Rich Content Sharing in Mobile Systems using Multiple Wireless Networks

Ngoc Do
Information and Computer Science
University of California, Irvine, USA
nmdo@ics.uci.edu

Nalini Venkatasubramanian
Information and Computer Science
University of California, Irvine, USA
nalini@ics.uci.edu

ABSTRACT

There has been an increasing popularity of applications deployed on mobile devices, such as smartphones or tablets. Many of them, e.g., YouTube [1], Pandora [2], Facebook [3] and etc, require access to the Internet for content sharing while running, and contribute a huge amount of data traffic sent through cellular networks [9], which causes cellular networks currently to be overloaded. Moreover, it is predicted that mobile data traffic will increase very fast in the next few years [9]. As a result, many cellular network providers are putting a lot of effort to seeking solutions for improving their network capacity, e.g., upgrade their infrastructure, as well as decide to move away from unlimited data plans to less flexible charging models [4]. In this paper, we address the problem of efficient rich content sharing from/to mobile devices by proposing practical approaches that provide high delivery performance, reduce cellular data traffic, and release the pressure of cellular networks' heavy load on mobile users and cellular network services providers. Our approaches [13–16] all share a common technique: using complementary networks, such as WiFi, WiFi ad hoc or Bluetooth, equipped in most modern mobile devices to offload data traffic previously planned to be transmitted over cellular networks. For each proposed approach, we prove its feasibility by testing it on an Android based testbed and evaluate its performance and scalability using simulations.

Categories and Subject Descriptors: C.2.1 [Computer-Communication Networks]: Network Architecture and Design

General Terms: Algorithm, Design, Experiment, Performance

1. INTRODUCTION

Recent market forecasts from Cisco [9] predict that mobile data traffic will increase 18 times over a span of five years (2011-2016) while mobile network connection speed will increase only 9-fold by 2016. Cellular service providers have already had a hard time to keep up with the staggering increase in data traffic [5,6], and will have to carefully engineer their networks of supporting the tremendous amount of mobile data traffic in the future. Cellular service providers may partially cope with the capacity issue by: (i) deploying more base stations, (ii) upgrading their current networks to new technologies, e.g., 4G networks LTE, or (iii) upgrading their base stations to support multicast/broadcast. However, these solutions incur high infrastructure costs and may not be compatible with current mobile devices. For example, currently the latest iPhone version, iPhone 4S, cannot access 4G networks. Therefore, a more efficient and low cost solution is desirable.

Nowadays, in addition to cellular network interface, modern mobile devices are equipped with other network technologies such as WiFi, WiFi ad hoc, Bluetooth or even a recent communication technology - WiFi Direct [7]. Cellular service providers thus have chance to offload cellular data traffic to an *auxiliary* network. A straightforward method that have been widely employed in mobile devices to offload cellular data traffic is enabling WiFi access, rather than using cellular connection, whenever a WiFi AP is found and a connection to the WiFi AP is successfully established. Offloading technique is used not only for data transfer, but also for phone call. For example, T-Mobile has recently deployed a service, WiFi Calling, allowing mobile users to make phone calls over WiFi to support customers who usually suffer from weak cellular signal strength or disconnection to a base station [11].

Cellular data traffic offloading has been studied in literature [12, 17, 19–23, 25]. The key idea is that cellular base station transmits data to mobile devices which are then relay the receiving data to other mobile devices over multi-hop ad hoc paths. Mobile devices which receive data over the cellular network and relay data over the ad hoc network is referred to as *proxies*. Existing works propose approaches for proxy selection such that the network throughput [12,17,23], or utilities are maximized [20].

Our work is different from previous studies in that most exiting approaches [12, 17, 20–22, 25] assume that cellular base stations can support multicast/broadcast, and we focus on rich content sharing because it is predicted that more than 66% of mobile data traffic's increase is due to rich content traffic [9]. Although current wireless network technologies can enable multicast/broadcast features at cellular base stations, using cellular multicast/broadcast for rich content dissemination is not efficient and practical because in multicast/broadcast modes, data is transmitted only at basic data rates to avoid packet loss. Therefore, currently cellular

Middleware 2012 Doctoral Symposium, December 3, 2012, Montreal, Quebec, Canada.

network services providers use only these features for short message dissemination (less than 90 characters) in emergency situations. Our approaches to delivering rich content from/to mobile device use only unicast that has currently been supported well by any cellular network technologies. We study and develop approaches for two application classes in our work:

1. Collaborative sharing: Applications belonging to this class share the *same* rich content to mobile devices (i.e., in other words, mobile devices are interested in the same content) over hybrid wireless networks. Thus, mobile devices have motivation to collaborate for content distribution. We focus on two types of data distributed: (i) delay sensitive (encoded live videos) and (ii) non-delay sensitive (images, large files, chunk based videos, etc).

2. Non-collaborative sharing: Applications in this class share rich content from/to mobile devices which are not interested in the same content. A mobile device does not have motivation to help other devices for content sharing due to limited data plan and energy consumption. We study and develop incentives that motivate mobile devices to collaboratively disseminate the content.

In all approaches we develop, we evaluate their feasibility by running experiments on an Android based testbed, and their scalability and performance by running simulations using a well-known commercial network simulator, Qualnet.

2. COLLABORATIVE SHARING

In this section, we consider two applications in which mobile devices collaboratively share rich content *interested in by all participants*. The first application deals with *delay sensitive* data - live video streaming. In this application, we propose approaches to streaming live videos encoded by a novel scalable video coding standard suitable for wireless networks from a streaming server located in Internet to mobile devices. Two wireless networks, cellular and WiFi ad hoc, are employed to deliver live videos optimally in terms of video quality perceived at all mobile users. The second application is used for disseminating large rich contents that are *non-delay sensitive* including large files, images, or even videos (chunk-based) to a large number of mobile devices. We pursue two metrics in this application: high reliability (the whole content should be fully received by all devices) and low latency (the whole content should be fast received).

2.1 Live Video Streaming

We consider scenarios where a streaming server receives live events (e.g., live soccer games) from some media sources, save them into videos, and disseminate the videos to mobile devices. In our work [13, 14], we employ a hybrid cellular and ad hoc network to deliver live videos to mobile users because cellular networks are not suitable to large-scale video dissemination. A measurement study [18] shows that each HSDPA cell can only support up to 6 mobile video users at 256 Kbps. However, disseminating live videos optimally in a hybrid wireless network is challenging because: (1) wireless networks are dynamic in terms of latency and capacity, (2) video data requires high throughput and low latency, and (3) devices are highly mobile while WiFi ad hoc range is typically short (less than 100 meters).

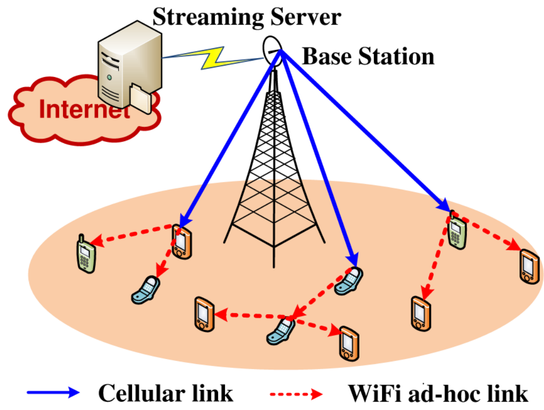

Figure 1: System architecture: Live Video Streaming. Video data is streamed from a server to mobile devices through cellular links and multi-hop ad hoc paths.

Fig. 1 depicts our system's architecture. In our approach, the streaming server transmits video data to some mobile devices over the cellular network. These mobile devices are then scheduled to relay the receiving video data to other mobile devices over WiFi ad hoc trees rooted at these devices. Our goal is to disseminate live videos to mobile devices such that video quality (measured by a well known metric PSNR) perceived at mobile devices are maximized.

To cope with challenges 1 and 2, we employ a scalable video coding standard H.264/SVC [26] to encode videos. A video is divided into D second *segments* (a typical value of D is 2 seconds [24]). Assume each segment can be independently decoded. A segment is encoded by H.264/SVC into L *layers*. In H.264/SVC [26], the first layer is referred to as the base layer, which provides a basic video quality. The higher layer are enhancement layers, which provide incremental quality improvements. Note that an enhancement layer is decodable if all layers below it are received. Thus, with H.264/SVC encoded videos, we can dynamically adjust the number of segment layers sent to each mobile devices based on the current bandwidth.

We develop a *scheduler* located at the streaming server, which computes *scheduling plans* to determine (i) which segment layers can be distributed with the current network bandwidth, and (ii) how to route segment layers to mobile devices. The scheduler runs every $D * W'$ seconds (i.e., an interval of W' segments) to come up with a schedule plan by selecting segment layers in a scheduling window of W consecutive segments for transmission. W and W' are system parameters, where $W \geq W'$. By changing W and W', the system can adapt well with the current network conditions, e.g., bandwidth and mobility (challenges 1 and 3).

For optimal video streaming, we design three scheduling algorithms for the scheduler to solve (i) and (ii): *POPT* (an optimization algorithm based on Mixed Integer Linear Programing), *MTS* (a heuristic algorithm based on Linear Programing), and *THS* (a heuristic algorithm requiring no optimization solver). Through testbed and simulation based evaluations, we show that our proposed system provides a

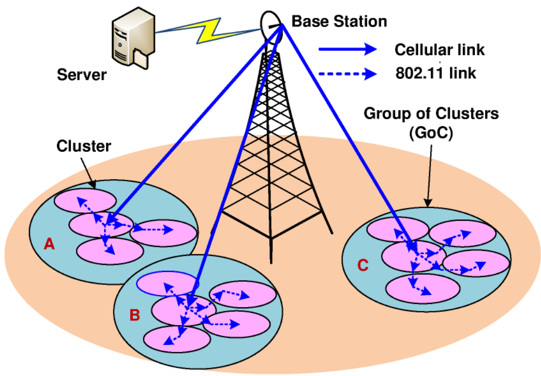

Figure 2: System architecture: Large content dissemination. The hybrid network is structured into clusters. Clusters are further grouped into groups of clusters.

much higher performance than existing systems using only cellular networks for live video streaming.

2.2 Non-delay Sensitive Data Sharing

Consider an emergency scenario in which a rescue team leader shares an image map of the rescue scene (e.g., a building) to mobile devices of his/her members. *Timely* dissemination is of course critical in such a scenario. Moreover, *high reliability* is required because members cannot recover the map if there is any single packet loss. A straightforward approach to delivering the map to the members is using a cellular network. The leader uploads the map to the server which later transmits the map to every member through cellular link. While the cellular network provides a high reliability because it covers a large area, this approach does not disseminate as fast as expected because the server has to deliver members one by one and the cellular network has low capacity. In our middleware based approach [16], we employ a WiFi ad hoc network interface to augment the cellular network for faster content distribution.

Design challenges of the above described system lie in (1) distributed content is large, and (2) WiFi ad hoc network is dynamic and suffers from interference. More specifically, to deliver rich contents with sizes from hundreds of KB to hundreds of MB, we need to divide each content into multiple chunks whose size is smaller than the maximum allowable size predefined by the underlying communication protocol. A receiver can reconstruct the original content only if all chunks of that content are successfully received. Packet loss in wireless networks is common due to mobility, interference, collision, and fading. Therefore, dividing a content into more chunks (happen if content is large) leads to a higher probability of content delivery failure. The WiFi ad hoc network is utilized to reduce bandwidth consumption on the cellular network. However, managing connection between mobile devices in the ad hoc network is difficult due to constant mobility and short communication ranges of the ad hoc links. Furthermore, since mobile devices share the same ad hoc spectrum, they are vulnerable to high interference in heavily loaded environments.

We employ a dissemination server to structure the hybrid network as well as store and disseminate rich contents to mobile devices. Before a content is distributed, it will be divided into multiple chunks whose size is less than or equal to the maximum size allowed by the underlying network layer for transmission without being fragmented. For example, T-Mobile 3G network allows to send chunks with size a bit larger than 1 KB. Fig. 2 shows the system architecture.

To deal with two above challenges, we structure the network into *clusters*. In each cluster, there is a central mobile device named as *cluster head* directly connecting to all other mobile devices in this cluster over ad hoc links. Further, clusters are integrated into *groups of clusters*. In a group of clusters, there is a central cluster connecting to all other clusters in the group. Cluster head of a central cluster is defined as *group of cluster head*. Cluster heads and group of cluster heads should be chosen based on bandwidth of their cellular links and their remaining energy because these mobile devices not only receive data from the server over the cellular network but also relay the receiving data to other mobile devices in the groups of clusters over the WiFi ad hoc network. The server participates only in forming groups of clusters. Clustering the network is performed in a distributed way by mobile devices. This helps reduce the server's workload.

We employ a *push-pull mechanism* in our system for chunk dissemination. Intuitively, the push phrase is to disseminate chunks fast while the pull phrase is to provide high reliability. In the push phrase, the server distributes chunks, one by one, to group of cluster heads over the cellular network. These devices then push the receiving chunks to cluster heads in their group over the WiFi ad hoc network. Because distance from a group of cluster head to any cluster heads in its group may be three hops, the group of cluster head will have to select some mobile devices as *relayers* to forward chunks to the cluster heads. Selecting a set of relayers such that the number of transmissions is minimized (this metric is important because this reduces wireless interference, energy consumption, and save bandwidth) is shown in our work as a *NP-Hard* problem. We solve this problem by proposing a greedy algorithm, and prove that this algorithm achieves a close approximation factor (1.2273 times) to the optimal solution. Once the cluster heads receiving the chunks, they further forward to their members in the clusters. By this way, each chunk will be pushed to every device in the network.

Chunks missing during the push phrase cannot be avoided due to packet loss in wireless networks. Therefore, the system employs a pull phrase to recover missing chunks. The cluster heads play key role in this phrase - they are responsible for collecting missing chunks for itself and their members. The cluster heads collect missing chunks for itself first from neighboring mobile devices over the WiFi ad hoc network. They only ask retransmission from the server for missing chunks that cannot be recovered from the neighboring devices because sending from the server will go through the cellular network. The cluster heads at the same time also transmit chunks missing at their members. This process terminates when all members completely receive the content.

A clear improvement for the above scheme is employing network coding to reduce the number of transmissions during the pull phrase. Another interesting problem is how to provide fairness to mobile devices. It is possible that among mobile devices participating in distributing content, there are a few ones that perform many more transmissions than the others. Using network coding for improving performance

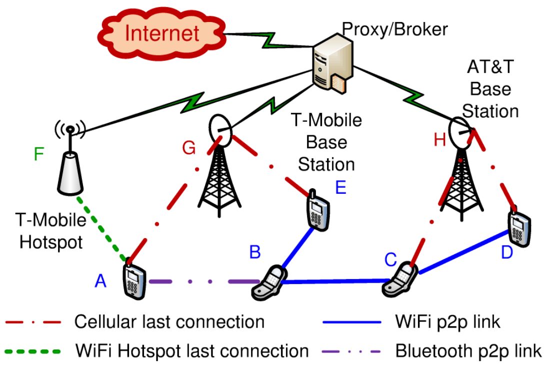

Figure 3: System architecture: Non-collaborative sharing. Mobile device B can request helps from mobile AP A or mobile AP E. B will pay a small monetary fee for mobile AP to use its service. Mobile AP A provides two services: cellular data connection (T-Mobile base station) and WiFi Hotspot data connection (T-Mobile WiFi Hotspot). Different services charge different prices.

and providing fairness are our future work.

3. NON-COLLABORATIVE SHARING

The above section describes our effort to designing approaches for sharing rich content that is interested by all mobile participants. The participants thus have a motivation for their collaboration to achieve high sharing performance as well as offload cellular data traffic. In this part, we study approaches for a class of rich content sharing applications in which not all of mobile devices are interested in receiving the content. If a mobile device is not interested in a content, intuitively it will not participating in delivering the content to other mobile devices because their data plan may not be unlimited and free, and their energy may drain fast.

In this work, our goal is to develop a *crowdsourcing oriented middleware* that can motivate mobile devices with data connection to the Internet to help the ones without such a connection for sharing rich content from/to a server located in the Internet. We refer mobile device with data connection as *mobile Access Point* (mobile AP) while still naming device without data connection as mobile device. We assume a mobile AP owns a non-free data plan offered by a network service provider. Data connection could be a connection to a base station or a WiFi Hotspot. An example scenario where the middleware can be deployed is that there is a mobile user taking a train from Los Angeles to San Francisco for travel. He forgets to bring a map of the destination city, and his phone does not have a cellular data plan to access the Internet. He thus look for a helper who can download the map and send it to him. To motivate the helper, the requester will pay a small monetary fee that covers two subcosts: (i) fee charged by the helper's network services provider and (ii) fee charged by the helper for local resources' consumption (e.g., energy). One more motivation for the helper to perform the request is that the helper has some unused quota in his data plan; The helper thus can get back part of monthly service fee for what he does not use.

In this middleware system, when a mobile device wishes to share a rich content to/from a server, it seeks for surrounding mobile APs over short range networks, e.g., WiFi ad hoc or Bluetooth, and asks for helps. Matching mobile devices with nearby mobile APs is not an easy task due to the following challenges:

1. How does a mobile AP make an admit/reject decision upon receiving a request from a mobile device? Admitting a larger number of requests brings higher revenues, but causes buffer overflow at the mobile APs and a longer end-to-end delay for file transfer. Higher delays may turn users away from the system.

2. How does a mobile device select a mobile AP and a corresponding service (Cellular or WiFi network)? Services charge different fees and provide different quality-of-services.

3. How do the mobile devices and mobile APs deal with uncertain channel conditions caused by mobility? The system has to solve the situation where a mobile device is moving out of its range before its file is completely transferred to the mobile AP. Similarly, a mobile AP's direct Internet access link may be disconnected due to mobility.

4. How does the system handle the associated security issues and legal implications of sharing mobile access? How does this scheme fit into the ISP/network provider ecosystem?

We employ Lyapunov optimization framework to design an admission control algorithm to address challenges 1 and 2. The admission control algorithm provides a facility for mobile AP to control the trade-off between the achieved revenue and the offering service quality. The service quality considered in our work is end-to-end delay mobile AP takes to deliver all contents admitted for transfer from mobile device to the server. When a mobile device searches for a mobile AP, it considers price requested and service quality offered by mobile APs before making decision.

To deal with challenge 3, we design a family of techniques to handle mobility. For example, we employ HELLO messages to keep track of the status of ad hoc links. Once a link breakage from a mobile device to a mobile AP is detected, the mobile device finds another mobile AP to continue sharing. HELLO messages should not be broadcast frequently due to interference and bandwidth wasting. We further employ chunk based sharing technique and link duration prediction technique to improve system performance.

The concern of the need for a tighter integration of the proposed scheme into the ISP/provider ecosystem is perhaps not as complex as one might imagine. Today, ISPs such as T-Mobile [10] or AT&T [8] offer plans that combine cellular access with a mobile Hotspot feature; AT&T users can enable a tethering feature on their device to allow peer-to-peer mobile users to access the Internet if they own 5 GB data plan. Note also that our middleware can be implemented in-network by service providers, where the providers facilitate a marketplace in which users within partner networks can exchange unused and residual data plan minutes at low costs – this can encourage users to continue their data plan subscription creating a win-win situation for both users and providers.

Our future work focuses on addressing security issues. The middleware system clearly requires security mechanisms to protect a users' file as it passes through arbitrary (and potentially untrusted) mobile APs, networks and middleboxes; this includes protection from DOS attacks and from malicious services. How to leverage cryptographic techniques to provide such end-to-end security is our current aim. Moreover, we would develop approaches to support dynamic pricing in the system in which mobile APs can offer prices which depend on not only transferred data amount but also resources available at mobile AP itself and at surrounding mobile APs.

4. REFERENCES

[1] http://www.youtube.com.

[2] http://www.pandora.com.

[3] http://www.facebook.com.

[4] http://www.mobiledia.com/news/96884.html.

[5] AT&T faces 5,000 percent surge in traffic. http://www.internetnews.com/mobility/article.php/3843001, 2009.

[6] T-Mobile's growth focusing on 3G. http://connectedplanetonline.com/wireless/news/t-mobile-3g-growth-0130, 2009.

[7] Wi-Fi certified Wi-Fi Direct: Personal, portable Wi-Fi that goes with you anywhere, any time. http://www.wi-fi.org/Wi-Fi_Direct.php, 2010.

[8] AT&T data plan. http://www.att.com/shop/wireless/plans/data-plans.jsp?fbid=w6awFbTp_qQ, 2012.

[9] Cisco visual networking index: Global mobile data traffic forecast update, 2011 to 2016. http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html, 2012.

[10] T-Mobile data plan. http://www.t-mobile.com/shop/plans/mobile-broadband-plans.aspx, 2012.

[11] T-Mobile's wifi calling. http://t-mobile-coverage.t-mobile.com/4g-wireless-broadband-service, 2012.

[12] R. Bhatia, L. Li, H. Luo, and R. Ramjee. ICAM: Integrated cellular and ad hoc multicast. *IEEE Transactions on Mobile Computing*, 5(8):1004–1015, August 2006.

[13] N. Do, C. Hsu, X. Huang, and N. Venkatasubramanian. An experimental study on scalable video streaming over hybrid cellular and ad hoc networks. In *Proc. of ACM/IFIP/USENIX International Conference on Middleware (Middleware'11), Poster Session*, Lisbon, Portugal, December 2011.

[14] N. Do, C. Hsu, S. Jatinder, and N. Venkatasubramanian. Massive live video distribution over hybrid cellular and ad hoc networks. In *Proc. of IEEE WoWMoM*, pages 1–9, Lucia, Italy, 2011.

[15] N. Do, C. Hsu, and N. Venkatasubramanian. Crowdmac: A crowdsourcing system for mobile access. In *Proc. of ACM/IFIP/USENIX International Conference on Middleware (Middleware'11)*, Montreal, Canada, December 2012.

[16] N. Do, C. Hsu, and N. Venkatasubramanian. Hybcast: Efficient rich content dissemination over hybrid cellular and ad hoc networks. In *Proc. of IEEE 31st International Symposium on Reliable Distributed Systems (SRDS'12)*, Irvine, CA, 2012.

[17] B. Han, P. Hui, V. Kumar, M. Marathe, J. Shao, and A. Srinivasan. Mobile data offloading through opportunistic communications and social participation. *IEEE/ACM Transactions on Mobile Computing*, 11(5):821–834, 2012.

[18] F. Hartung, U. Horn, J. Huschke, M. Kampmann, T. Lohmar, and M. Lundevall. Delivery of broadcast services in 3G networks. *IEEE Transactions on Broadcasting*, 53(1):188–199, March 2007.

[19] H. Hsieh and R. Sivalumar. On using peer-to-peer communication in cellular wireless data networks. *IEEE Transactions on Mobile Computing*, 3(1):57–72, March 2004.

[20] S. Hua, Y. Guo, Y. Liu, H. Liu, and S. Panwar. Scalable video multicast in hybrid 3G/ad-hoc networks. *IEEE Transactions on Multimedia*, 13(2):402–413, 2011.

[21] L. Lao and J. Cui. Reducing multicast traffic load for cellular networks using ad hoc networks. *IEEE Transactions on Vehicular Technology*, 55(3):317–329, May 2006.

[22] L. Law, K. Pelechrinis, S. Krishnamurthy, and M. Faloutsos. Downlink capacity of hybrid cellular ad hoc networks. *IEEE Transactions on Networking*, 18(1):243–256, February 2010.

[23] H. Luo, R. Ramjee, P. Sinha, L. Li, and S. Lu. UCAN: a unified cellular and ad-hoc network architecture. In *Proc. of ACM International Conference on Mobile Computing and Networking (MobiCom'03)*, pages 353–367, San Diego, CA, September 2003.

[24] P. Ni, A. Eichhorn, C. Griwodz, and P. Halvorsen. Fine-grained scalable streaming from coarse-grained videos. In *Proc. of International workshop on Network and Operating Systems support for Digital Audio and Video*, pages 103–108, Williamsburg, VA, September 2009.

[25] J. Park and S. Kasera. Enhancing cellular multicast performance using ad hoc networks. In *Proc. of IEEE Wireless Communications and Networking Conference (WCNC'05)*, pages 2175–2181, New Orleans, LA, March 2005.

[26] H. Schwarz, D. Marpe, and T. Wiegand. Overview of the scalable video coding extension of the H.264/AVC standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(9):1103–1120, September 2007.