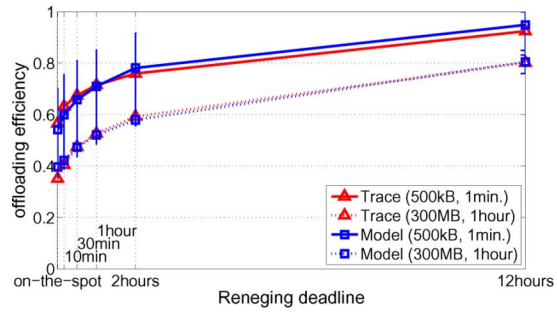


Fig. 21. Comparison of offloading efficiencies between trace-based and model-based simulations after random elimination of 20% WiFi APs.

The arrival process and file size distributions are in general quite diverse, depending on the target applications. We use a Poisson arrival process and exponential or heavy-tail file-size distributions, e.g., Weibull. In Section II-B, we showed that the distributions of connection times and interconnection times have good fit with truncated Pareto distributions, $\alpha < 1.2$.

The queueing system with reneging and service interruptions often obviates a tractable analysis if the statistical properties of the input parameters are as complex as heavy-tailed distributions. Nonetheless, this simple model is still useful for the following reasons. We can use this model to perform a *model-based simulation* in which the system parameter values such as the ON and OFF periods and the file arrival rates are selected from input distributions, which can be obtained from real traces or arbitrarily modeled. This simulation is much simpler than the trace-driven simulation we used in Section III, which needs to emulate user mobility from input mobility traces. Second, for some restricted set of the input distributions, we can obtain a closed-form solution, allowing us to predict offloading efficiency through simple numerical computation.

B. Trace-Based Versus Model-Based Simulation

To measure the accuracy of our model, we run a model-based simulation using the input distributions measured from the real traces. The input distributions obtained from the traces are the distributions of connection and interconnection times and the distribution of WiFi data rates. We vary the arrival process of files and the distribution of file sizes. The results are compared to those from the trace-driven simulations in Section III. Recall that from our MLE based fitting in Section II-B, the best-fit truncated Pareto parameters are $\alpha = 0.54$ for connection time and $\alpha = 0.63$ for interconnection times. The server capacity for the ON state is assumed to be 1.97 Mb/s, which is the average data rate from the real trace.

Fig. 20 compares offloading efficiency between the trace- and model-based simulations for two considerably different cases: 500-kB/min and 300-MB/h intensities with Poisson arrivals and exponential file sizes, and from short (10 min) to long (12 h) reneging deadlines. We observe that the results from the two simulations have minor difference, ranging from 0.1% to 10%. This difference comes from the imperfect fitting of truncated Pareto distribution to the actual connection and interconnection time distribution.

To test the accuracy of the model-based simulation, we test the offloading efficiency under the condition where 20% of WiFi APs in our traces are randomly removed using the technique in Section III-G. For the traces with 20% less WiFi APs, we obtain the best-fitting parameters $\alpha = 0.56$ and $\alpha = 0.59$ for interconnection and connection time distributions and compare the performances as shown in Fig. 21. We again observe less than 5% of performance differences between trace-driven simulations and model-based simulations.

C. Queue With Tractable Cases

Our continuous queueing model runs at the bit level in which an arrival of a file is modeled by a burst arrival of bits corresponding to the file size. The burst arrival is very hard to model in the continuous queueing model. Instead, we consider offloading at the file granularity. This makes the queueing system much more tractable, and with some restricted cases where all input distributions are exponential, we can obtain a closed-form solution. We assume that the file sizes, connection times, and interconnections follow exponential distributions with rates μ, θ_c, and θ_i, respectively. The arrival process of files follows a Poisson process with rate λ. Denote by c the capacity of the server when the server is ON.

We consider two types of reneging deadlines: *random* and *deterministic*. Random deadlines model a situation that users may set their own deadlines and are randomly picked from an exponential distribution with mean τ. Deterministic deadlines are typically set by each application to a constant. One side effect of considering files as the basic unit is that the deadline of a file may expire in the middle of the transfer. In such a case, we allow the file to continue to be transmitted through WiFi as long as the server state is ON.

Modeling Based on Markov Chain: The system can be described as a two-dimensional continuous-time Markov chain where the state is a tuple of $(Q(t), S(t))_{t\geq 0}$. $Q(t)$ is the number of files at time t and $S(t)$ is the state of the server (ON or OFF). Fig. 22 describes this Markov chain where

$$r_k = \begin{cases} \frac{k}{\tau}, & \text{random reneging} \\ \frac{\tau^{k-2}e^{-\mu\tau}}{\int_0^\tau t^{k-2}e^{-\mu t}}dt, & \text{deterministic reneging.} \end{cases} \quad (1)$$

For random reneging deadlines, the memoryless property of exponential random variables simplifies the solution. The transitions between ON and OFF occur at rates θ_c and θ_i. The number of files in the queue decreases due to reneging as well as service, both of which are proportional to the number of files at that time. A similar problem was studied in the queueing community [34] in the context of the machine-repair problem with impatient customers.

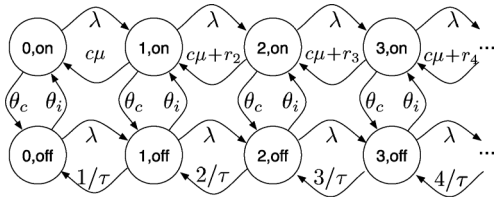


Fig. 22. 2-D Markov chain for the queueing system with reneging and service interruptions for exponential interarrival times with rate λ, file size with rate $c\mu$, contact time with rate θ_c, and the intercontact time with rate θ_i.

For deterministic deadlines, the system is not clearly Markovian. Thus, it is challenging to analyze the system. There exist a few studies on deterministic reneging, see, e.g., [9] and [16]. However, they deal only with the systems without service interruptions. We develop a tractable method to analyze our system by extending [9] and [34]. Note that the stationary distribution of a non-Markovian system is equivalent to that of the Markovian system in Fig. 22. This notion is extremely powerful since it allows to obtain a closed-form stationary solution even for the case of deterministic deadlines. We now elaborate on this by treating ON and OFF states separately.

1) *ON:* The transition between states, say, from (k, ON) to $(k-1, \mathrm{ON})$, is made by either finishing the transfer of a file in the queue or after its deadline expiration. The service rate is $c\mu$, whereas the rate of leaving the queue due to deadline expiration is complicated because the deadline is deterministic which requires recording the remaining deadline. Fortunately, in [9], such rates have been studied and are shown to be the same as (1).
2) *OFF:* The server sleeps during this state, and files will just leave the queue whenever the deterministic deadline τ expires at the rate of $1/\tau$. The evolution of the number of files in this queue is the same as that of the "customers" of the $M/D/\infty$ queue where the workload of all customers is τ. The workload processing in $M/D/\infty$ queue model can be interpreted as file drops in our model because, in both cases, deterministic τ is applied to all customers from the moment they entered the queue. It is known that $M/G/\infty$ has the *insensitivity* property to the workload distribution [16], i.e., the steady-state probabilities and output process are independent of the distribution of the customers' workload G.

Thus, the stationary distribution of the process $(Q(t), S(t))$ in the original system is equivalent to that computed from the Markovian system in Fig. 22. Markov-chain-based modeling facilitates the computation of stationary distributions, denoted by $\boldsymbol{\pi}(\tau) = [\pi_{ij}(\tau)]$, $i \in \{0, 1, \ldots\}$, $j \in \{\mathrm{ON}, \mathrm{OFF}\}$ for the reneging deadline τ. It turns out that the transition rate matrices of Fig. 22 have a block tridiagonal form and similar to those in quasi-birth-death (QBD) Markov process, allowing a variety of matrix-geometric techniques [22], [34]. We omit the resulting closed-form expression for brevity. The offloading efficiency $E_{\mathrm{off}}(\tau)$ can be obtained by deriving the ratio of the average leaving rates of files due to reneging over the average incoming rates, i.e.,

$$E_{\mathrm{off}}(\tau) = 1 - \frac{\sum_{k\geq 2} r_k \pi_{k,\mathrm{ON}}(\tau) + \sum_{k\geq 1} \frac{k}{\tau} \pi_{k,\mathrm{OFF}}(\tau)}{\lambda}.$$

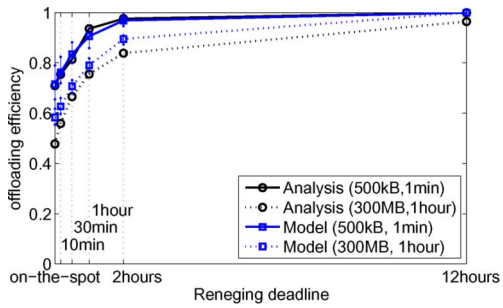


Fig. 23. Analytical result and simulation for exponential interarrivals times, service ON–OFF, and file-size distributions with deterministic reneging.

Fig. 23 compares the offloading efficiency between the theoretical analysis and the model-based simulation when deterministic reneging deadlines and exponentially distributed ON and OFF times are used. The figure verifies a good match between the two with the maximum difference of about 20%. The gap comes from the difference in the granularity of handling reneging. In the queueing simulation, reneging is applied at a bit unit, whereas in the theoretical model, it is applied at a file unit. Thus, in Fig. 23, the experiment with the smaller file size shows less error.

V. Related Work

Balasubramanian *et al.* [6] develop several techniques combining 3G networks and WiFi for reducing the total cost of data transfer. The proposed techniques are similar to Breadcrumb [23]. The authors use a citywide measurement data of 3G and WiFi network availability obtained from 20 transit buses in a city and war-driving in two other cities. The work focuses on gains achieved by on-the-spot offloading or delayed offloading with a very short delay deadline (up to 100 s). Based on these traces, they report about 10%–30% on-the-spot offloading efficiency and about 20%–33% offloading gain of delayed offloading over on-the-spot. Since their traces are taken during driving, they contain a lot of short connection and interconnection times with WiFi, which contribute to the substantial gains of delayed transfers with short deadlines (also low efficiency of on-the-spot offloading).

There are several more measurement studies focusing on WiFi and 3G network availability over given movement paths. Han *et al.* [17] suggest a two-pass measurement methodology involving rough search and detailed measurement phases for WiFi APs. Gass *et al.* [15] present a detailed measurement result by comparing the characteristics of 3G networks and WiFi in a city. Both of these results are based on war-driving by vehicles or by walk.

Ra *et al.* [26] present an online algorithm called SALSA over mobile smartphones with 3G/EDGE/WiFi interfaces that optimizes energy and delay tradeoffs using a Lyapunov optimization framework. SALSA is tested over real 3G/EDGE/WiFi measurement performed using 66 sample walk traces of about 1 h length in various areas including campus, shopping mall and airport. Balasubramanian *et al.* [7] present a different type of energy and delay tradeoffs arising from energy consumption characteristics of multimodal wireless terminal equipped with WiFi, 3G and GSM mobile network technologies. Based on a measurement study, they develop a energy consumption model

for each technology. The model is then used to design an algorithm that schedules (i.e., delays) transmissions to minimize the overall time spent in high-energy states (i.e., energy tail) while respecting user-specified delay-tolerance deadlines.

Nicholson *et al.* [23] propose a scheme that can predict near future WiFi connectivity and quality. The scheme enables mobile devices to schedule their data transfers to harness higher transmission rates of WiFi APs. It exploits users' tendency of following regular movement patterns around the region where static WiFi APs are deployed. The authors show that delaying transmissions according to short-term forecasts can achieve higher data rate as well as lower power consumption.

VI. Discussions and Caveats

The perhaps biggest surprise in our analysis is 65% traffic reduction currently achievable by on-the-spot WiFi offloading without use of any delay. Assuming most mobile data demands are from smartphone users, this gain is what the carriers are currently achieving. Roughly, it implies that about 35% of the projected 7 GB/month per-user usage in 2014 (about 2.5 GB) will be transferred through 3G networks. With additional incentives for delayed offloading, this gain can quickly grow. This means that from the user's perspective, with a fixed price plan of 2 GB/month over 3G networks (what is currently adopted by AT&T for iPhone 4G), average users do not overscribe at all. With more creative price plans for delayed transfers, users may even opt for a cheaper monthly data plan and can offload most of excess data traffic.

This paper focuses only on *temporal offloading*. However, allowing delays in applications also enables load balancing. End-to-end data rates at night are much higher than active hours (9:00 $\sim$ 24:00) because we tend to experience stable links overnight at home as well as less congestion in the backhaul network. Delayed transfer, especially with long delay deadlines, is likely to enable traffic dispersion over time so as to shift the high daytime demand for networking resources to the nighttime. Temporal and spatial asymmetry of WiFi connectivity can be further exploited to develop an adaptive offloading policy, which will improve offloading efficiency. As an example, when the time of the day is close to a typical hour when a user comes back to home, the deadline can be flexibly extended to exploit guaranteed WiFi connection opportunity at home as long as the users prefer to maximize offloading efficiency. Similarly, when users visit unknown places, the deadline can be adjusted to be short since the user may have limited WiFi connection opportunities in such places. The study of adaptive offloading policies is left as an interesting future work.

Our study makes a number of convenient assumptions. First, we assume that the measured data rates of WiFi in our traces are sustained independent of load in the network. Although the measured data rates account for traffic conditions (e.g., contention and dynamic data rates) existing at the time of connection, we ignore the issue of increased contention in the future as more users use WiFi offloading. Measuring and predicting the exact data rates for the future are very challenging. This factor depends on the tradeoff between capacity and demands offered by the current WiFi technology, which is still developing, so we do not have a clear answer for how we can incorporate the impact of the increased load on the performance of WiFi offloading. However, our results are still meaningful as they can be viewed as an upper bound on the performance gain since contention can only increase with more usage. In other words, our results are meaningful if the carriers can provision enough WiFi resource to sustain the current WiFi data rates.

The main focus of our study is purely performance-oriented. We ignore a number of technical and policy issues in our study. First, energy consumption is high if mobile devices constantly scan for WiFi connectivity. A number of solutions (e.g., [5], [11], and [32]) for this problem are developed. Rahmati and Zhong [5] proposed an intelligent energy-saving algorithm for predicting WiFi availability and device scans for WiFi APs only in areas where WiFi is likely available. Several researchers [11], [32] developed an energy-efficient location-tracking system for mobile phones based on map matching and war-driving or magnetometer and accelerometer sensor readings, which consume only a small fraction of power used for GPS. As users tend to maintain regular mobility patterns daily, mobile phones can perform scanning only when they are in a prerecorded area of WiFi stations. Although these techniques can save energy for WiFi discovery, it is not easy for them to guarantee full discovery. Therefore, in order to study the capacity region of WiFi offloading, we intentionally do not apply these techniques in DTap.

We also do not examine issues of security and administration or billing control. As user data are diverted away from the carriers' network, carriers may lose control over the data being offloaded. Despite these issues, we believe that the impact of our work is significant: Since our findings conclude that offloading is an effective means for accommodating the current and future traffic growth, our simulation tools can offer important guidance for network providers in deploying and upgrading their networks and also in designing successful and creative price plans. Given the strong performance advantages of WiFi offloading, we foresee that there will be technical solutions as well as policy and price restructuring to address these issues in the near future.

References

[1] "Urban tomography project," 2008 [Online]. Available: http://tomography.usc.edu/

[2] Cisco, San Jose, CA, "Cisco visual networking index: Global mobile data traffic forecast update, 2009–2014," Feb. 2010 [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html

[3] "Data, data everywhere," *Economist* Feb. 2010 [Online]. Available: http://www.economist.com/specialreports/displayStory.cfm?story_id=15557443

[4] A. Abhari and M. Soraya, "Workload generation for YouTube," *Multimedia Tools Appl.*, vol. 46, no. 1, pp. 91–118, 2010.

[5] R. Ahmad and Z. Lin, "Context-for-wireless: Context-sensitive energy-efficient wireless data transfer," in *Proc. ACM MobiSys*, 2007, pp. 165–178.

[6] A. Balasubramanian, R. Mahajan, and A. Venkataramani, "Augmenting mobile 3G using WiFi," in *Proc. ACM MobiSys*, 2010, pp. 209–222.

[7] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, "Energy consumption in mobile phones: A measurement study and implications for network applications," in *Proc. ACM SIGCOMM IMC*, 2009, pp. 280–293.

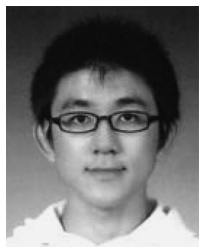
[8] A.-L. Barabasi, "The origin of bursts and heavy tails in human dynamics," *Nature*, vol. 435, pp. 207–211, May 2005.

[9] D. Y. Barrer, "Queuing with impatient customers and ordered service," *Oper. Res.*, vol. 5, no. 5, pp. 650–656, 1957.

[10] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on the design of opportunistic forwarding algorithms," in *Proc. IEEE INFOCOM*, 2006, pp. 1–13.
[11] I. Constandache, R. R. Choudhury, and I. Rhee, "Towards mobile phone localization without war-driving," in *Proc. IEEE INFOCOM*, 2010, pp. 1–9.
[12] W. T. Eadie, M. G. Roos, and F. E. James, *Statistical Methods in Experimental Physics*. Amsterdam, The Netherlands: Elsevier, 1971.
[13] H. Falaki, D. Lymberopoulos, R. Mahajan, S. Kandula, and D. Estrin, "A first look at traffic on smartphones," in *Proc. ACM IMC*, 2010, pp. 281–287.
[14] H. Falaki, R. Mahajan, S. Kandula, D. Lymberopoulos, R. Govindan, and D. Estrin, "Diversity in smartphone usage," in *Proc. ACM MobiSys*, 2010, pp. 179–194.
[15] R. Gass and C. Diot, "An experimental performance comparison of 3G and Wi-Fi," in *Proc. ACM PAM*, 2010, pp. 71–80.
[16] D. Gross, *Fundamentals of Queueing Theory*. Hoboken, NJ: Wiley, 2008.
[17] D. Han, M. Kaminsky, A. Agarwala, K. Papagiannaki, D. G. Andersen, and S. Seshan, "Mark-and-Sweep: Getting the "inside" scoop on neighborhood networks," in *Proc. ACM IMC*, 2008, pp. 99–104.
[18] T. Kaneshige, "AT&T iPhone users irate at idea of usage-based pricing," *PCWorld* Dec. 2009 [Online]. Available: http://www.pcworld.com/article/184589/atandt_iphone_users_irate_at_idea_of_usagebased_pricing.html
[19] T. Karagiannis, J.-Y. L. Boudec, and M. Vojnovic, "Power law and exponential decay of inter contact times between mobile devices," in *Proc. ACM MobiCom*, 2007, pp. 183–194.
[20] K. Lee, I. Rhee, J. Lee, Y. Yi, and S. Chong, "Mobile data offloading: How much can WiFi deliver?," in *Proc. ACM CoNEXT*, 2010, pp. 425–426.
[21] S. C. Networking, "Correlations between ftp and ping," 2011 [Online]. Available: http://www.slac.stanford.edu/comp/net/wan-mon/tutorial.html
[22] M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models*. Baltimore, MD: Johns Hopkins Univ. Press, 1981.
[23] A. J. Nicholson and B. D. Noble, "BreadCrumbs: Forecasting mobile connectivity," in *Proc. ACM MobiCom*, 2008, pp. 46–57.
[24] U. Paul, A. P. Subramanian, M. M. Buddhikot, and S. R. Das, "Understanding traffic dynamics in cellular data networks," in *Proc. IEEE INFOCOM*, 2011, pp. 882–890.
[25] G. Prodhan and G. E. McCormick, "S. Korea still top world communications economy," Sep. 2011 [Online]. Available: http://af.reuters.com/article/maliNews/idAFL5E7KF3SH20110915
[26] M.-R. Ra, J. Paek, A. B. Sharma, R. Govindan, M. H. Krieger, and M. J. Neely, "Energy-delay tradeoffs in smartphone applications," in *Proc. ACM MobiSys*, 2010, pp. 255–270.
[27] M. Reardon, Cisco predicts wireless-data explosion"," Feb. 2010 [Online]. Available: http://news.cnet.com/8301-30686_3-10449758-266.html
[28] I. Rhee, M. Shin, S. Hong, K. Lee, and S. Chong, "On the levy walk nature of human mobility," in *Proc. IEEE INFOCOM*, 2008, pp. 924–932.
[29] W. Rothman, "The definitive coast-to-coast 3G data test," Dec. 2008 [Online]. Available: http://gizmodo.com/5111989/the-definitive-coast+to+coast-3 g-data-test
[30] A. Sharma, V. Navda, R. Ramjee, V. N. Padmanabhan, and E. M. Belding, "Cool-Tether: Energy efficient on-the-fly wifi hot-spots using mobile phones," in *Proc. ACM CoNEXT*, 2009, pp. 109–120.
[31] J. Strauss, D. Katabi, and F. Kaashoek, "A measurement study of available bandwidth estimation tools," in *Proc. ACM IMC*, 2003, pp. 39–44.
[32] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson, "VTrack: Accurate, energy-aware road traffic delay estimation using mobile phones," in *Proc. ACM SenSys*, 2009, pp. 85–98.
[33] T. F. Vania Conan and J. Leguay, "Characterizing pairwise inter-contact patterns in delay tolerant networks," in *Proc. Int. Conf. Auton. Comput. Commun. Syst.*, 2007, Article no. 19.
[34] K.-H. Wang and Y.-C. Chang, "Cost analysis of a finite M/M/R queueing system with balking, reneging, and server breakdowns," *Math. Methods Oper. Res.*, vol. 56, no. 2, pp. 169–180, 2002.
[35] X. Zhuo, W. Gao, G. Cao, and Y. Dai, "Win-Coupon: An incentive framework for 3G traffic offloading," in *Proc. IEEE ICNP*, 2011, pp. 206–215.

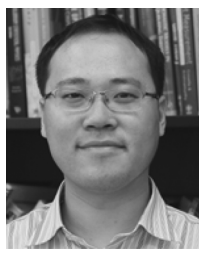
Kyunghan Lee (S'07–A'10) received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2002, 2004, and 2009, respectively.

He is currently an Assistant Professor with the School of Electrical and Computer Engineering, Ulsan National Institute of Science and Technology (UNIST), Ulsan, Korea. Prior to joining UNIST, he was with the Department of Computer Science, North Carolina State University, Raleigh, as a Senior Research Scholar. His research interests include the areas of human mobility modeling, delay-tolerant networking, information-centric networking, context-aware service design, and cloud-powered network service design.



Joohyun Lee (S'11) received the B.S. degree in electrical engineering and computer science from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2008, and is currently an integrated M.S. and Ph.D. candidate at KAIST.

His research interests are in the areas of mobility-aware networks, delay-tolerant networks, and network economics.



Yung Yi (S'04–M'06) received the B.S. and M.S. degrees in computer science and engineering from Seoul National University, Seoul, Korea, in 1997 and 1999, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Texas at Austin in 2006.

From 2006 to 2008, he was a Post-Doctoral Research Associate with the Department of Electrical Engineering, Princeton University, Princeton, NJ. Now, he is an associate professor at the Department of Electrical Engineering at KAIST, South Korea. His current research interests include the design and analysis of computer networking and wireless systems, especially congestion control, scheduling, and interference management, with applications in wireless ad hoc networks, broadband access networks, economic aspects of communication networks economics, and greening of network systems.

Dr. Yi has been serving as a TPC member at various conferences including ACM MobiHoc, Wicon, WiOpt, IEEE INFOCOM, ICC, GLOBECOM, and ITC. His academic service also includes the Local Arrangement Chair of WiOpt 2009 and CFI 2010, the Networking Area Track Chair of TENCON 2010, the Publication Chair of CFI 2010, and a Guest Editor of the Special Issue on Green Networking and Communication Systems of *IEEE Surveys and Tutorials*. He also serves as the Co-Chair of the Green Multimedia Communication Interest Group of the IEEE Multimedia Communication Technical Committee.

Injong Rhee (S'89–M'94) received the Ph.D. degree in computer science from the University of North Carolina at Chapel Hill.

He is a Professor with the Department of Computer Science, North Carolina State University, Raleigh. His areas of research interests include computer networks, congestion control, wireless ad hoc networks, and sensor networks.

Prof. Rhee is an Editor of the IEEE TRANSACTIONS ON MOBILE COMPUTING.

Song Chong (M'93) received the B.S. and M.S. degrees in control and instrumentation engineering from Seoul National University, Seoul, Korea, in 1988 and 1990, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Texas at Austin in 1995.

Since March 2000, he has been with the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, where he is a Professor and was the head of the Communications and Computing Group. Prior to joining KAIST, he was a Member of Technical Staff with the Performance Analysis Department, AT&T Bell Laboratories, Holmdel, NJ. He has published more than 100 papers in international journals and conferences. His current research interests include wireless networks, future Internet, and human mobility characterization and its application to mobile networking.

He is an Editor of *Computer Communications* and the *Journal of Communications and Networks*. He has served on the Technical Program Committee of a number of leading international conferences including IEEE INFOCOM and ACM CoNEXT. He serves on the Steering Committee of WiOpt and was the General Chair of WiOpt 2009.