Topology-aware integration of cellular users into the P2P system

Mohammad Zulhasnine
Systems and Computer Engineering
Carleton University, Ottawa, Canada
Email: mzulhasn@sce.carleton.ca

Changcheng Huang
Systems and Computer Engineering
Carleton University, Ottawa, Canada
Email: huang@sce.carleton.ca

Anand Srinivasan
EION Inc., Ottawa, Canada
Email: anand@eion.com

***Abstract*—Integration of cellular users into the peer-to-peer (P2P) network is still in limbo due to limitations caused by heterogeneity, mobility and time-varying capacities of the wireless channel. If traditional Chord is employed to include users from the cellular networks, users under the same base station scatter in logical topology randomly. In this paper, we present a novel cellular Chord (C-Chord) P2P system that integrates the cellular users into the well-established structured P2P network in topology-aware fashion. C-Chord offers the cellular users a choice of downloading contents either from the Internet peers at faster rate or from other cellular users from the same base station avoiding Internet data penalty. We conduct extensive simulations to evaluate the performance of the proposed C-Chord P2P system. The path-length per lookup query is smaller than that of the traditional Chord system. Overhead due to renewal of routing information is also smaller for the cellular users in the C-Chord system. We also measure the throughput at the cellular receivers to analyze the effects of selecting peers either from same base station or from outside the Internet gateway.**

I. INTRODUCTION

As more appealing applications and services are offered, we are being less wired on the contrary. Fourth-generation (4G) mobile technology aims at meeting the growing demands of supporting data applications at faster speed. Cisco estimates 131 percent compound annual growth rate of the mobile traffic between 2008 and 2013 [1]. Cellular users can share ring tones, themes, wallpapers, games, or music by joining into the P2P paradigm rather than downloading from commercial content providers. Integration of cellular users into the P2P network is still in limbo due to limitations caused by heterogeneity, mobility and time-varying capacities of the wireless channel. P2P system relies on overlay network, and yields some penalties in terms of bandwidth usage due to a partial removal of routing intelligence of the network layers. Several researchers emphasized on the scalable topology control protocol to discover neighboring wireless peers to save valuable wireless bandwidth [2], [3], [4]. [2] proposed a topology-aware P2P protocol for wireless network where physically closed nodes get adjacent identifiers (*ids*) exploiting nodes' position information. The authors in [3] proposed a wireless P2P system that constructs an adjacency set based on application-level properties. Both methods in [2], [3] are only suitable for infrastructure-less wireless network and do not consider Internet gateway in the design. [4] proposed a location-aware Chord [5] based P2P algorithm for wireless mesh network to match the overlay network and physical network. However, this method requires location-awareness possibly using the GPS receivers. The users also may not feel comfortable to release geographical location due to privacy and security reasons. Few authors also studied how to integrate cellular network into the P2P paradigm [6], [7]. [6] proposed a network-aware P2P file architecture for wireless mobile networks. This scheme assigns the peers to a network-aware cluster using a network prefix division and thereby enables the files to be searched first with nearby peers. The bootstrap peer maintains up-to-date cluster routing table to direct new peers joining the network to the appropriate clusters. The authors in [7] claimed that the P2P network model for Internet access in cellular wireless data networks does not translate into better throughput performance. Instead as-is might actually degrade the throughput performance of the network due to a bottleneck of the channel around the base station, protocol inefficiencies and host mobility. The authors suggested base station assistance and multi-homed peer relays approaches to achieve the performance enhancement while providing fair service and resilience to mobility. However, support from management and network operator is most unlikely in the context of P2P network. In this paper, we propose an efficient P2P system that integrates cellular users in topology-aware style. The proposed C-Chord P2P system enforces a certain desirable structure to the overlay network. Our proposed system needs base station's unique numeric cell identifier (Cell-ID), already available to all cellular users, and does not require base station's assistance, unlike [7]. Different from [6], our model does not require any super peer that maintains indexes of shared file and peers' location. The key contributions to the work are:

- Our proposed C-Chord P2P system integrates the cellular users into the popular Chord P2P system in topology-aware fashion. Cellular users have the option to choose between Internet peers and cellular peers. The users of the cellular network can either download diverse content from the stable Internet peers at faster rate or download social contents from the peers within the same base station avoiding Internet data penalty.
- Our proposed system reduces the high management cost of routing information with fewer entries in the routing table for the cellular peers.

978-1-4244-8327-3/11/$26.00 ©2011 IEEE

- We measure the mean hop-count per lookup request and compare it with that of the traditional Chord system.
- We also measure the throughput experienced by the cellular receivers to analyze the effects of peer selection strategy.

II. SYSTEM MODEL AND THE PROBLEM STATEMENT

In the steady state, there are N wired peers that form the main chord with K keys. There are additional U cellular peers from V number of cell sites that form V auxiliary Chord rings. u_i denotes any cellular peer which belongs to $i-$th base station. U_i denotes the number of cellular peers of $i-$th auxiliary Chord or under the $i-$th base station, that is in the steady state $\sum_{i=1}^{V} U_i = U$. Peers, in the Chord system, construct an overlay network with location-independent virtual addresses. If traditional Chord is employed to include users from the cellular networks, users under the same base station scatter in logical topology randomly. Cellular users thus do not have the option of choosing between local peers and the Internet peers. By choosing local peers (i.e. peers from the same base station) cellular users may not only avoid Internet data penalty but also may look for direct connectivity employing device-to-device (D2D) communication. On the other hand, cellular users with better data plan from the commercial provider, may also like to connect with Internet peers to download diverse content at a faster rate. We therefore need to design an efficient P2P system that integrates cellular users in topology-aware fashion and offers a selection choice between local peers and Internet peers. The design should also reduce (compulsory copies and update) cost for routing information with peers joining/leaving the network all the time.

III. THE PROPOSED C-CHORD P2P SYSTEM

We use the simplest and prevalent protocol Chord as the base protocol. However, our system will work for other DHT system. First, we briefly describe the base Chord mechanism and then describes the details of our proposed C-Chord P2P system.

A. Background of the base Chord P2P system

Chord provides a mechanism to store key/data pair onto nodes responsible for them in a distributive manner employing consisting hashing function and efficiently locates the node that stores a specific data item associated with a key. A key k is associated to each resource available in the network. An m bit number or *id* is generated for each key k by hashing the resource name, and for each node n by hashing the IP address. Chord maps both peer and key *id*s on the same circle of numbers from 0 to $2^m - 1$. We use the notation id_n and id_k just to indicate that the *id*s belong to a node n and a key k, respectively. Each key is stored to the first node whose *id* is equal to or follows (the *id* of) k in the circle. This node is defined as the *successor node* of key k and denoted as $successor(k)$. Each node, n, maintains a *finger table* with (at most) m entries to accelerate the routing process. Readers are encouraged to read [5] for detail descriptions. Although routing in Chord becomes more efficient due to the introduction of the *finger table*; maintenance overhead of the distributed hash table (DHT) routing structure grows with the number of entries in the *finger table* and the peers' churn rate.

B. The proposed C-Chord P2P system

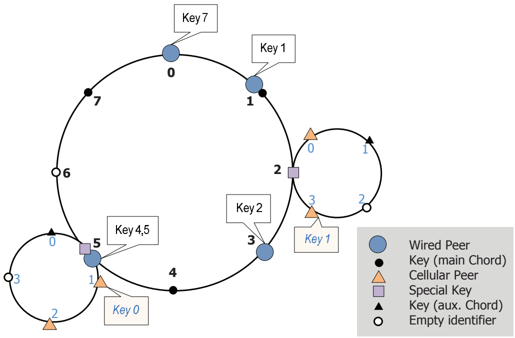

Fig. 1. C-Chord P2P system

The proposed C-Chord P2P system incorporates the cellular network into the traditional large-scale Chord model where the cellular users under the same base station may choose to communicate locally or harvest diverse contents from the Internet peers as well. C-Chord model consists of one large main Chord (*m*-Chord) ring that contains Internet peers, and several (equal to the number of base stations) auxiliary Chord (*a*-Chord) rings containing all the cellular peers as depicted in Fig. 1. *a*-Chord rings are attached to the *m*-Chord ring exploiting a *special key*. We describe this special key in the next paragraph. Cellular users with the same base station are part of the same *a*-Chord ring and therefore, discover the neighboring wireless peer through application driven mechanism which is more realistic in the context of P2P file sharing.

We design the *special key* and the data associating it such that each *a*-Chord is tightly coupled with the *m*-Chord. Every base station has a unique numeric Cell-ID and cellular users retrieve this Cell-ID through cell search procedure. The unique Cell-ID serves as the common key to the users under that particular base station. This is the beauty of our proposed mechanism. An m bit *id* is generated by hashing each Cell-ID and denoted as id_{cell}. The data associates with this special key with identifier id_{cell} has the format $\mathcal{IP}^{key} = \{ip_1, ip_2, \ldots, ip_J\}$ which is a list of J number of current cellular users' IP addresses forming the *a*-Chord under the same station. Any cellular user entering the C-Chord system generates the *special key* by hashing its own base station Cell-ID and joins the corresponding *a*-Chord ring from the list $\mathcal{IP}^{key}$. $\mathcal{IP}^{gw} = \{ip_1, ip_2, \ldots, ip_{J'}\}$ denotes the list of IP addresses of the *successor* of the *special key* and adjacent *predecessors*. The cellular peer in the *a*-Chord ring maintains $\mathcal{IP}^{gw}$ that serves as the gate way towards the *m*-Chord. Cellular users' resources may be concealed in the *a*-Chord ring from the *m*-Chord nodes. To make cellular users a valuable trading partners with the peers in the *m*-Chord; some of those popular resources are hashed and published in the *m*-Chord through the peers in the list $\mathcal{IP}^{gw}$. Both *m*-Chord ring and *a*-Chord rings maintain each node's *successor* and hold node $successor(k)$ accountable for key k in the dynamic network with peers joining and leaving all the time. For efficient lookup, the *m*-Chord ring and *a*-Chord rings also maintain *finger tables*. Number of peers in each *a*-Chord ring is far fewer than the number of Internet peers in the *m*-Chord. Each cellular peer, therefore, maintains a *finger table* with fewer entries.

```
/* $u_j$ from the $j$-th base station joins the network            */
/* $n'$ is an arbitrary node in the main Chord                      */
/* $id^j_{cell}$ is the key $id$ of $j$-th base station's Cell-ID   */
$u_j$.cu_join($n'$)
if ($n'$) then
    $n = n'.search\_special\_key(id^j_{cell})$;
    if $n \neq nil$ then                    /* $u_j$ joins in $j$-th aChord */
        /* Copies the list of $ip$ addresses                        */
        $\mathcal{IP}^{key}_j = get(id^j_{cell})$;
        /* Joins the $j$-th aChord                                  */
        $u_j.join(ip)$    /* $ip \in \mathcal{IP}^{key}$ and currently active */
        Set $\mathcal{IP}^{gw}_j$;
        Update $\mathcal{IP}^{key}_j$ by replacing last entry with $u_j$'s ip address;
    else                          /* $u_j$ is the only node of $j$-th aChord */
        for $i = 1$ to $m'$ do
            $finger[i].node = u_j$;
            $predecessor = u_j$;
        end
    end
else                                  /* $u_j$ is the only node in mChord */
    for $i = 1$ to $m$ do
        $finger[i].node = u_j$;
        $predecessor = u_j$;
    end
end
/* ask node $n$ to search for the key's $id_k$                      */
$n$.search_special_key ($id_k$)
$n' = find\_successor(id_k)$;
if $n'$ contains the key then  return $n'$;
else return nil;
```

Fig. 2. Pseudocode for the cellular user joining process.

1) Node's joining process: Peers may join (or leave) the system at any time in the dynamic network. Here we discuss how a cellular peer u_j from the j-th base station joins the P2P network. We denote id^j_{cell} as the *m*-Chord *id* of the j-th base station Cell-ID. We assume that when cellular node u_j wants to join the system, it is informed of any Internet peer n in the *m*-Chord by some external mechanism. Then it contacts the Internet node n and searches the special key corresponding to its own Cell-ID. The Internet node n does not require any special properties to guide the cellular peer. C-Chord performs the following task when a cellular peer u_j joins the network.

- Find the *successor* of the special key id^j_{cell} on the *m*-Chord.
- Update and copy the data associated with the key id^j_{cell}. The data is simply the list of IP addresses of the cellular users participating in the *a*-Chord from the j-the base station.
- Join the *a*-Chord by contacting any of the cellular users from the list of IP addresses.

Fig. 2 depicts the pseudocode for the cellular user joining process. For example, consider a cellular user with *id*$=0$ is informed of the *m*-Chord node '1'. Cellular user's base station Cell-ID is '2' (*special key id*) and is stored in the *m*-Chord node '3'. See Fig. 1. Cellular user finds the successor (*m*-Chord node '3') of the *special key* '2'. Cellular user '0' then gets the data corresponding to *special key* '2' which is only the list of IP addresses of some cellular nodes already joined the *a*-Chord ring (the smaller ring on the right side). Cellular user '0' updates its own IP address by modifying the data corresponding to the *special key* '2' and also initiates its *finger table.*

2) Lookup process of a cellular node: If the cellular node in the C-Chord system prefers to download from the Internet peer, it sends query to successor of the *id* corresponding to its own base station's Cell-ID or to any of the node from the list $\mathcal{IP}^{gw}$. Anyone might prompt to raise the scalability issue here and argue that search process might overwhelm the successor lists with queries. Bear in mind, as the cellular users increase so is the number of base stations and the number of *a*-Chord rings. Also, only some percentage cellular users from each base station joins the P2P from each base stations. As the users from the cellular networks are distributed and sorted by their Cell-ID. This innate property of the cellular network provides scalability to the C-Chord system. If the cellular user rather wants to avoid Internet data penalty, it first tries to resolve a query within the *a*-Chord system and then forwards the query to the *m*-Chord.

3) Resilience to peer failures: In the C-Chord P2P system, the large *m*-Chord ring and all *a*-Chord rings inherit the resilience to peer failure property from the base Chord system. Even though the *special key* may suffered from node failure; the existing detach *a*-Chord can rebuild the key due to uniqueness of Cell-ID. In case new peers form another *a*-Chord by hashing the same base station Cell-ID, old *a*-Chord can reclaim and reunite with the new *a*-Chord by the virtue of the uniqueness of the *special key*. This inborn property provides robustness to the C-Chord P2P system. Also, *special key* failure only affects the corresponding small *a*-Chord ring, not the large *m*-Chord ring and other *a*-Chord rings in the C-Chord P2P system.

IV. PERFORMANCE MEASUREMENT

We first evaluate the performance of our proposed C-Chord P2P system in terms of path-length (number of hop-count) per lookup request. Then we numerically analyze the C-Chord based P2P system in terms of throughput experienced by the cellular receivers and investigate the effects of peer selection strategy.

A. Hop-count per lookup request

We implement the C-Chord protocol iteratively where each searching for a specific key sends queries to a series of nodes and each time advances closer to the *successor*. Mean hop-count between two arbitrary nodes in resolving a given query influences the performance of the routing protocol and the scalability of the P2P system as the number of participating peers could be large. The base Chord model in [5] has mean hop-count $\frac{1}{2}\log_2 N$ in practice, where N is the network size. When a cellular user requests a query from the *a*-Chord ring of size N_a, the mean hop-count to resolve the query is

$$\frac{1}{2}p\log_2 N_a + \frac{1}{2}(1-p)\log_2 N_m \tag{1}$$

Where p is the probability that the successor exists in the *a*-Chord and N_m is the size of the *m*-Chord.

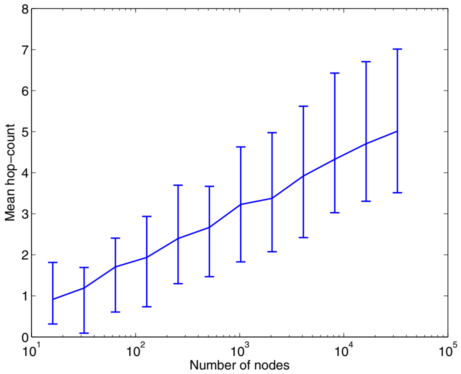

Fig. 3. Mean hop-count as a function of network size for $p = 0.5$.

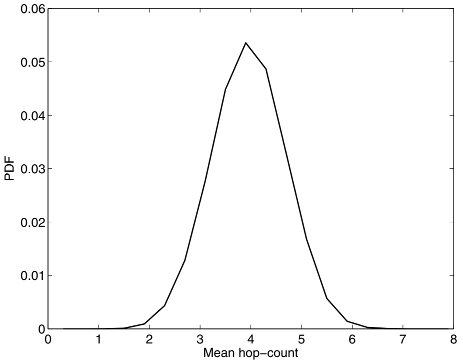

Fig. 4. The PDF of the mean hop-count with 2^{12} nodes. 50% of the nodes are cellular peers. The number of base stations or *a*-Chord rings is 32, each *a*-Chord consists of 64 peers.

In our simulation, the number of nodes in the overlay network is $N = 2^k$ that store 100×2^k keys. We arbitrary consider 50% of N nodes are cellular peers to highlight the effects of nodes from the cellular network. We varied k from 4 to 15, and also increase the number of base stations as the number nodes from the cellular network increases. Each node from the *a*-Chord randomly requests a query for a set of keys and we calculate the hop-count to resolve each query. First we illustrate the lookup efficiency of our C-Chord system and then we compare C-Chord with the base Chord model. Fig. 3 shows the mean hop-count as a function of k for $p = 0.5$. The error bar shows the 1st and 99th percentiles. As expected the mean hop-count increases with network size according to Eq. (1) approximately. For example with $N_m = 2^{11}$, $N_a = 32$, and $p = 0.5$, the mean hop-count is 4.0 approximately. Fig. 4 shows the probability density function of the hop-count for this network size.

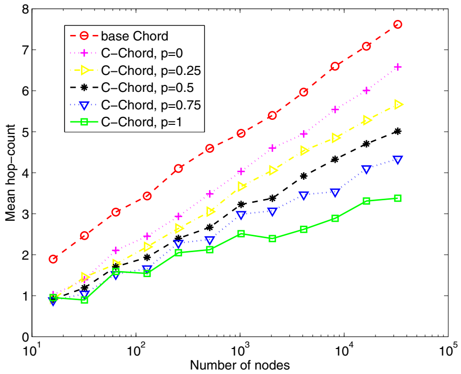

Fig. 5. Comparison of mean hop-count between base Chord and C-Chord with different values of p.

We also compare the mean hop-count per lookup request between the base Chord and C-Chord P2P system. Since the cellular node from the *a*-Chord ring first sends query to its existing *a*-Chord ring, and then to the larger *m*-Chord ring, the mean hop-count decreases with the probability the *successor*'s presence in the *a*-Chord ring increases. Our proposed C-Chord system shows improve performances with smaller hop-count per lookup query over the base Chord model. Hence, the C-Chord P2P system integrates cellular network efficiency.

B. Network traffic flows in the P2P system

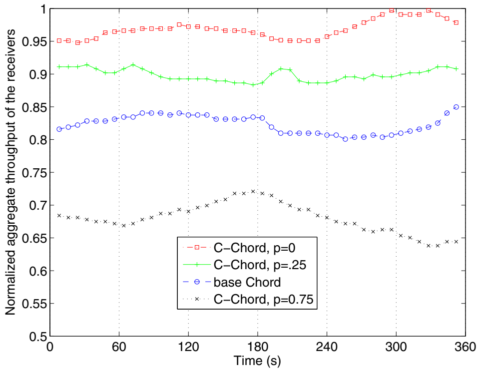

Fig. 6. The normalized aggregate throughput of the receivers for scenario 1.

We also analyze C-Chord system to investigate the effects of selecting peers either from same base station or from outside the Internet gateway. We implement the C-Chord system on several different networks that includes peers from the Internet and also from the cellular networks. For each network, we select few nodes randomly that download files of approximately 100 MB from different senders. The number of bits in the *id* space is 160 and 40 for the *m*-Chord and the *a*-Chord, respectively. We perform each experiment 50 times tracing throughput of the receivers on each second. The aggregate throughput are averaged over all runs on each second. The physical links between Internet peers are generated by using a stochastic loss model [8]. The available bandwidth is set randomly in the range $[0.75R_0, 1.25R_0]$, where R_0 is the base

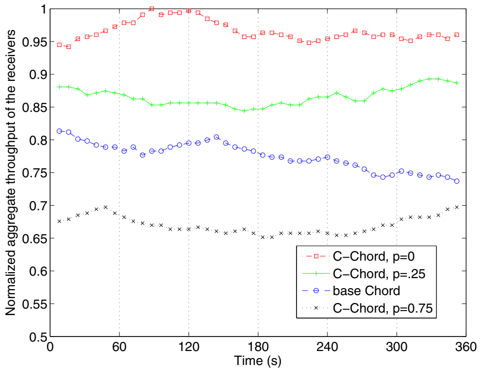

Fig. 7. The normalized aggregate throughput of the receivers for scenario 2.

bandwidth. We also set the maximum bandwidth of the cellular peers to $0.1R_0$, 10 times smaller than the wired network. We set $R_0 = 100\ kbps$, which is realistic as senders are tapped while uploading. To capture the time varying capacity of the wireless channel, we follow the simulation settings as in [9]. As the simulation patterns remain similar, we present our results for two network scenarios. In both scenario, the number of Internet peers and cellular peers are equal. In scenario 1, the number of Internet peers is 256 and there are 8 base stations each having 16 cellular users. While in scenario 2, the number of Internet peers is 512 and there are 16 base stations each having 32 cellular users. We pick randomly 5 pairs (source and downloaders) from scenario 1 and 10 pairs from scenario 2. Fig. 6 and 7 show the normalized aggregate throughput (per second) of the receivers for scenario 1 and 2 respectively. If x sources are selected locally within the *a*-Chord ring from total y sources, the probability of finding the sources within the *a*-Chord ring, $p = x/y$. The receivers in all cases are chosen from cellular networks, as our primary interest is the performance of the cellular nodes. The run time of the simulation is 6 minute excluding initial setup time.

With higher values in p, (for example $p = 0.75$) the aggregate throughput is much lower as the receivers of the cellular networks download most of the contents locally. Some of the sources suffer from time varying wireless capacity and lower upload bandwidth. The base Chord P2P system also performs worse in terms of aggregate throughput than that of C-Chord P2P system with lower values of p. In the base Chord P2P system, the receivers cannot differentiate between cellular peers and Internet peers. The aggregate throughput is the highest when peer choose (or are forced) to download from the Internet peers only. That is when $p = 0$ in the C-Chord P2P system. All the receivers of the cellular networks download contents from the more stable Internet peers in *m*-Chord ring. However, in this case Internet data penalty is the highest.

V. CONCLUSION

In this paper, we have proposed C-Chord P2P system that integrates the cellular users into the popular Chord P2P system by enforcing a certain desirable structure to the overlay network. C-Chord has provided the cellular users a choice of downloading contents either from the Internet peers at faster rate or from other cellular users from the same base station avoiding Internet data penalty. In the simulation we have shown that mean hop-count per lookup request decreases with the increase of the probability of finding peers within the *a*-Chord ring. This is due to the fact the size (in terms of number of nodes) of the *a*-Chord ring is much smaller than that of the *m*-Chord ring. Moreover, the proposed C-Chord system has shown improved performance with lesser mean hop-count than that of the traditional Chord system for all probability of finding peers within the *a*-Chord ring. In the C-Chord P2P system, maintenance of routing structure is more efficient with lesser entries *a*-Chord's *finger table*. We also have numerically analyzed the C-Chord based P2P system in terms of throughput experienced by the cellular receivers and investigated the effects of peer selection strategy. The base Chord P2P system has shown worse performance in terms of aggregate throughput than that of C-Chord P2P system with lower probability of finding the peers within the same *a*-Chord. The aggregate throughput was the highest when cellular peers downloaded contents from the more stable Internet peers in *m*-Chord ring paying highest Internet data penalty. Although in our simulation we have scheduled few peers leaving the system and few others join the system at random time, we have not measured the degree of fault tolerance. Measuring the degree of fault tolerance even with concurrent join/failure is our ongoing task. Our C-Chord system identifies users from the same base station. Cellular users in close proximity may communicate directly, by employing D2D connectivity offloading base station load [10]. We are over enthusiastic to enable this D2D option within the *a*-Chord ring in future.

REFERENCES

[1] Cisco, "Visual networking index: Forecast and methodology, 2008-2013," *White Paper*, 2009.

[2] S. Wang and H. Ji, "Realization of topology awareness in peer-to-peer wireless network," in *WiCOM*, Sept. 2009, pp. 1–4.

[3] A. K. H. Leung and Y.-K. Kwok, "On localized application-driven topology control for energy-efficient wireless peer-to-peer file sharing," *IEEE Trans. Mobile Comput.*, vol. 7, no. 1, pp. 66–80, Jan. 2008.

[4] C. Canali, M. E. Renda, P. Santi, and S. Burresi, "Enabling efficient peer-to-peer resource sharing in wireless mesh networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 3, pp. 333–347, Mar. 2010.

[5] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," in *Proc. ACM SIGCOMM*, 2001, pp. 149–160.

[6] C.-M. Huang, T.-H. Hsu, and M.-F. Hsu, "Network-aware P2P file sharing over the wireless mobile networks," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 1, pp. 204–210, Jan. 2007.

[7] H.-Y. Hsieh and R. Sivakumar, "On using peer-to-peer communication in cellular wireless data networks," *IEEE Trans. Mobile Comput.*, vol. 3, no. 1, pp. 57–72, Jan-Feb 2004.

[8] M. Hefeeda, A. Habib, B. Botev, D. Xu, and B. Bhargava, "PROMISE: peer-to-peer media streaming using collectcast," in *Proc. ACM MM*, 2003, pp. 45–54.

[9] P. Mogensen, W. Na, I. Z. Kovacs, F. Frederiksen, A. Pokhariyal, K. I. Pedersen, T. Kolding, K. Hugl, and M. Kuusela, "LTE capacity compared to the shannon bound," in *IEEE VTC, Spring*, 2007, pp. 1234–1238.

[10] M. Zulhasnine, C. Huang, and A. Srinivasan, "Efficient resource allocation for device-to-device communication underlaying LTE network," in *IEEE WiMob*, 2010, pp. 368–375.