OPPORTUNISTIC CONTENT PUSHING VIA WIFI HOTSPOTS

Tao Han, *Student Member, IEEE,* and Nirwan Ansari, *Fellow, IEEE*
Advanced Networking Laboratory
Department of Electrical and Computer Engineering
New Jersey Institute of Technology, Newark, NJ, 07102, USA
Email: {th36, nirwan.ansari}@njit.edu

***Abstract*—Metro WiFi networks can be potentially leveraged to offload data traffic from congested cellular networks. In this paper, we design a content pushing system which pushes the content to mobile users through opportunistic WiFi connections. The system responds a user's pending requests or predicted users' future requests, codes these requested contents by using Fountain codes, predicts the user's routes, and prelocates the coded content to the WiFi access points along the user's routes. The content prelocation scheme is based on an order-2 Markov predictor and the connection duration statistics. Through simulations, we show that a significant amount of content can be delivered to mobile users through the proposed system.**

***Keywords*—WiFi Networks, Content Delivery Networks, Wireless Networking.**

I. Introduction

With the rapid development of wireless access techniques and user devices, Internet applications are gradually moved to wireless networks. According to the mobile data traffic measurements from a Korea cellular network operator, the monthly average mobile data traffic was tenfold in 2010 as compared to that of 2009 [1]. The mobile data traffic volume is expected to increase by 26-fold between 2010 and 2015 [2]. The significant data traffic increase may congest the mobile network, and lead to long latency in content delivery. Offloading data traffic from congested cellular networks is a promising method to reduce congestion in cellular networks. The existing cellular data traffic offloading mechanisms can be classified into two categories. The first category is relying on the mobile users to disseminate the content. MADNet [3], [4] is designed based on this idea. Given a set of mobile users who request some content from the servers, MADNet selects a subset of users and deliver the content to them through cellular networks. The subset of users further disseminate the content through opportunistic ad-hoc communications to those users outside the subset. The key designing issue in MADNet is how to select the subset of users in order to maximize the traffic offloading. The authors proposed strategies to select the subset of users based on either users' activities or mobilities. Delay of the content delivery in MADNet is not guaranteed. To meet the delay requirements, Whitbeck *et al.* [5] proposed to push the content to the users through cellular networks when ad-hoc communications fail to deliver the content within some target delay. Mayer *et al.* [6] applied the similar idea to mobile to mobile message delivery rather than base station to mobile message delivery. The authors proposed a routing scheme which allows the message routes within delay tolerant networks (DTNs) as long as the probability of the message that can be delivered through cellular networks is high. If the DTN is unlikely to deliver the message within the deadline, the message will be routed to cellular networks. Li *et at.* [7] analyzed the mobile data offloading problem mathematically. They formulated the offloading problem as a submodular function maximization problem, and proposed several algorithms to achieve the optimal solution. To encourage mobile users participate in traffic offloading, Zhou *et al.* [8] proposed an incentive framework that motivate users to leverage their delay tolerance for cellular data offloading. Mashhadi *et al.* [9] proposed a proactive caching mechanism for mobile users in order to offload the cellular traffic. When the local storage does not have the request content, the proactive caching mechanism will set a target delay for this request, and explores the opportunities to retrieve data from the neighboring mobile nodes. The proactive caching mechanism requests data from the cellular networks when the target delay is violated.

The other category is to explore the metro scale WiFi networks to offload cellular data traffic. Lee *et al.* [10] showed from their measurement results that a user is in WiFi coverage for 70% of the time on average, and if users can tolerate a two hour delay in data transfer, the network can offload about 70% of cellular traffic to WiFi networks. Deshpande *et al.* [11] compared the performance of 3G and metro scale WiFi for vehicular network access and showed that even though suffering frequent disconnections, WiFi delivers high throughput when connected. Gass *et al.* [12] tested the communication possibility with a WiFi access point while the user is moving, and verified through an experimental evaluation that a significant amount of the data can be downloaded and uploaded through opportunistic WiFi connections [13]. Gass *et al.* [13] pointed out that the full potential of WiFi access points in term of the transmission rate is not fully actualized and is limited to the rate of the backhaul connection to which the access point is connected. To eliminate backhaul bottlenecks, Gass *et al.* [14] proposed in-motion proxy and data rendezvous protocol to enable downloads of a large amount of data during short period opportunistic WiFi connections. The in-motion proxy prefetches a user's requesting data from the original server to its local cache, and streams the data to the user when they are connected to WiFi access points. The in-motion proxy

978-1-4673-2204-1/12/$31.00 ©2012 IEEE

enables large data transfers to be completed with several short connection durations. The data rendezvous protocol eliminates the bottlenecks between the in-motion proxy and WiFi access points through access point prediction and selection.

In this paper, we rely on metro WiFi networks to push content to mobile users. The proposed content delivery system aims to push content to users through opportunistic WiFi connections. The system prepares the content for the users based on their demands or the system's predictions, traces the users' locations and prelocates the content to the WiFi access points along users' routes, and estimates the content delivery process. The proposed content delivery system is different from In-Motion proxy [14] in several aspects. First, In-Motion proxy is designed to eliminate the backhaul bottleneck while transmitting data using WiFi networks. Our work is to design a content delivery system that takes advantages of metro WiFi to push content to mobile users. Second, In-Motion proxy assume the availability of GPS and routes information to predict the access points that the users will visit in the future, our work applies order-2 (O(2)) Markov predictor [15] to analyze users' traces and predict users' locations and next movements. Third, In-Motion proxy mentions that it selects WiFi access points that may provide longer connection durations and higher capacity wireless link. However, the authors did not present their algorithm about the access point selection. Our proposed content delivery system monitors the average connection duration at each WiFi access point, and bases on these historical information to predict the connection duration of users at individual WiFi access points. We present the design of the proposed content delivery system in the following section.

II. Metro WiFi Network Measurements

To understand the performance of a metro WiFi network, we execute two network measurements. In the first measurement, we locate WiFi hotspots and test their downlink and uplink speed and round trip time (RTT). Table I shows the network measurement results. The average downlink and uplink speed are 4086 kbps and 2024 kbps, respectively. The average RTT is 81 ms. The performance of the metro WiFi network is comparable with that of commercial cellular networks. In the

TABLE I
WiFi Network Speed Test

Download (kps)	Upload (kps)	RTT (ms)
4086	2424	81

second measurement, we implement a content downloader on an android smart phone, and drive through a downtown area with a dense metro WiFi deployment to collect the data trace. When there are WiFi connections, the downloader will automatically establish connections to download content from remote servers. We collect data traces using tcpdump. We drive the car at a speed varying from 20 mph to 30 mph. Fig. 1 shows the downlink throughput of a metro WiFi network to the android smart phone in the moving car. If the smart phone is able to establish a connection with metro WiFi hotspots, it can download content at a speed of about 400 Kbps. Since metro WiFi hotspots are not seamlessly deployed, the smart phone may lose WiFi connections when it moves outside the coverage area of WiFi hotspots. Therefore, during the time period that the smart phone is outside the coverage, the network throughput is zero. In addition, we calculate the

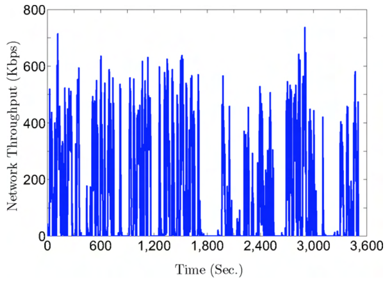

Fig. 1. WiFi network donwlink throughput.

cumulative function of the number of transmitted packets, the number of out sequence packets, and the number of retransmissions. The number of successful transmitted packets per second is less than 50 at most of the time. The maximum number of retransmissions per second is around 20, and in most of the time, the number of retransmissions is from 2 to 10. The packet retransmissions indicate excessive packet loss on the network.

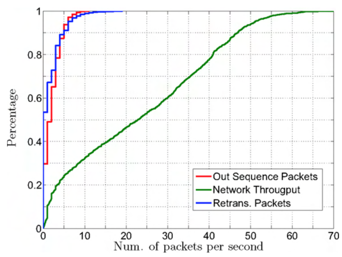

Fig. 2. Cumulative distribution function.

From the measurement study, we find that 1) metro-WiFi networks are capable to deliver content to mobile users via opportunistic WiFi connections; 2) the opportunistic WiFi links are less reliable and may experience excessive packet loss.

III. Mobile Content Delivery System

The proposed mobile content delivery system takes advantages of opportunistic WiFi connections to deliver content to mobile users. The proposed system is a server or proxy side program that can track the users' locations, predict users' routes, and push content to the users. The content delivery system includes four modules as shown in Fig. 3. The first module is the content preparation module which prefetches or stores the content that will be pushed to the users. The second module is the content precoding module which encodes the content using digital Fountain codes to enhance the reliability of the delivered content. The third module is the content pushing module which predicts the users' roaming routes, finds out the access points which are most likely to be accessed by individual mobile users, and prelocates the coded content to these access points. The fourth module is content delivery estimation module which estimates whether the delivering content is successfully delivered, and signals the content queue to feed new content to the content precoding module upon completion of the current content delivery. In this section, we presents the design of the proposed content delivery system.

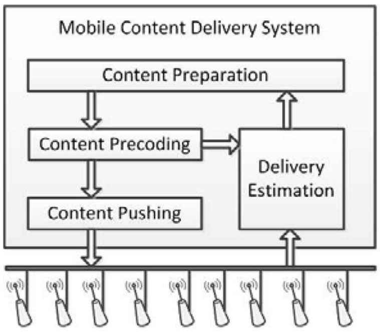

Fig. 3. Mobile content delivery system.

A. Content Preparation Module

The content preparation module determines what content to be pushed to the users. This module first checks whether the users have pending requests for some content. If so, this module fetches the corresponding content and prepares to push them to the users. If there is no pending requests, the content preparation module fetches the most popular content based on its log history information, and feeds them to the content precoding module.

B. Content Precoding Module

The content precoding module applies digital fountain codes to encode the on demand content to enhance the reliability of the content delivery. The proposed content delivery system relies on WiFi hotspots deployed near the intersections of roads to push the content to mobile users. It is challenging to deliver content though WiFi hotspots to mobile users. The duration that individual mobile users connect to individual WiFi access points may be too short to download all the content. The current WiFi coverage is not seamless. Therefore, the users has to disconnect from the WiFi connection when they roam out of coverage of the WiFi access point, and wait until there are WiFi access points available in their range. Thus, the wireless connection is unreliable in term of the connection duration. The server may have to rely on several access points to push the on demand content to individual mobile users. In addition, as shown in the network measurements, the packet loss is frequent due to wireless errors. The information about the lost packet is not fed back to the server since the server is trying to push the content to the user and does not require any feedback from the users. Therefore, when the user reconnects to the WiFi network, the server may not know how many packets the user has received correctly and what are the next packets it should push to the user. To solve this problem, we apply digital fountain codes to encode the content. Digital fountain codes are rateless codes which are suitable for delivering content with unknown wireless channel conditions. A fountain code encodes a set of n input symbols into a potentially limitless stream of output symbols. The original n input symbols can be recovered from any set of m output symbols. For well designed fountain codes, e.g., Raptor code [16], m is only slightly larger than n. By using digital fountain codes, the server can send the encoded symbols to the users when they resume their connections without being required to remember the point where the individual users lose connection. The users store the received coded symbols for further processing. The server can estimate the delivery process by counting the number of the transmitted symbols.

C. Content Pushing Module

The content pushing module is the core module of the proposed mobile content delivery system. This module performs two functions: 1) predicts the routes of the roaming users, and 2) prelocates the content to WiFi access points along the users' routes. Since our proposed content push server aims to push content through metro WiFi networks, we do not assume the availability of either cellular data for users' routes prediction [17] or GPS information such as the starting points, the destination and the routes from users' devices [14]. However, we assume the users are rational and follow their favorite routes in terms of shortest distance, shortest time or personal preferences most of the time. Therefore, it is possible to predict users' routes from users' historical routes information. Without users' roaming information, the content delivery system does not know which access points individual users will access, and it cannot prelocate the content unless it broadcasts the the content to all the access points which is a waste of bandwidth. The proposed content pushing module predicts users' routes based on users' current and historical WiFi hotspots connection information and the metro WiFi network deployment information. We assume the server has complete knowledge of WiFi deployment information and can map WiFi hotspots on the road network. It is a realistic assumption because we rely on the commercial WiFi hotspots whose deployments are well planned. According to the measurement in [12], the in-motion users are able to connect to WiFi access points near them even they are moving fast. The short period connection may not allow us to deliver content to users, but we can use this connection information to predict the user's next location. Song *et al.* [15] evaluated the accuracy of several routes predictors using WiFi traces and showed that the order-2 (O(2)) Markov predictor with a fallback to first order model when the second order model fails on prediction was the most accurate routes predictor. We thus apply this predictor for user path prediction. The O(k) Markov predictor assumes the user's next location can be predicted from the user's current location and the sequence of the k most recent locations in the location history. Assume a user's location history is $L = a_1 a_2 \cdots a_n$, and $L(i, j) = a_i a_{i+1} \cdots a_j$. Let the user's location be a random variable X, and $X(i, j) = X_i X_{i+1} \cdots X_j$ define the context $c = L(n + 1 - k, n)$; the Markov assumption is that the user's next location, X, follows:

$$
\begin{aligned}
&P(X = a|X(1, n) = L) \\
&= P(X = a|X(n + 1 - k, n) = c) \\
&= P(X_{i+k+1} = a|X(i + 1, i + k) = c). \qquad (1)
\end{aligned}
$$

In other words, the probability depends only on the k most recent locations and the probability is the same when the context is the same. The probability can be estimated as

$$\widehat{P}(X_{n+1} = a|L) = \frac{N(ca, L)}{N(c, L)}. \qquad (2)$$

Here, $N(s^{'}, s)$ denotes the occurrences of the substring $s^{'}$ appeared in string s. The proposed content pushing module applies the O(2) Markov predictor to predict a user's next location, and uses the O(1) predictor when the O(2) Markov predictor fails to provide a prediction. Our system uses WiFi access point index to generate the state of Markov process from users' mobility history, and updates the transition probabilities according to Eq. 3. For example, one of the user's historical routes is shown in Fig. 4. The states and transition

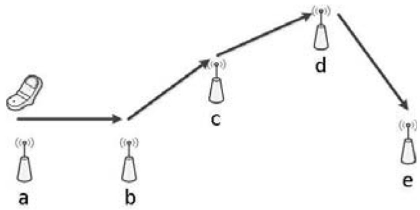

Fig. 4. An example of a user's historical route.

probability updates are shown in Table II.

TABLE II
STATES AND TRANSITION PROBABILITY UPDATES

States	Probability Updates
(-,a)	N/A
(a,b)	$\widehat{P}(X = b\|(-, a))$
(b,c)	$\widehat{P}(X = c\|(a, b))$
(c,d)	$\widehat{P}(X = d\|(b, c))$
(d,e)	$\widehat{P}(X = e\|(c, d))$

The O(2) Markov predictor provides good estimation of the user's next locations. However, it is not clear how much content can be delivered when the user connects to the WiFi access point in the predicted location. We integrate the connection duration database with the O(2) Markov location predictor to forecast the amount of content that can be delivered. We only consider the connection durations because the amount of content that can be delivered through opportunistic connection is highly dependent on the connection duration [12]. However, our model can be easily extended to consider the connection quality by integrating access point quality information [18].

The content pushing module monitors and updates the average connection duration of WiFi access points. The connection durations information reflects the traffic condition near the access points. The longer the average connection period, the heavier the traffic near the access point. Therefore, a longer connection duration is expected when a user arrives at the access point. Given the connection duration information, the content pushing module predicts the amount of content that should be prelocated to the access point. The average connection duration at access point, j, is calculated as

$$T_j = \frac{\sum_{i=1}^{N}(t_{i,j})}{N}, \qquad (3)$$

Here, T_j is the average connection duration at access point j, $t_{i,j}$ is the connection duration of user i at access point j, and N is the total number of users who have accessed the access point. Assume all the access points have the same downlink rate, r, then the amount of content that should be prelocated to an access point is the product of the downlink rate r and connection duration T.

D. Delivery Estimation Module

The delivery estimation module estimates whether the object is successfully delivered based on the user connection information feedback from the access points and the coding information from the content precoding module. If the current delivery is complete, the delivery estimation module signals the content preparation module to feed the next content to the content precoding module. The delivery estimation module estimates the number of delivered objects n_d according to the data rates, r, of the access points and the user's actual connection period, $t^{'}$:

$$n_d = \frac{rt^{'}}{q} \qquad (4)$$

Here, q is the symbol size. If n_d is not smaller than m, which is the number of symbols required to decode the packet, the delivery estimation module assumes that the object is successfully delivered.

IV. SIMULATION

We set up the simulation with 400 WiFi access points. Each access point indicates an intersection of the roads. We select one source access point where the user first accesses the Internet, and three destination nodes. We generate the user's routes trace by simulating the user's travel from the source node to the destination nodes for 300 times. Assume the user follows the shortest path routes from the source node to destination node for 90 percent of his/her travels, and follows other routes for 10 percents of his/her travels.

Based on the generated trace, we apply the O(2) Markov predictor to predict a user's routes during the simulation, and monitor the connection duration of the user at each access point. According to the connection duration history, we forecast the future connection duration at each access point. We assume the downlink rate of WiFi access points is 10 Mbits per second. We compare the total amount of content prelocated to the WiFi access points and the total amount of content pushed to the mobile users of four different schemes. The first scheme is the original scheme (ORI) which prelocates the content to all the neighboring access points of the user's current accessing point. The amount of the content prelocated to the access point is the product of the downlink rate of the access point and a predefined connection duration which is 30 second in the simulation. The second scheme is referred to as the most visit scheme (MV) which prelocates the content to the most visited nodes according to the O(2) Markov predictor. The amount of the prelocated content equals to the access point downlink rate multiplied by 30 seconds. The third scheme is MV with connection time estimation (MV_TS) which prelocates the content to the most visited nodes. The amount of content equals to the product of an estimated connection duration and the downlink rate of the access point. The fourth scheme is a hybrid prelocation scheme (HYB) which prelocates the content to the most visited nodes and the neighboring nodes with the largest estimated connection durations. The amount of content is calculated by using the same method as the third scheme. The proposed content delivery system uses HYB to prelocate the content. Fig. 5 shows the amount of content prelocated to

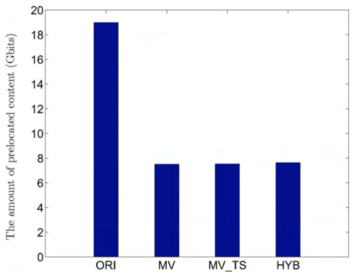

Fig. 5. The amount of prelocated content

the access points. We can see from the figure that the MV, MV_TS, and HYB save more than 50 percent bandwidth as compared to the original scheme. Fig. 6 shows the amount

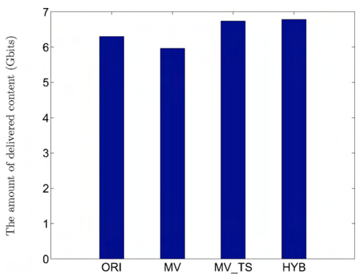

Fig. 6. The amount of delivered content

of content delivered to the users. HYB and MV_TS have the largest amount of content delivered to the users because both of them consider the historical connection duration of each access point in prelocating the content to the access point.

V. Conclusion

In this paper, we have presented the design of a content delivery system which leverages on metro-WiFi networks to push content to the mobiles. Through simulations, we have demonstrated that a significant amount of content can be delivered to the users through the proposed content delivery system.

Acknowledgment

This paper was invited for presentation at IC-NIDC 2012 with the support from the 111 Project of China under Grant No. B08004.

References

[1] Y. Choi, H. W. Ji, J. yoon Park, H. chul Kim, and J. Silvester, "A 3w network strategy for mobile data traffic offloading," *Communications Magazine, IEEE*, vol. 49, no. 10, pp. 118 –123, Oct. 2011.

[2] Cisco System Inc., "Cisco visual networking index: Global mobile datatraffic forecast update, 2010-2015," Feb. 2011. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.pdf

[3] B. Han, P. Hui, V. A. Kumar, M. V. Marathe, G. Pei, and A. Srinivasan, "Cellular traffic offloading through opportunistic communications: a case study," in *Proceedings of the 5th ACM workshop on Challenged networks, CHANTS' 10*, Sep. 2010.

[4] B. Han, P. Hui, and A. Srinivasan, "Mobile data offloading in metropolitan area networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 14, no. 4, pp. 28–30, Nov. 2010.

[5] J. Whitbeck, M. Amorim, Y. Lopez, J. Leguay, and V. Conan, "Relieving the wireless infrastructure: When opportunistic networks meet guaranteed delays," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium on a*, Lucca, Italy, Jun. 2011.

[6] C. Mayer and O. Waldhorst, "Offloading infrastructure using delay tolerant networks and assurance of delivery," in *Proceedings of IFIP Wireless Days*, Niagara Falls, ON, Canada, Oct. 2011.

[7] Y. Li, G. Su, P. Hui, D. Jin, L. Su, and L. Zeng, "Multiple mobile data offloading through delay tolerant networks," in *Proceedings of the 6th ACM workshop on Challenged networks, CHANTS' 11*, Sep. 2011.

[8] X. Zhuo, W. Gao, G. Cao, and Y. Dai, "Win-coupon: An incentive framework for 3g traffic offloading," in *Network Protocols (ICNP), 2011 19th IEEE International Conference on*, Vancouver, BC Canada, Oct. 2011.

[9] A. Mashhadi and P. Hui, "Proactive caching for hybrid urban mobile networks," 2010. [Online]. Available: http://www.cs.ucl.ac.uk/research/researchnotes/documents/RN_10_05_000.pdf

[10] K. Lee, I. Rhee, J. Lee, Y. Yi, and S. Chong, "Mobile data offloading: how much can wifi deliver?" *SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. 425–426, Aug. 2010.

[11] P. Deshpande, X. Hou, and S. R. Das, "Performance comparison of 3g and metro-scale wifi for vehicular network access," in *Proceedings of the 10th annual conference on Internet measurement, IMC '10*, Melbourne, Australia, Nov. 2010.

[12] R. Gass, J. Scott, and C. Diot, "Measurements of in-motion 802.11 networking," in *Proceedings of the Seventh IEEE Workshop on Mobile Computing Systems & Applications*, Washington, DC, USA, Aug. 2006.

[13] R. Gass and C. Diot, "An experimental performance comparison of 3g and wi-fi," in *Proceedings of the 11th international conference on Passive and active measurement, PAM'10*, Zurich, Switzerland, Apr. 2010.

[14] ——, "Eliminating backhaul bottlenecks for opportunistically encountered wi-fi hotspots," in *Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st*, Taipei, Taiwan, May 2010.

[15] L. Song, D. Kotz, R. Jain, and X. He, "Evaluating next-cell predictors with extensive wi-fi mobility data," *Mobile Computing, IEEE Transactions on*, vol. 5, no. 12, pp. 1633 –1649, Dec. 2006.

[16] M. Luby, A. Shokrollahi, M. Watson, and T. Stockhammer, " Raptor Forward Error Correction Scheme for Object Delivery," 2007. [Online]. Available: http://tools.ietf.org/html/rfc5053

[17] K. Laasonen, "Clustering and prediction of mobile user routes from cellular data," in *Knowledge Discovery in Databases: PKDD 2005*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2005, vol. 3721, pp. 569–576.

[18] A. J. Nicholson and B. D. Noble, "Breadcrumbs: forecasting mobile connectivity," in *Proceedings of the 14th ACM international conference on Mobile computing and networking, MobiCom '08*, San Francisco, California, USA, Sep. 2008.