Energy-Efficient Peer-to-Peer Caching and Mobility Management in 4G Hybrid Networks

Mehdi Azarmi
Bharat Bhargava

CSD TR #08-030
December 2008

Energy-Efficient Peer-to-Peer Caching and Mobility Management in 4G Hybrid Networks

Mehdi Azarmi, Bharat Bhargava
{mazarmi, bbshail}@purdue.edu

Abstract

Hybrid wireless networks are an integration of infrastructure-based (e.g., Cellular or Wireless LANs) and mobile ad hoc networks. They are emerging as an attractive solution to providing seamless mobile data services. However, performance of hybrid networks is severely impacted due to high energy consumption and uneven load distribution among nodes. In this paper, we introduce a novel scheme called *energy-efficient peer-to-peer caching with optimal radius* (EPCOR), that reduce energy consumption and distributes load equitably among mobile peer nodes. In this scheme, a peer-to-peer (P2P) overlay network is built among the mobile nodes to facilitate cooperative data sharing in order to relieve the traffic bottleneck at the base station. In particular, for energy conservation, each mobile user (MU) in P2P overlay shares a data item in a *cooperation zone*. An analytical model is developed to evaluate the performance improvement due to energy conservation in the EPCOR scheme. An algorithm is developed to determine the optimal radius of the cooperation zone, based on the trade-off between performance improvement and the overhead of cooperation. Both analytical and NS-2 based simulation results show that EPCOR achieves significant performance improvement in energy saving, network throughput and load balance among mobile peers in hybrid wireless networks.

Keywords: Energy Efficient Mobile Peer-to-Peer Networks, Cooperative Caching, Data Dissemination.

I. INTRODUCTION

Recent advances in wireless communications have made mobile computing and wireless Internet a reality. To provide wireless Internet services, currently there exist two major categories of *infrastructure-based* networks: cellular networks and Wi-Fi networks (e.g., IEEE 802.11 based wireless LANs). Cellular networks operate on licensed frequencies and offer wide geographic coverage (up to 20 Km) but limited bandwidth (e.g., 107.2 Kb for GPRS). On the other hand, Wi-Fi networks operate on unlicensed frequencies and offer high bit rate (e.g., 11 Mbps for IEEE 802.11b

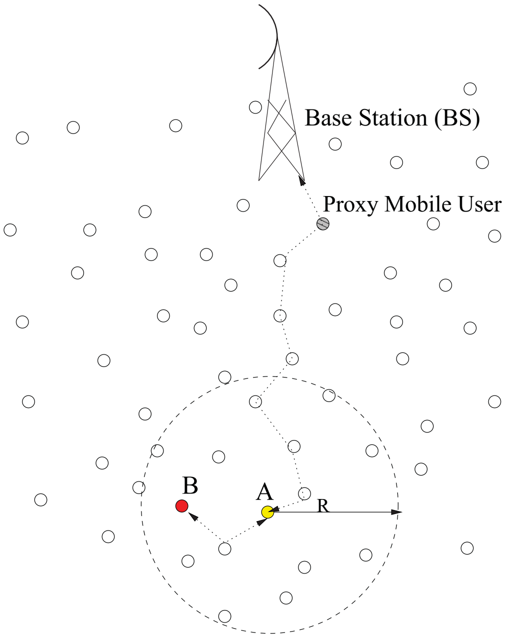

Fig. 1. An Example of Multi-Hop Hybrid Wireless Networks

and up to 54 Mbps for IEEE 802.11g) but very limited coverage (up to 250 m). In order to provide mobile data services efficiently, both high bit rate and wide coverage are desirable.

Ad hoc networks were originally designed for environments that lack established infrastructure. Communication networks for such scenarios as battle field and disaster relief operations are examples of ad hoc networks. In ad hoc networks, each mobile user (MU) retrieves the desired information from other MUs through multi-hop paths in which MUs cooperatively act as routers to relay the data traffic However, due to the absence of stable network connection, a pure ad hoc network has a low reliability to provide mobile data services. Recently, several models have been proposed for designing *hybrid networks* that are an integration of ad hoc and infrastructure-based networks with a goal to extend the coverage area of Wi-Fi networks [12] [18] [19] [39] and also improve the throughput of cellular networks [10] [21] [36]. In hybrid networks, the base stations (or access points) are assumed to be attached to the Internet or an infostation [13]. When an MU, say A, moves out of the transmission range of its BS or if the quality of cellular wireless link is low, other MUs along the path to the BS allow A to continue to access information by relaying its packets to the BS. The proxy MUs around the BS serve as a bridge to exchange data between the BS and other MUs. Figure 1 illustrates an example of hybrid networks. Since the hybrid networks significantly improve the data rate and extend the wireless service coverage area at low cost, this network model offers an attractive solution to providing mobile data services.

However, there are three major challenges for providing seamless mobile data services in hybrid wireless networks.

1) Due to the frequent data exchange with the BS, the battery power of the MUs that reside closer to the BS depletes faster than those of other MUs, thus resulting in a shorter lifetime of the whole network.
2) Since all MUs retrieve data from the BS, the channel congestion and disconnection with the BS (the bottleneck) degrades the network throughput.
3) The MUs that are far away from the BS suffer from high access latency if all requested data items are fetched through the BS.

In order to overcome these limitations, first we investigate the service environments of hybrid wireless networks. Because hybrid wireless networks are usually designed for the environments where the density of MUs is high enough to support peer-to-peer communication among MUs in the network. Examples of such environments include airport, a commercial center, a campus or an urban environment. In such environments, users usually access local and general information, such as news headlines, weather reports, sports, map, music, video files or mobile games. The information requested in such environments shows high spatial locality. According to the measurement conducted in a wireless campus network [17], more than 20% of all requests from users are for data objects that have been requested by a nearby user within the last one hour. Therefore, caching and cooperative sharing of frequently accessed data objects among MUs can significantly reduce the energy and bandwidth consumption of each data request. In the example shown in Figure 1, an MU, say A, requests for a data object. If A can find the data object in the cache of a nearby MU, say B, instead of fetching the data object from BS, this will save bandwidth and energy to retrieve the data object and balance the data traffic in the hybrid wireless networks, since the number of hops between A and B is supposedly less than that between A and BS. This observation motivates us to design an efficient mechanism to enhance peer-to-peer data communication among MUs to reduce energy and bandwidth consumption and improve load balance of hybrid wireless networks.

In this paper, we propose a novel scheme, called *energy-efficient peer-to-peer caching with optimal radius* (EPCOR) to support peer-to-peer data accesses in multi-hop hybrid networks. In the EPCOR scheme, each mobile user (or peer) has a cache to store the frequently accessed data items. Cached data items in each peer node satisfy local queries as well as those from peers. Additionally, a peer-to-peer (P2P) overlay network is built among the mobile users to help them cooperatively share cached data items. The neighborhood relationship between the peers in the overlay network is maintained by proactive exchange of cache index messages between them. To conserve energy, message exchanges are localized in a *cooperation zone*, the size of which is determined by setting a hop-to-live value for each cache index message. We develop an analytical model to evaluate the energy saving performance of the EPCOR scheme. By trading-off between performance improvement and the overhead of cache index messages, an algorithm is developed from the model to determine the optimal radius of the cooperation zone. Furthermore, in order to reduce the cost of proactive exchange of cache index message, a piggyback dissemination scheme is proposed to further enhance the performance of EPCOR scheme. Extensive simulation experiments are conducted to validate our analytical results. Theoretical as well as simulation results show that EPCOR can significantly improve the performance by saving energy, reducing access latency, increasing network throughput and balancing data traffic under different cache sizes, data update rates and mobility patterns. A preliminary version of this work are published in [32].

The rest of the paper is organized as follows. Section II gives an overview of the related work. Section III presents the detailed description of the EPCOR scheme. Section IV develops a model of our proposed system, and derives some analytical results to evaluate its performance. Implementation issues are discussed in Section V. Simulation results are discussed in Section VI. While, the last section concludes the paper.

II. RELATED WORK

Caching technology has been widely studied in the context of wired networks to improve the performance of World Wide Web. Wessels and Claffy [35] introduced the Internet cache protocol (ICP) to support communication between caching proxies via message exchanges. Cache digests [29] and summary cache [5] enable proxies to exchange information about cached content. Li, et al. [15] addressed the problem of placement of proxies and employed a dynamic programming method to determine an optimal placement. A cooperative hierarchical web caching architecture and its fundamental design principles are investigated in [4]. Recently, peer-to-peer mechanisms [11] [20] [34] have been proposed to share client cache content without proxy infrastructure. Squirrel [11] builds a cooperative web cache on top of the Pastry peer-to-peer routing substrate [28]. BuddyWeb [34] uses a custom peer-to-peer overlay among the clients to route object requests. Linga, et al. [20] proposed a churn-resistant peer-to-peer web caching system to handle churn attacks based on Kelips overlay. However, the problem addressed in wired networks is significantly different from that in wireless mobile environments. These protocols and architectures usually assume fixed network topology and require high computation and communication cost. Due to resource constraints (i.e., bandwidth, battery power, memory and computing capacity) portable devices and node mobility, the above techniques do not adapt well in wireless networks.

Lau, et al. [14] proposed a cooperative caching architecture for supporting continuous media proxy caching in mobile ad hoc networks. They introduced an application manager to transparently perform data location and session migration of continuous media streams among all proxy caches. To tolerate network partitions and improve data accessibility, Hara [9] proposed several replica allocation methods for ad hoc networks. In these schemes, the replicated data are relocated periodically based on access frequency and overall network topology. Although data replication can improve data accessibility, the overhead for relocating replicas periodically is significantly high in mobile ad hoc networks. Nuggehalli, et al. [23] addressed the problem of optimal cache placement in ad hoc wireless networks and proposed a greedy algorithm, called POACH, to minimize the weighted sum of energy expenditure and access delay. Papadopouli and Schulzinne [24] proposed the 7DS architecture, a peer-to-peer data sharing system, which defines a couple of new protocols for sharing and disseminating data among users that experience intermittent connectivity to the wireless Internet. A cooperation concept was introduced in 7DS for data sharing among all mobile hosts. Similar to our proposed work, Lim, et al. [16] suggested a cooperative caching scheme for Internet based mobile ad hoc networks. A broadcast based simple search scheme is proposed to establish cooperation among all MUs in the network to share the cached data items. Although the broadcast (flooding) based search can locate the nearest requested data item, the energy and bandwidth cost of the search is significantly high for a mobile ad hoc network. Sailhan and Issarny [31] limited the broadcast range in collaborative Web caching in ad hoc networks to minimize energy consumption and network load. They proposed a fixed broadcast range based on the underlying routing protocol. However, the mobile users' location, data popularity and network density often change in a real mobile environment, so the fixed broadcast scheme is hard to adapt to real mobile applications. Yin and Cao [37] investigated cooperative caching algorithms in ad hoc networks to support data access. They mainly focus on the problem of choosing data item or path information of data items to the cache of intermediate nodes when a data item is passing by. A hybrid caching scheme is proposed to use three thresholds - data item size, update frequency and an estimation of number saved hops, to decide whether to cache data or path information. However, since only the nodes on the route path between data source node and requesting node have chance to participate in cache cooperation, very limited performance improvement is achieved through cooperative caching in their schemes. Moreover, the optimal values of three thresholds used in the hybrid caching scheme changes dynamically with mobility, cache size, and network density. In effect the hybrid scheme is not easily adaptable to dynamic multi-hop wireless networks.

III. DESCRIPTION OF EPCOR SCHEME

Each mobile user in a hybrid network has dual communication capability. If an MU is within the transmission range of a base station, it accesses the information directly from the BS in a single hop; otherwise the MU uses peer-to-peer ad hoc links to retrieve data through multiple hops. Let the BS be a data source that contains a database of all requested data items. The BS may retrieve the data items from the Internet through a wired network, or from an attached infostation. A data query initiated by an MU is sent to the BS along with the routing path; upon receiving the request, the BS responds with the requested data item. A multi-hop routing protocol (e.g., [8], [25], [26]) is assumed to route data packets in the network.

In the EPCOR scheme, a peer-to-peer (P2P) overlay network is created among the mobile nodes to facilitate co-operative sharing of data. Two peers in the overlay network periodically exchange cache index messages to maintain their neighborhood relationship in the overlay. In order to conserve energy, the message exchange is localized in a cooperation zone. We define the cooperation zone (CZ_{ij}) of a MU, say MU_i, for sharing the data object d_j as a set of MUs in the overlay. Each MU that belongs to CZ_{ij} receives the cache index message, including the index of d_j from MU_i. The radius of a cooperation zone is defined as the maximal number of hops between MU_i and any other MU in the cooperation zone. In the EPCOR scheme, each MU uses a resource table to maintain the indices of the data items cached at neighboring MUs in the P2P overlay. For a cache miss, the MU looks up the resource table and forwards the query to a neighbor that has the requested data item. If more than one neighbor has a copy of the requested item, then the one that entails the least cost of data retrieval is chosen. With reference to Figure 1, when a data query is initiated in an MU, say A, it first looks for the data item in its own cache. If there is a cache miss, A checks if the item is cached in other neighboring peers, say B, in the P2P overlay; then the query is forwarded to B. Suppose the number of hops between A and B is less than those between A and BS, there is a saving of bandwidth and energy in retrieving the data item.

A. Data Structures

Figure 2 shows the main data structures maintained in each peer in the EPCOR system. These data structures are resource table (RT), neighbor table(NT), update list and cache. They are formally described below along with message formats:

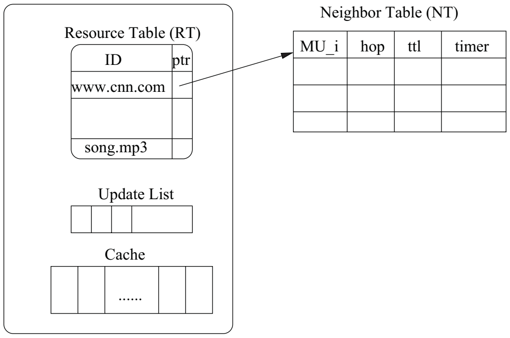

Fig. 2. Data structures at a mobile node

- $CL = \{d_1, d_2, ..., d_n\}$: set of cached data items.
- $UIM = \{< id_1, htl_1, ttl_1 >, ..., < id_n, htl_n, ttl_n >\}$: update index message where id_j is index of data item d_j; htl_j is its hop-to-live value; ttl_j is the time to live value of d_j, which is used to indicate d_j's validation.
- $L_{upd} = \{< id_1, flag >, ..., < id_n, flag >\}$: update list to record the updated data items since the last UIM exchange, where $flag$ bit indicates the states of the item (i.e., 0 for deleted item, 1 for added item).
- $RT_j =< id_j, ptr >$: the entry of resource table for data item d_j where ptr is the pointer to a neighbor table.
- $RT = \{RT_1, RT_2, ..., RT_n\}$: resource table which contains the index information for the data items cached in neighboring peers;
- $NT_j = \{< MU_1, hop_1, ttl_1, timer_1 >, ..., < MU_n, hop_n, ttl_n, timer_n >\}$: the neighbor table of RT_j, where hop_i is the number hops of the neighboring peer MU_i from current peer; ttl_i is the time to live value of the data item d_j cached at MU_i; $timer_i$ is the aging timer for the entry of MU_i.

In the EPCOR scheme, each MU maintains a resource table for the cache index information of all neighbors in the overlay. For each entry of a data object in the resource table, we maintain a neighbor table which includes the information of neighboring peers in the cooperation zone that have cached this data object. In order to handle mobility of MUs, we use a $timer$ in the each entry of neighbor table to indicate whether the neighboring peer currently resides in the cooperation zone or not.

B. *Peer-to-Peer Overlay Maintenance*

The P2P overlay of EPCOR scheme is maintained through periodic exchange of update index messages (UIM) among mobile peers. When a mobile peer successfully fetches (or evicts) a copy of an item, it records index information of the data items in the update list, L_{upd}. In one UIM exchange cycle, the mobile peer creates and disseminates a UIM message according to the information in L_{upd}. Peers share each data item in a cooperation zone. With each data item's index included in the UIM, we associate a *hop-to-live* (htl) value to indicate the radius of cooperation zone of the corresponding data item. A time to live (ttl) value is also included to indicate the estimation of valid time period of the corresponding item. When an MU, say MU_i, receives a UIM from a source, MU_s, the htl value of each entry in UIM is re-calculated. If the htl value of an entry reaches zero, it is deleted from the UIM to indicate that the UIM has reached the boundary of the cooperation zone for that data item in the overlay. The calculation of optimal cooperation radius is discussed in Section IV. In EPCOR, each MU maintains a resource table for the cache information of all neighbors in the overlay. After receiving a UIM from the source MU_s, for each entry with $ttl > 0$ in the UIM, the MU_i checks if the resource table already has an entry for that data item. If not, a new resource entry is created and inserted in the resource table and an entry of MU_s is included in the neighbor table. If the resource entry is already created at the resource table, the source MU is added to the neighbor table. If the neighbor table is full, the neighbor with lowest value of rate ttl/hop or with expired $timer$, is deleted from the neighbor table. For each entry with $ttl = 0$ in the UIM, the corresponding entry of the source MU_s in the neighbor table is deleted, because the data item cached in the MU_s is invalidated. In order to handle mobility of MUs, each entry of neighbor table has an aging timer value ($timer$). After receiving a UIM from MU_s, the MU_i needs to update all the neighbor entries of the MU_s by resetting the $timer$ to the initial value. When the $timer$ expires, the entry is deleted from the neighbor table, so that the MU that moved out of the cooperation zone can no longer participate in the cooperative cache sharing. The detailed algorithm of P2P overlay maintenance is shown in Figure 3.

Additionally, Figure 4 gives a simple example to illustrate the proposed EPCOR scheme. In this Figure, MU_1 is one hop away from MU_2 and MU_3, and 2 hops from MU_4. MU_1 caches d_1,...,d_5. When MU_1 constructs UIM, the htl values of d_1, d_2, d_3, d_4, d_5 are 2, 2, 2, 0 and 1 respectively. Thus, a neighbor table entry of MU_1 is added in the resource tables of MU_2 and MU_3 for the indices of d_1, d_2, d_3, and d_5. MU_1 is also included in the resource table of MU_4 for the indices of d_1, d_2 and d_3. The index of d_5 is deleted from the UIM relayed at MU_2 after the htl value of d_5 reaches zero. It is also shown in Figure 4 that since the htl values of all cached data items in MU_4 are less than 3, the cached data items' information is not included in the resource table of MU_3, which is 3 hops away from MU_4.

```
**Procedure UIM Exchange**{}
(1) In a UIM exchange cycle, $MU_i$ constructs a $UIM$ as follows:
    **For** each entry $< id_x, flag >$ in the update list $L_{upd}$
        **If** $flag == 1$
            Get $ttl_x$ value from cached data item $d_x$;
        **Else**
            $ttl_x = 0$;
        Calculate the optimal radius ($R_{ix}$) for sharing $d_x$;        /* calculate the radius of cooperation zone */
        **If** $R_{ix} > 0$
            $htl_x = R_{ix}$;
            Insert $< id_x, htl_x, ttl_x >$ into $UIM$;
    Forward $UIM$ to neighboring nodes;
    $L_{upd} = NULL$;        /* empty the update list */

(2) When $MU_i$ receives $UIM$ from $MU_s$ via a neighboring MU:
    Reset the $timer$ field of $MU_s$ entries in all neighbor tables;
    **If** current $UIM$ is equal to one of previously received $UIM$s
        Discard $UIM$ and return;
    **For** each entry $< id_x, htl_x, ttl_x >$ in $UIM$
        **If** $ttl_x > 0$
            Update the source table and add an entry of $MU_s$ to neighbor table;
        **If** $ttl_x = 0$
            delete the entry of $MU_s$ from the neighbor table;
        Re-calculate the optimal radius ($R_{sx}$) for sharing $d_x$;        /* re-calculate the radius of cooperation zone */
        **If** $R_{sx} \leq 0$
            delete $< id_x, htl_x, ttl_x >$ entry from the $UIM$;
    **If** $UIM \neq NULL$
        Forward $UIM$ to neighboring nodes;
```

Fig. 3. UIM Exchange Algorithm of EPCOR

C. Object Lookup and Cache Management

When MU_i initiates a data query, it first checks the local cache for the requested data item. On a cache miss, MU_i searches its resource table. If the requested data item is cached by neighboring peers in the P2P overlay, the query is forwarded to the neighbor that has the least hop. Otherwise, the query is forwarded to the BS to retrieve the data item. As shown in Figure 4, if MU_1 requests for d_{10}, the request message is forwarded to MU_4. If MU_i receives a query from a source, MU_s, MU_i first checks its local cache; if available, the data item is sent to MU_s. If the requested data is not in MU_i's cache, the query is forwarded to the neighboring peer MU_x that cached the requested item and has the least hop from MU_i. If MU_i finds itself as the destination of the query and does not cache the requested item, it means that the source MU_s has inconsistent cache index information of MU_i. In this case, the query is forwarded to the BS to retrieve the data item. For example, in Figure 4, MU_4 receives a request message for d_8 from MU_2. In this case, the request message is forwarded to the BS to retrieve the requested data item. Inconsistencies may occur due to loss of UIM or transmission errors. When the source MU receives the requested data item from other MUs or the BS, a cache replacement policy, e.g., least recently used (LRU), is used to evict the existing data items to make space for

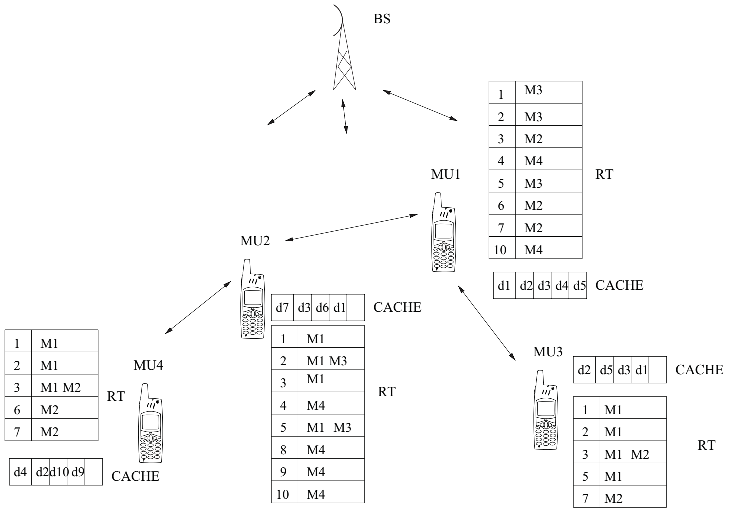

Fig. 4. A simple example of EPCOR scheme

the incoming data item. The indices of data items that are evicted from the cache are kept in the update list (L_{upd}) as items with $flag = 0$; and the indices of incoming data items with $flag = 1$ are kept. The update list is used for the construction of next UIM. The algorithm for object lookup and cache management is formally described in Figure 5.

IV. MATHEMATICAL ANALYSIS

In this section, we develop an analytical model of the EPCOR scheme to evaluate its performance. An algorithm is developed from the analytical model to calculate optimal radius of the cooperation zone of each data sharing based on energy efficiency.

A. An Energy Model

In [6] [7], Feeney and Nilsson have reported a detailed energy consumption model for performance analysis of mobile ad hoc networks based on energy consumption measurements of some commercially-available IEEE 802.11 NICs operating in ad hoc mode. To the best of our knowledge, this is a most accurate model of energy consumption in ad hoc networks in the literature. In our analysis, we use this model to calculate the energy consumption of a MU for sending and receiving a message in unicast or broadcast mode. Our analysis is independent of the energy model, other energy models can also be used in our analysis of the EPCOR scheme.

We briefly introduce the energy consumption model presented in [6] and [7]. The energy model and its detailed description about IEEE 802.11 properties are not the focus of this work. Readers can find more details in [6] and [7]. According to their model, energy consumed by a mobile user using IEEE 802.11 wireless interface for sending,

```
Procedure Cache Management{}
(1) When $MU_i$ initiates a query for data item $d_j$
    If $d_j$ is in cache    /* a cache hit*/
        return $d_j$;
    Else
        If $d_j$ is in the resource table;
            choose the neighbor $MU_x$ with the least hop;
            send the query to $MU_x$;
        Else
            send the query to base station;

(2) When $MU_i$ receives a query;
    If requested item $d_j$ is in cache    /* a peer cache hit*/
        send $d_j$ to source peer;
    Else
        If $id_j$ is in resource table;
            choose the neighbor $MU_x$ which has the least hop;
            send query to $MU_x$;
        Else
            If the destination of query is $MU_i$
                send query to base station;
            Else
                send query to the destination;

(3) When $MU_i$ receives a data item $d_j$;
    If residual cache space is less than size of $d_j$;
        Evict victim item $d_t$ by using cache replacement policy;
        $L_{upd} = L_{upd} \cup < id_t, 0 >$;
    cache $d_j$;
    $L_{upd} = L_{upd} \cup < id_j, 1 >$;
```

Fig. 5. Caching Algorithm of EPCOR Scheme

receiving or discarding a message is given by

$$E(s) = m \times s + b \tag{1}$$

where s is the message size, m denotes the incremental energy cost associated with a message, and b is the energy cost for the overhead of message. The parameters m and b are different for sending and receiving messages. In particular, the energy model proposed in [6] describes the energy consumption for broadcast and unicast traffic of IEEE 802.11 NICs operating in ad hoc mode.

1) Broadcast Traffic: IEEE 802.11 standard [38] uses Carrier Sensing Multiple Access with Collision Avoidance (CSMA/CA) technology to avoid frames collision and share channel between multiple senders and receivers. In CSMA/CA, before sending a broadcast packet, the sender listens briefly to the channel. If the channel is clear, the message is sent and received by all nodes in the wireless (radio) range. Otherwise, the sender must back off and try later. In [6], the energy cost associated with sending a broadcast packet is given by:

$$E_{bd\_sd}(s) = m_{bd\_sd} \times s + b_{bd\_sd} \tag{2}$$

The cost associated in receiving a broadcast packet is:

$$E_{bd\_rv}(s) = m_{bd\_rv} \times s + b_{bd\_rv} \tag{3}$$

2) Unicast Traffic: For unicast traffic in order to present collisions caused by hidden terminals, a mechanism called virtual carrier sensing is used in addition to CSMA/CA in IEEE 802.11 standard. If a source wants (or desires) to transmit a message, it first sends a Request-To-Send (RTS) control message, identifying the destination. The destination responds with a Clear-To-Send (CTS) message. Upon receiving the CTS, the source sends the data and awaits an ACK from the destination. Thus, the energy cost model for unicast send as shown in [6] is,

$$E_{p2p\_sd}(s) = b_{sendctl} + b_{recvctl} + m_{p2p\_sd} \times s + b_{p2p\_sd} + b_{recvctl} \tag{4}$$

where $b_{sendctl}$ and $b_{recvctl}$ are the energy costs for send and receive of small control messages (RTS, CTS and ACK). The cost of a unicast receive is given by,

$$E_{p2p\_rv}(s) = b_{recvctl} + b_{sendctl} + m_{p2p\_rv} \times s + b_{p2p\_rv} + b_{sendctl} \tag{5}$$

Based on the energy model given in Equations (2)-(3), we derive the energy cost of broadcasting a message in ad hoc networks. When a packet is broadcast, all nodes within the transmission range of the sender will receive it. Defining the fixed channel access costs b_{bd_rv} and b_{bd_sd} and the incremental payload cost m_{bd_sd} and m_{bd_rv}, if r is the radio transmission range, the average number of receivers within the transmission range of the sender is given by $\rho \times \pi \times r^2$, where ρ is node density of the network. Thus, the total energy cost associated with a broadcast send and receive is given by,

$$E_{total\_bd}(s) = E_{bd\_sd}(s) + \rho \times \pi \times r^2 \times E_{bd\_rv}(s) \tag{6}$$

In ad hoc networks, the energy cost of relaying a message at an intermediate node is given by,

$$E_{relay}(s) = E_{p2p\_rv}(s) + E_{p2p\_sd}(s) \tag{7}$$

In IEEE 820.11, messages may be lost due to collision or other failures and the protocol provides for various MAC layer retransmissions. Therefore, each element of protocol in Equations (4)(5)(7) includes an implicit factor of

$(1+N_{retransmissions/duplicates})$ where $N_{retransmissions/duplicates}$ is the average number of retransmissions for sending a packet in the network. Based on this energy model, we analyze the energy cost performance of EPCOR scheme in the next subsection.

B. Assumptions and Notations

Before presenting the analytical model of EPCOR scheme, we introduce the assumptions made in our analysis as follows.

1) All nodes in the network are two-dimensionally Poisson distributed with density ρ, i.e., the probability $p(i, A)$ of finding i nodes in an area of size A is given by,

$$p(i, A) = \frac{(\rho A)^i e^{-\rho A}}{i!} \tag{8}$$

2) All nodes have the same transmission and receiving range, denoted by r. N is the average number of neighbor nodes within a circular region of radius r. Therefore, we have $N = \rho \pi r^2$.
3) All proxy nodes that exchange data traffic with BS reside within circular area of radius of r from the BS.
4) The energy cost of data processing is negligible compared with the energy cost of data communication.
5) All nodes use least recently used (LRU) replacement policy to manage their cache.
6) The data requests of each node follow a Poisson process.

According to [27], the energy cost for transmitting 1K bits of information is approximately the same as the energy consumed to execute 3 million instructions. Therefore, we only consider the energy cost of data communications in our analysis. The other notations used in the analytical model are listed as follows.

- H_i: number of route hops between BS and MU_i;
- H_{ik}: number of route hops between MU_i and MU_k ;
- s_i: size (bytes) of a data item d_i;
- s_{uim}: size of an update index message (UIM);
- s_q: size of a query message;
- CZ_{ij}: the set of MUs in cooperation zone of MU_i for sharing data item d_j;
- R_{ij}: radius of cooperation zone CZ_{ij};
- λ_{ij}^{q}: MU_i's query rate for data item d_j;
- λ_{ij}^{uim}: MU_i's UIM dissemination rate for data item d_j;

- $f_{ij} = \frac{\lambda_{ij}^{uim}}{\lambda_{ij}^{q}}$: MU_i's relative UIM dissemination rate compared with query rate for d_j;
- P_{ij}: cache hit ratio of data item d_j in MU_i;
- E_{ij}: overall energy cost of MU_i for accessing the data item d_j;
- ρ: node density of the network.

C. Problem Formulation

In the EPCOR scheme, each MU maintains a cooperation zone in the localized P2P overlay for each data item by setting the htl value in UIM. When MU_i queries for a data item d_j, if d_j is located in the cooperation zone, MU_i will retrieve the data item from the nearest neighbor node, otherwise the query message will be routed to the BS to retrieve the item. The overall energy cost (E_{ij}) of MU_i accessing a data item d_j can be calculated by considering the energy cost (E_{ij}^1) used to disseminate UIM messages in the P2P overlay and the energy cost (E_{ij}^2) used to retrieve d_j. Thus,

$$E_{ij} = E_{ij}^1 + E_{ij}^2 \tag{9}$$

Since the radius of cooperation zone CZ_{ij} is R_{ij}, the average number of MUs in the CZ_{ij} is $\rho\pi R_{ij}^2$. The relative UIM dissemination rate is f_{ij}. In a UIM dissemination process, each MU that belongs to CZ_{ij} will send the UIM once and receive the same UIM from all neighbors multiple times. Thus, according to the UIM dissemination algorithm (see Fig. 3) and Equation (6), E_{ij}^1 is calculated as:

$$E_{ij}^1 = \rho\pi R_{ij}^2 \times f_{ij} \times E_{total\_bd}(s_{uim}) \tag{10}$$

According to the caching algorithm in Fig. 5, as MU_i queries for d_j, if d_j is cached in MU_i, we assume the energy cost of data retrieval is zero. If d_j is cached in some MUs belonging to CZ_{ij}, the item is retrieved from the nearest node, so the average energy cost (e_{ij}^1) of this case is calculated as:

$$e_{ij}^1 = \sum_{l=1}^{R_{ij}} l(1 - \prod_{H_{ti}=l}(1 - P_{tj})) \prod_{H_{ki}<l}(1 - P_{kj}) \times [E_{relay}(s_q) + E_{relay}(s_j)] \tag{11}$$

If d_j is not cached in any MU within the CZ_{ij}, the query is forwarded to the BS. The requested item is retrieved from the BS. Thus, the average energy cost (e_{ij}^2) for this case is calculated as follows:

$$e_{ij}^2 = \prod_{H_{ti} \leq R_{ij}} (1 - P_{tj}) \times H_i \times [E_{relay}(s_q) + E_{relay}(s_j)] \tag{12}$$

Therefore, the average energy cost (E_{ij}^2) used to retrieve d_j is given as,

$$E_{ij}^2 = e_{ij}^1 + e_{ij}^2 \tag{13}$$

The total energy cost of any MU_i to access a data item d_j is thus expressed as follows,

$$\begin{aligned} E_{ij} &= \rho\pi R_{ij}^2 \times f_{ij} \times E_{total\_bd}(s_{uim}) \\ &+ \left[\sum_{r=1}^{R_{ij}} r(1 - \prod_{H_{ti}=r} (1-P_{tj})) \prod_{H_{ki}<r} (1-P_{kj}) + \prod_{MU_t \in CZ_{ij}} (1-P_{tj}) H_i \right] \times [E_{relay}(s_q) + E_{relay}(s_j)] \end{aligned}$$

In order to optimize the energy consumption of whole network, we need to minimize the energy cost (E_{ij}) due to each data access at each node by choosing the optimal size of the cooperation zone.

Theorem 1: For any MU_i in the network, there exists an optimal radius $0 \le R_{ij}^{opt} \le H_i$ of the cooperation zone for sharing any data item d_j, so that the overall energy cost E_{ij} of accessing d_j is minimal.

Proof: In order to prove the existence of R_{ij}^{opt}, we first calculate how much energy saving we can achieve, if we increase the size of cooperation zone. Given a radius R, we define ΔE_{ij}^1 as the increased energy cost due to UIM dissemination if the radius is increased by one (from R to $R+1$). ΔE_{ij}^2 is defined as the saved energy cost of data retrieval due to bigger cooperation zone if the radius is increased by one. According to Equation (10), ΔE_{ij}^1 is calculated as

$$\Delta E_{ij}^1 = \rho\pi(2R+1) \times f_{ij} \times E_{total\_bd}(s_{uim}) \tag{14}$$

and according to Equation (11)-(13), ΔE_{ij}^2 is calculated as,

$$\Delta E_{ij}^2 = \prod_{H_{ti} \le R} (1-P_{tj})(1 - \prod_{H_{ti}=R+1} (1-P_{tj}))(R+1-H_i) \times [E_{relay}(s_q) + E_{relay}(s_j)] \tag{15}$$

In order to make the problem tractable, we use average cache hit ratio (P_j) of all nodes belonging to CZ_{ij} for data item d_j to calculate the probability of caching d_j in any MU belonging to CZ_{ij}.

$$P_j = \frac{\sum_{H_{ti} \le R} P_{tj}}{\rho\pi R^2} \tag{16}$$

According to the nodes Poisson distribution assumption as in Equation (8), the probability that d_j is not cached in any node within a circular area of radius R is calculated as,

$$\sum_{i=0}^{\infty} (1-P_j)^i \frac{(\rho\pi R^2)^i}{i!} e^{-\rho\pi R^2} = e^{-P_j \rho\pi R^2} \tag{17}$$

Thus, we derive ΔE_{ij}^2 as,

$$\Delta E_{ij}^2 = e^{-P_j \rho \pi R^2}(1 - e^{-P_j \rho \pi (2R+1)})(R + 1 - H_i) \times [E_{relay}(s_q) + E_{relay}(s_j)] \tag{18}$$

$R < H_i$, $\Delta E_{ij}^2(R) \leq 0$ implies we can always achieve an energy saving of $|\Delta E_{ij}^2|$ for each data retrieval of d_j, as the radius of the cooperation zone increases by one. $0 \leq R < H_i$, $\Delta E_{ij}^1 > 0$, implies energy cost of UIM message dissemination increases with the radius of the cooperation zone. If we want to achieve minimal energy cost of each data access of d_j, we need to make sure $\Delta E_{ij}^1 + \Delta E_{ij}^2 < 0$ for each step of UIM dissemination.

The proposed iterative algorithm find the optimal radius R_{ij}^{opt}. Initially, R is set as 0. At each step, R is incremented by 1. In each step, if $\Delta E_{ij}^1 > |\Delta E_{ij}^2|$, the iteration stops and optimal radius, $R_{ij}^{opt} = R$. $R = H_i - 1$ implies $\Delta E_{ij}^2 = 0$ and $\Delta E_{ij}^1 > 0$, so $\Delta E_{ij}^1 > |\Delta E_{ij}^2|$, thus, the iteration will always stop and find the optimal radius before R increases to H_i. Therefore, for any MU_i, the algorithm can find an optimal radius $0 \leq R_{ij}^{opt} \leq H_i$ for the cooperation zone of sharing the data item d_j, so that the energy cost E_{ij} of each retrieval of d_j is minimal. ■

D. Numerical Results

In this subsection, we present some numerical results on the performance of EPCOR based on the numerical model derived above. We evaluate the EPCOR scheme under a hybrid network with one BS and density of $\rho = 100$ nodes/km^2 and node communication range of $r = 200$ m. The size of a query message is $s_q = 10$ bytes, and the size of UIM is $s_{uim} = 3$ bytes/item. Each MU has the same data access pattern. We consider the energy consumption of MU_i that is $H_i = 10$ hops away from the BS. Also, MU_i accesses a data item (d_j) of size $s_j = 100K$ bytes and average cache hit ratio $P_j = 0.01$.

1) Analysis of an Individual MU: First, we investigate the energy consumption of data requests of an individual MU under different data sizes, cache probabilities, and distances from the BS. We assume that indices of 10 items are grouped for one UIM exchange ($f_{ij} = 0.1$). As shown in Figure 6(a), there exists an optimal radius for energy cost of data requests for different data sizes. The amount of energy saving for the request of large data items (e.g., $s_j = 1MB$) is much more than that of small data items. This is because the amount of energy cost of data retrieval (E_{ij}^2) is dominant in the overall energy cost (E_{ij}) when the size of item is large. As the size decreases, we can still save some energy by choosing the optimal radius. If the data size is too small, we set the radius as zero (no sharing with other peers) to prevent the MU from wasting energy.

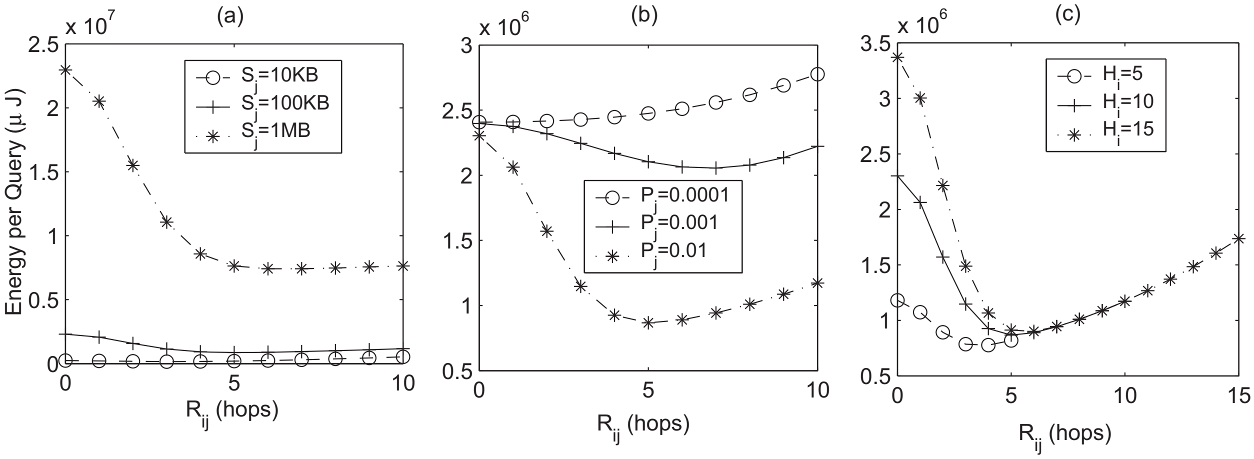

Fig. 6. Analytical Results for Energy Efficiency Performance of an Individual MU: (a) different sizes of data items, (b) different cached probabilities, (c) different distances from BS.

Figure 6(b) shows that the overall energy cost of accessing a data item with high average cache ratio (P_j) is less than that of accessing an item with low average cache ratio. This is because an MU has a higher chance to obtain the popular data items either from its own cache or the neighboring MUs, which reduces the number of hops for a data retrieval. In order to obtain the data item whose average cache hit ratio is low, the MU needs to search a wider area, thus the optimal radius for a less popular data item is larger than that for more popular data items. As shown in Figure 6(b), when the cached probability is 0.0001, the data item does not need to maintain a cooperation zone (i.e., optimal radius equal to zero) for energy efficiency purpose.

Next, we vary the MU's distance (H_i) from the BS to examine its impact on the energy efficiency. As shown in Figure 6(c), if there is no cooperation among the MUs (i.e., radius equal to zero), the MU that is 15 hops away from the BS incurs 200% more energy to retrieve the data item than an MU that is 5 hops away from the BS. If we assume that each MU uses optimal radius to form its cooperation zone, the MU with 15 hops needs about 20% more energy to retrieve the data item than the one that is 5 hops away. This feature of EPCOR significantly improves the fairness of energy consumption of each MU at different locations, thus improving the life time of the entire network.

2) Analysis of the Entire Network: As shown in Figure 7, we consider a hybrid network in a circular area with radius of L, and the BS resides at the center of the circle. If each MU uses its optimal cooperation radius to form the cooperation zone for each item sharing, we can calculate the optimal energy cost for each data request of MUs that have different distances from the BS by using the iteration algorithm in proof of Theorem 1. Let $E(R_k^{opt})$ denote average

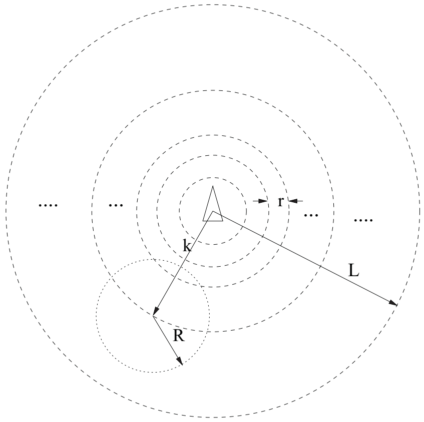

Fig. 7. A Hyrbrid Wireless Network with Radius L

energy cost for each data request of MUs that are k hops away from the BS. In the network, the average number of MUs that are k hops away from the BS is equal to $\rho\pi(k^2 - (k-1)^2)$ where ρ is the node density. The total average number of MUs in the network is $\rho\pi L^2$.

We use the average *energy per query* (EPQ) as a metric to evaluate the energy consumption of each query in the whole network. EPQ is defined as the sum of energy consumptions of all MUs divided by the total number of queries in a given period. Thus, in the time period t, each MU has $\lambda \times t$ data requests, where λ is the average query rate from each MU. There are $L^2\pi\rho\lambda t$ data requests generated in the whole network. The energy consumption of the MUs that are k hops away from the BS is denoted as $E(R_k^{opt})$. Therefore, the EPQ performance of EPCOR for the whole network is given as:

$$EPQ = \frac{\sum_{k=1}^{L}(k^2 - (k-1)^2)\pi\rho E(R_k^{opt})\lambda_j t}{L^2\pi\rho\lambda t} = \frac{\sum_{k=1}^{L}(2k-1)E(R_k^{opt})}{L^2} \tag{19}$$

Since the wireless channel bandwidth of BS is the bottleneck in determining the throughput of hybrid wireless networks, we use another metric, called *base station retrieval ratio* (BRR) to evaluate the burden of base station. BRR is defined as the number of data items retrieved from the BS divided by the total number of requested data items of all MUs in the network. If BRR is high, it means that the channel is more congested since more data items are retrieved from the BS. According to Equation (17), the probability of a cache miss in the cooperation zone of the MU that is k hops from the BS is $e^{-P_i\pi\rho(R_k^{opt})^2}$. Therefore, the BRR of the whole network is given by:

$$BRR = \frac{\sum_{k=1}^{L}(k^2 - (k-1)^2)\pi\rho e^{-P_i\pi\rho(R_k^{opt})^2}\lambda t}{L^2\pi\rho\lambda t} = \frac{\sum_{k=1}^{L}(2k-1)e^{-P_i\pi\rho(R_k^{opt})^2}}{L^2} \tag{20}$$

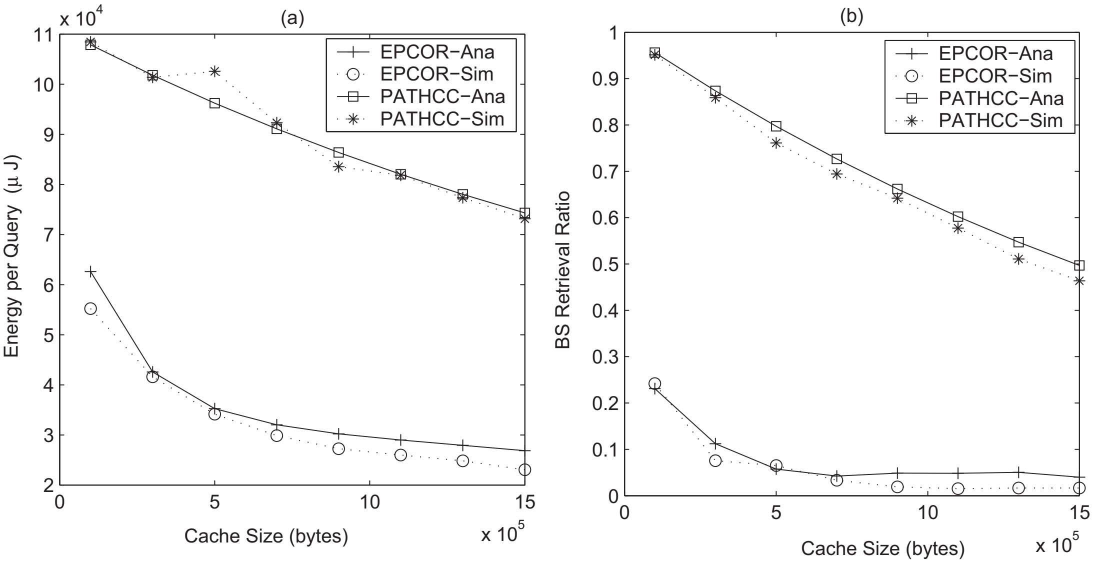

Fig. 8. Analytical and simulation results of a whole network (a) Energy per Query, (b) Base Station Retrieval Ratio

Now we present the analytical results of the EPQ and BRR performances of a simple hybrid wireless network. The analytical (Ana) results are compared with the simulation (Sim) results. We consider a hybrid wireless network in a circular area of radius $L = 1200$ meters. Each MU has same the cache space and transmission range ($r = 200$ meters). There are 1000 data items in the system, each of size of 10 Kbytes and same popularity for all MUs. The size of UIM is $s_{uim} = 2$ bytes/item. In this evaluation, we compare EPCOR with path cooperative caching (PATHCC) [37], in which the cooperation radius is set to zero and only the nodes that reside in the route path from the source node to the BS participate in the cooperation to share the cache data. In our simulations, all MUs are assumed to be stationary and the packet delivery to nodes is instantaneous and error-free. This simulation thus faithfully represents the request generation and data retrieval for the different cache schemes.

As shown in Figure 8(a), the EPQ performance of both EPCOR and PATHCC drops as the cache size increases. A network with EPCOR deployed saves almost 70% energy compared to that using PATHCC. Figure 8(b) shows that EPCOR has much lower BRR than PATHCC, thus significantly reducing the traffic at the base station. Both Figures 8(a) and (b) demonstrate that the simulation results match the analytical results very closely, thus validating our analytical model of system.

V. IMPLEMENTATION ISSUES

In this subsection, we address four critical implementation issues, namely estimation of run-time parameters, incremental calculation of optimal radius, maintenance of data consistency and piggyback UIM message dissemination.

Additionally, the time and space complexity of EPCOR are also discussed.

A. Estimation of Run-time Parameters

To implement the proposed algorithms, the parameter estimation mechanisms must be addressed. As shown in Equations (14) and (15), several parameters are involved in calculating the optimal radius. The incremental energy coefficient (m) and the overhead (b) in Equation (1) are system parameters obtained from specifications of the wireless network card. The UIM dissemination frequency (λ_{ij}^{uim}), size of UIM (s_{uim}) and size of request message (s_q) are obtained from the settings of EPCOR. Data query rate λ_{ij}^{q} can be estimated by using the sliding average method [33]. The distance H_i of each MU in terms of the number of hops from the BS is available at the routing table. The node density (ρ) of the whole network can be estimated at the BS and disseminated to all MUs.

In order to estimate the cache hit ratio of data items, we assign a metadata for each requested data item d_j (including local and peer requests) in the MU_i. The metadata includes two counters: query counter (c_j^q) and hit counter (c_j^h). When MU_i initializes a query for data item d_j, or MU_i receives a query from other peers, the corresponding request counter c_j^q is incremented by 1. When MU_i satisfies a query for d_j, the hit counter c_j^h is incremented by 1. Thus, the hit ratio of data item d_j at MU_i is estimated as $P_{ij} = c_j^q / c_j^h$.

B. Incremental Calculation of Optimal Radius

In Equation (18), we use average cache hit ratio (P_j) to calculate the optimal radius. MUs have different cache hit ratios for the same data item and the hit ratio of each item may dynamically change in an MU. Therefore, in order to efficiently estimate P_j dynamically and calculate the optimal radius R_{ij}, we use an incremental method.

In this method, P_j is added as an additional attribute to the entry of UIM for d_j. When MU creates a UIM message, its local hit ratio of d_j is set as the value P_j. In each broadcast step, if MU_i receives a UIM from neighboring MU, this attribute is updated as $P_j = \frac{P_j * n + P_{ij}}{n+1}$, where n is the number of MUs the UIM has visited after leaving from the source MU and P_{ij} is the hit ratio of d_j at MU_i. After MU_i receives the UIM, ΔE_{ij}^1 and ΔE_{ij}^2 are calculated according to Equations (14) and (18). According to the proof of Theorem 1, if $\Delta E_{ij}^1 > |\Delta E_{ij}^2|$, MU_i stops dissemination of the UIM message. Otherwise, MU_i broadcasts the UIM message to all neighbor nodes. Therefore, the average cache hit ratio P_j and optimal radius are incrementally calculated as UIM propagates through the cooperation zone in each step of dissemination. Since this method does not require global cache hit ratio information of all MUs in the cooperation zone, the method can significantly reduce the cost of the calculation of optimal radius.

C. Data Consistency

In this paper, we assume that all data updating occurs at the BS. The data item is associated with a time-to-live (TTL) value after retrieval from the BS. When an MU receives a query for a cached item d_j whose TTL value has not expired, the MU responds to query with the valid data item. If the requested data item is stale, the request is forwarded to the BS or other peers to retrieve the valid one. In estimating the hit ratio, we consider a stale hit the same as a cache miss, so a data item with low TTL value (i.e., invalidated frequently) has a high cache miss ratio.

D. Piggyback UIM Message Dissemination

In most ad-hoc routing protocols (e.g., AODV [26], DSDV [25], or ZRP[8]), two neighboring MUs need to frequently exchange hello messages in routing layers to discover and maintain their neighborhood relationship. Due to the channel contention [7], the energy cost of independently sending a small size message is much more than the energy cost of piggybacking with other messages. This feature motivates us to use a piggyback dissemination scheme to propagate UIM messages of EPCOR through the P2P overlay network. In the piggyback dissemination scheme, after a UIM message is created as shown in Figure 3, a pending message queue is used to store the created UIM messages. When the routing layer notifies a hello message exchange, the UIM messages in the pending queue are piggybacked with hello message, and broadcast to all neighboring nodes. The neighboring nodes can extract UIM message from the received hello message and update their resource table and neighbor table. If the UIM dissemination is not stopped, the UIM message will be inserted into pending queue for subsequent broadcast. Otherwise, the UIM message is deleted from the pending queue. Since the piggyback dissemination does not require each MU proactively sending UIM messages, it significantly reduces the energy and bandwidth costs of UIM message dissemination. Furthermore, it also eliminates the chance that nodes cannot go to the sleep mode due to UIM message dissemination, hence it improves the energy saving at each node.

E. Complexity

From Figure 3, the time complexity to construct and process a UIM is $O(n)$, where n is length of L_{upd}. Each MU needs $O(D*M)$ space to maintain a resource table and all neighbor tables, where D is the maximum number of entries of a resource table and M is the maximin number of MUs in a neighbor table. From Figure 5, a lookup operation in resource table requires $O(log\ D + log\ M)$ time, if we use priority queues to maintain the resource entries and neighbor entries.

VI. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed EPCOR scheme through simulation experiments. Different metrics are used to evaluate four main aspects of its performance. They are energy efficiency, access latency, throughput and load balance.

A. Simulation Model and System Parameters

We use the Network Simulator (*NS-2*) [40] to study the performance of EPCOR. Since EPCOR does not rely on the specific routing protocol, in our simulations, we use DSDV [25] as the routing protocol to route data traffic in the hybrid wireless network. Our simulations use IEEE 802.11 as the MAC protocol and also use two-ray ground reflection as the radio propagation model [30]. The default transmission range of each MU is $r = 200$ meters. For the mobility model, as shown in Figure 9, we assume a service area of 1500m × 1500m. One fixed base station is located at the coordinate (750m, 0m). All mobile nodes moving in this area follow the random way point mobility model [3], which is commonly used to model the movement of individual pedestrians. According to this model, each mobile node starts at a location chosen uniformly at random inside the service area. For each movement, the target location is also randomly chosen, and the moving speed is uniformly random in the range $[0, v_{max}]$. After the mobile node reaches its destination, it pauses for a period of time (t_p) before continuing its next movement.

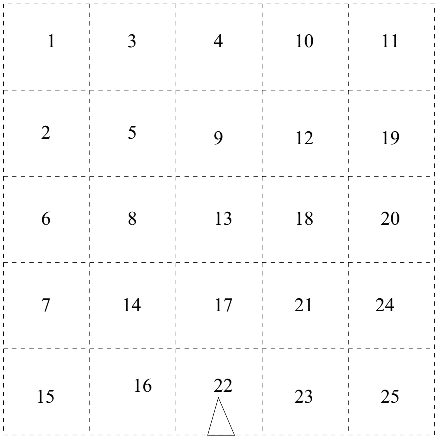

Fig. 9. A Simulated Hybrid Network

For Web data, the data size distribution follows the lognormal model [2]. In our simulation also, the lognormal model is used for the size distribution of all data items in the system. The cutoff minimal size of the data item is assumed to be 128 bytes, the cutoff maximal size is 40 Kbytes, and the average size is 12.5 Kbytes. The update intervals (i.e., time-to-live values) of all data items follow uniform distribution in the range of [0, ttl_{max}]. The data requests of each mobile user follow a Poisson process. After a request is sent out, if the MU does not receive the data item, it waits for an interval (t_w) before resending the same request message. The MU does not generate new request until the query is served. We consider *Zipf-like* [1] distribution for the data popularity pattern of MU's access, in which the access probability a_i of data item d_i is proportional to its popularity rank, *rank(i)*. That is, $a_i = A/rank(i)^{\theta}$, where A is the normalization constant and θ is a parameter between 0 and 1. As shown in Figure 9, the entire service area is equally divided into 5 × 5 square grids. Beginning from the left lowest grid, we index the grids as 1, 2, 3,..., 25 in a zig-zag diagonal-wise fashion. The MUs in the same grid have the same *Zipf* based data popularity pattern. MUs in different grids have different shift values for the *Zipf* pattern. For an MU in grid i, the *id* of data access is shifted by i so that $id = (id + i)$ mod N, where N is the total number of data items in the system. According to this model, the data access of two nearby grids show high similarity while the data access of two far-off grids show a low similarity. In the simulations, a first come and first serve (FCFS) policy is used at the BS to serve the incoming data requests. The settings of system parameter are given in Table I. Some parameters may vary in the following experiments.

TABLE I
SYSTEM PARAMETER SETTING

Parameters	Default values	Range of values
Number of mobile devices	120	N/A
Number of data items	3000	N/A
Cache size	1500 $Kbytes$	500-2500 $Kbytes$
Mean access rate (λ_i)	1/30 sec^{-1}	1/10-1/40 sec^{-1}
Average Update Interval	10000 sec	625-10000 sec
Pause time (t_p)	2 sec	N/A
Max moving speed (v_{max})	2 m/s	2-20 m/s
Initial value of aging timer ($timer$)	120 sec	N/A
Bandwidth	11 $Mbps$	N/A
UIM exchange Interval (T)	60 sec	N/A
Waiting interval (t_w)	5 sec	N/A
Request message size (s_q)	10 $bytes$	N/A
UIM message size (s_{uim})	3 $bytes/item$	N/A

B. Simulation Results

For each experiment, the total simulation time is 5000 seconds and the results after initial 1000 seconds are recorded for performance evaluation. All MUs' caches are filled with randomly chosen data items at the initial state. Four major metrics are used in the evaluation: *hit ratio* (HR), *peer hit ratio* (PHR), *energy per query* (EPQ) and *average latency*. We define PHR as the ratio of the number of data items retrieved from peer mobile users to the total number of requested data items of a mobile user. Recall that EPQ is the ratio of the total energy consumed in a given period to the total number of requests in that period. Under the same data request pattern, EPQ reflects the energy consumption performance of different schemes. Energy consumption of each mobile device is measured by the linear model proposed in [7], as explained in Section IV.

In the performance study, two variants of EPCOR scheme are evaluated. In basic EPCOR scheme (EPCOR), each MU disseminates UIM messages independently. In the enhanced scheme (EPCOR-PG), MUs piggyback UIM messages with the underlaying routing hello messages as described in Section V.D. These two variants of EPCOR scheme are compared with three other schemes, namely NOCACHE, PATHCC and EXPRING. In NOCACHE scheme, no cache space is used for each MU so that every data item is retrieved from the base station. In PATHCC (path cooperative caching) scheme [37], each node does not have resource table to maintain the neighbors information so that each data request is forwarded to the BS, and only the nodes on the route path from the source node to the BS participate in the cooperation to share the cached data. A broadcast search based cooperative caching scheme has been proposed in [16] for the hybrid wireless networks. In our comparison, in order to reduce data traffic of request message flooding, the expanding ring scheme (EXPRING) [22] is deployed to set the range of request message flooding. This algorithm can be viewed as successive instantiation of flooding search with increase in TTL value ranging from 1 to the number of hops from the BS. In the default setting, the waiting time between two consecutive broadcasts is 5 seconds. In our simulations, the same data access patten and mobility model are applied to all above schemes. The same least recently used (LRU) cache replacement policy is also deployed in PATHCC, EXPRING and EPCOR schemes.

(1) Effect of Cache Size

In this experiment, we evaluate the performance of EPCOR under different cache sizes and compare with other three schemes. In Figure 10, the lower part of each column represents the hit ratio (HR) of the corresponding scheme, and the upper part of each column represents the peer hit ratio (PHR) of the scheme. We observe that for identical cache sizes, four schemes show almost the same HR performance, since all four schemes use the LRU cache replacement policy. The EXPRING and EPCOR-PG schemes have the best PHR performance (note that EXPRING uses flooding to locate the nearest requested data item in the network). The EPCOR outperforms the PATHCC in terms of PHR, and NOCACHE have zero HR and PHR performance since no cache is used in each MU. The enhanced EPCOR-PG significantly improves PHR performance over basic EPCOR, since the former uses piggyback method to disseminate UIM messages which reduces the cost of UIM dissemination so that each MU can maintain a larger cooperation zone for each data item. Figure 10 also shows that the HR and PHR performance of EXPRING, EPCOR and PATHCC schemes increase with larger cache sizes.

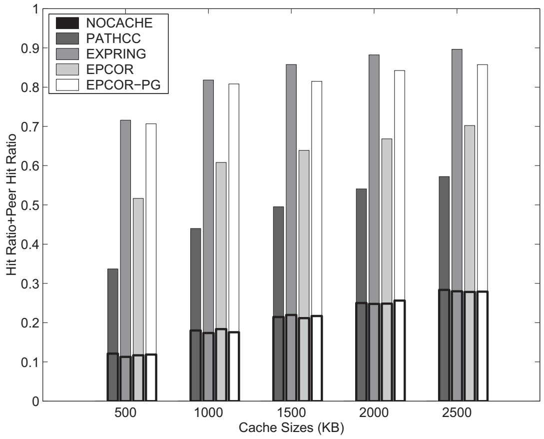

Fig. 10. Hit Ratio (HR) and Peer Hit Ratio (PHR) vs Cache Size

Figure 11 shows that EPCOR performs better in terms of EPQ than the other three schemes. This is because EPCOR maintains a P2P overlay to share the cached data, so that the number of hops for one data item retrieval is minimized. Moreover, the energy cost of UIM dissemination is limited by using optimal radius value. Although EXPRING can locate the nearest requested data item, the energy cost of the flooding search is huge for a wireless network, so EXPRING has the worst EPQ performance. Because PATHCC deploys a cache at each MU and the peers between the source node and BS participate in the cache sharing, PATHCC consumes less energy for each data retrieval than NOCACHE which does not deploy cache for each peer. In Figure 11, the basic EPCOR scheme consumes about 40% less energy than PATHCC. The enhanced EPCOR-PG scheme has better EPQ performance than the basic one. This is because of energy saving in UIM piggyback dissemination and better PHR performance.

Figure 12 shows that the EPCOR schemes outperform the other three schemes in terms of average access latency. EXPRING, EPCOR and PATHCC achieve a performance improvement with increasing cache size, while the improvement of EPCOR and EXPRING is greater than that of PATHCC. This is because only few MUs participate in the cooperation for data sharing in PATHCC, as the probability of retrieving the requested data item from the nodes on the path toward BS improves slightly when the cache size increases. As shown in Figure 12, since both of PATHCC and EPCOR have opportunities to retrieve the data from other near peers, they have low access latency than NOCACHE.

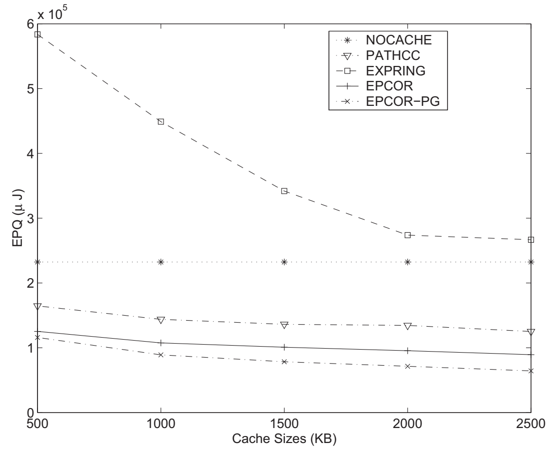

Fig. 11. Energy per Query (EPQ) vs Cache Size

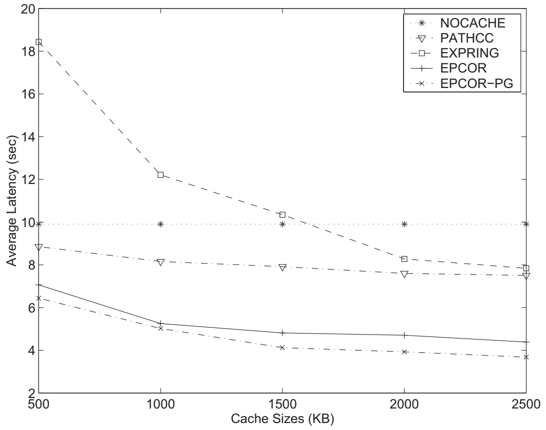

Fig. 12. Average Access Latency vs Cache Size

(2) Effect of Query Rate

In this simulation, we change the average query interval of each peer to investigate the scalability of each scheme. The cumulative data traffic of base station in 4000 seconds is measured as the performance metric. Figure 13 shows cumulative BS data traffic during 4000 seconds simulation under different query rates. In the experiment, we use two different waiting intervals (5 seconds and 10 seconds) for expanding ring flooding scheme (EXPRING) to evaluate its affect on the performances of bandwidth consumption and network throughput. EXPRING-5 and EXPRING-10 represent the scheme with 5 and 10 seconds waiting intervals, respectively.

As shown in Figure 13, NOCACHE has high BS traffic at low and moderate query rates (i.e., 1/40-1/20 sec^{-1}). This is because all requested data items are retrieved from the BS in this scheme. EXPRING and EPCOR have similar BS

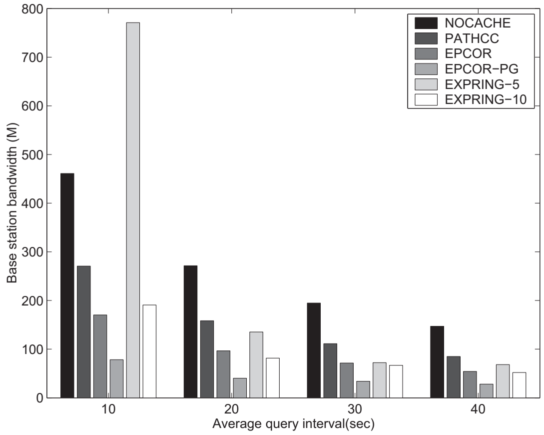

Fig. 13. BS Bandwidth Consumption vs Query Rate

data traffic under low and moderate query rates. Because in EXPRING and EPCOR schemes mobile peers retrieve the most requested items from other peers so that the traffic is distributed in the whole network, which relieves the traffic burden of the BS. However, when the query rate is high (i.e., 1/10 sec^{-1}), the BS data traffic of EXPRING-5 dramatically increases. This is because EXPRING-5 needs to flood its request message every 5 seconds if the requested data item can not be located. Under high query rate, the flooding messages quickly exhaust the network bandwidth. Most nodes fail to retrieve the items from other peers, so that the nodes have to increase the flooding range until they reach the base station. EXPRING-10 reduces the flooding traffic by using longer waiting interval (10 seconds), which reduces the traffic congestion at BS. The P2P overlay of EPCOR enhances peer to peer data retrieval among all nodes, and the dynamic cooperation radius setting reduces the UIM dissemination traffic (e.g., most nodes around the BS do not generate UIM traffic) Thus, EPCOR has the least BS bandwidth consumption under different query rates. If we consider the bandwidth of BS as the bottleneck of the throughput of the entire network, EPCOR improves the network throughput by efficiently enhancing the peer-to-peer data retrieval among all nodes in the network. In the enhanced EPCOR scheme, since piggyback mechanism is used to disseminate the UIM messages, the network traffic is significantly reduced. Figure 14 shows the access throughput of six schemes under different query rates. We measure the total number of successful data queries in the given simulation period to indicate the throughput performance of these schemes. As shown in the figure EPCOR has the best throughput among all schemes under different query rates. Due to the network congestion, the expanding ring scheme suffers a low throughput at the high query rate.

(3) Effect of Data Update Rate

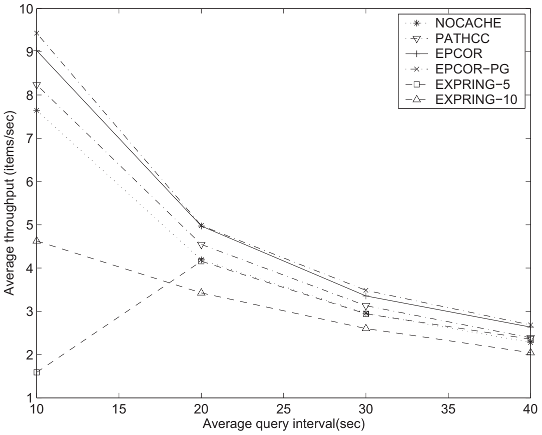

Fig. 14. Average Throughput vs Query Rate

In this experiment, we evaluate the performance of EPCOR under different data update rates. Figure 15 shows the performance of HR and PHR of five schemes. As the update rate increases, both HR and PHR metrics decrease. This is because fewer and fewer valid data items are cached in each node to serve local queries and those from other peers.

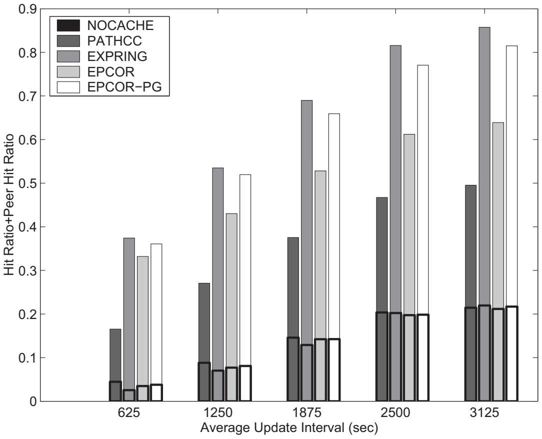

Fig. 15. Hit Ratio and Peer Hit Ratio vs Data Update Rate

Figures 16 and 17 show the access latency and EPQ performances of four schemes under different data update rates. Figure 16 shows that the access latencies of EXPRING, PATHCC and EPCOR increase as the data update rate increases. PATHCC and EPCOR have similar access latency as NOCACHE at high data update rate, and EXPRING suffers long access latency at high data update rates. When data items are updated frequently, few valid items are available at the cache of each node and therefore most queries are forwarded to the BS to retrieve the data items. In the EXPRING scheme, if one query fails, the node needs to increase the broadcast hops for the next query. Therefore, when update rates are high, most queries need to reach BS to retrieve the requested item, leading to a broadcast storm in the network. Thus, EXPRING has a long access latency and high energy consumption at high data update rates.

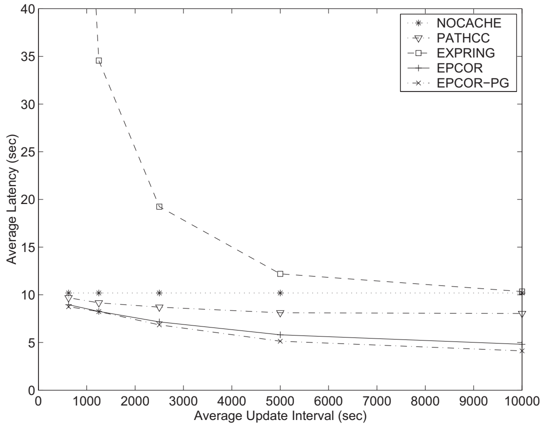

Fig. 16. Average Access Latency vs Data Update Rate

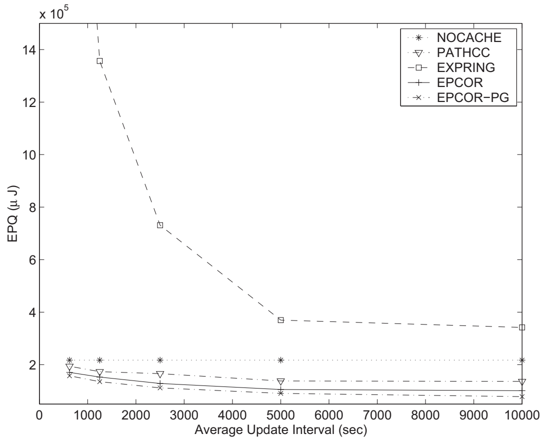

Fig. 17. Energy per Query vs Data Update Rate

(4) Load Balance and Effect of Mobility

In order to investigate the load balance in each scheme, we run the simulation for 4000 seconds and record the energy consumption of each MU under three different mobility scenarios: $v_{max} = 2m/s$, $10m/s$ and $20m/s$. Because the energy cost of each MU in the simulation period indicates the amount of data traffic that the MU has processed, we calculate the standard deviation of energy consumption of all MUs at the end of the experiment to measure the load balance of each scheme. Figure 18 shows the energy consumption standard deviation (ECSD) of all MUs for each scheme after 4000 seconds of simulation.

It is shown in the figure that each scheme has a high ECSD performance at low mobility scenario (i.e., $v_{max} = 2m/s$), and ECSD drops as the mobility increases. Under a low mobility scenario, the MUs that are close to the BS

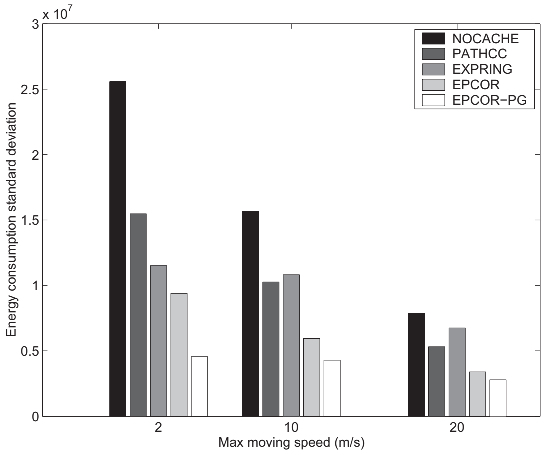

Fig. 18. Energy Consumption Standard Deviation vs Mobility

need to relay data traffic between the BS and other MUs. Therefore, the MUs around the BS have a much higher energy consumption than the MUs that are far away from the BS. This results in a poor load balance in hybrid networks. As shown in Figure 18, NOCACHE scheme has a high ECSD at low mobility. This is because no cache is deployed at each MU in NOCACHE and all queries are forwarded to the BS to retrieve data items. Thus the MUs closer to the BS consume more energy due to the frequent data relaying, leading to high ECSD performance. Since the battery life of the MUs that relay data traffic to BS is the indicator of the life time of the entire network, NOCACHE has a short life time due to the poor load balance. Figure 18 shows that EPCOR outperforms the other three schemes under different mobilities. This is because the P2P overlay of EPCOR uses a cooperation zone to share each data item among all peers, and the radius of cooperation zone varies according to the distance from BS. The data items of far-off MUs have a bigger cooperation radius than that of nearby MU. Thus, the data traffic is equally distributed in the whole network, and EPCOR significantly improves the load balance performance of hybrid networks. Enhanced EPCOR maintains bigger cooperation zone of each data item due to the low cost of UIM piggyback dissemination, hence it improves load balance among all nodes in the network. As the mobility increases, ECSD of each scheme drops. In a high mobility scenario, the average moving speed of each MU is high so that each MU has equal opportunity to appear at different locations of the hybrid networks. Thus, the energy consumption of nodes is similar to one another as the network topology changes fast.

VII. CONCLUSION

Inheriting positive features of infrastructure-based and mobile ad hoc networks, the hybrid wireless networks are expected to become effective and popular in providing low cost mobile data services with high reliability. Poor load balance, high latency and high energy consumption are among the major constraints for data access in such networks. In this paper, we proposed a novel caching scheme, called *energy efficient peer-to-peer caching with optimal radius* (EPCOR) to reduce latency and energy consumption associated with data accesses in such networks. In this scheme, a peer-to-peer (P2P) overlay is built based on network proximity and data preference to enhance peer-to-peer communication and load balancing among all mobile users. In the P2P overlay, each MU shares a data item in a cooperation zone by proactive dissemination of cache index information. An analytical model is developed to evaluate the performance of EPCOR, and an algorithm is developed from the model to calculate the optimal radius of the cooperation zone of each data item based on energy efficiency. Both the analytical and experimental results demonstrate that EPCOR outperforms existing approaches in terms of energy saving, and also shows significant improvements in terms of network throughput, access latency and load balance.

The proposed cooperative caching scheme is leading to further research work in several areas related to data management in mobile computing environments. Since the cache index information of neighboring peers is available at the resource table, it can help peers improve the performance of cache replacement algorithm to reduce the data redundancy in networks. To study the effect of continued and rapid node failure in the localized overlay on the performance of EPCOR, is a part of our future work.

REFERENCES

[1] L. Breslau, P. Cao, J. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-Like Distributions: Evidence and Implications," *In IEEE Proceedings of INFOCOM*, pp 126-134, 1999.

[2] P. Barford and M. Crovella,"Generating Representative Web Workloads for Network and Server Performance Evaluation," *Proceedings of the ACM SIGMETRICS Conference*, pp. 151-160,2005.

[3] J. Broch, D. Maltz, D. Johnson, Y. Hu, and J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," *ACM MobiCom*, pp. 85-97, October 1998.

[4] H. Che, Y. Tung, and Z. Wang,"Hierarchical Web Caching Systems: Modeling, Design, and Experimental Results," *IEEE Journal on Selected Areas in Communication*, 20(7), pp 1305-1315, 2004.

[5] L. Fan, P. Cao, J Almeida, and A. Broder, "Summary Cache: A Scalable Wide Area Web Cache Sharing Protocol," *ACM SIGCOMM*, pp. 254-265,2001.

[6] L. Feeney, " An energy consumption model for performance analysis of routing protocols for mobile ad hoc networks. " *In ACM Journal on Mobile Networks and Applications*, 6(3), pp. 239-249, 2004.

[7] L. Feeney, and M. Nilsson, "Investigating the Energy Consumption of a Wireless Network Interface in an Ad hoc Networking Environment," *In IEEE Proceedings Of INFOCOM*, 2001.

[8] Z. J. Haas and M.R. Pearlman,"The performance of query control schemes for the zone routing protocol," *ACM/IEEE Thransactions on Networking*, 9(4), pp. 427-438,2001.

[9] T. Hara, "Effective Replica Allocation in Ad Hoc Networks for Improving Data Accessibility," *In IEEE INFOCOM*, 2005.

[10] H-Y. Hsieh and R. Sivakumar, "On Using Peer-to-Peer Communication in Cellular Wireless Data Networks," *IEEE Tran. on Mobile Computing*, 3(1), pp. 57-72, Jan-Mar 2006.

[11] S. Iyer, A. Rowstron, and P. Druschel, "Squirrel: A Decentralized Peer-to-peer Web Cache" *Proc. 21st Annual ACM Symposium on Principles of Distributed Computing (PODC)*, pp. 203-212, July 2006.

[12] R. Karrer, A. Sabharwal, and E. Knightly, "Enabling Large Scale Wireless Broadband: The Case for TAPs" *Proc. of HotNets*, 2003.

[13] U. Kubach, and K. Rothermel, "Exploiting Location Information for Infostation-Based Hoarding" *In ACM Intl. Conf. on Computing and Networking (Mobicom)*, pp. 15-27, 2003.

[14] W. H. O. Lau, M. Kumar, and S. Venkatesh, "A Cooperative Cache Architecture in Supporting Caching Multimedia Objects in MANETs," *The Fifth International Workshop on Wireless Mobile Multimedia*, pp. 56-63, September 2004.

[15] B. Li, M.J. Golin, G. F. Iaiano, and X. Deng, "On the Optimal Placement of Web Proxies in the Internet," *In IEEE Proceedings of INFOCOM*, pp. 1282-1290, March 1999.

[16] S. Lim, W. C. Lee, G. Cao, and C. R. Das, "A Novel Caching Scheme for Internet based Mobile Ad Hoc Networks," *In IEEE Int'l Conf. on Computer Comminications and Networks (ICCCN)*, pp.38-43, October 2003.

[17] M. Lindsey, M. Papadopouli, F. Chinchila and A. Sing,"Measurement and analysis of the spatial locality of wireless information and mobility patterns in a campus," *Techical Report TR03-006, Department of Computer Science, The University of North Carolina at Chapel Hill*, March 2003.

[18] B. Liu, Z. Liu, and D. Towsley, "On the Capacity of Hyrbid Wireless Networks," *In IEEE INFOCOM*, pp. 1543- 1552, March 2003.

[19] Y. D. Lin and Y.C. Hsu, "Multihop Cellular: A New Architecture for Wireless Communications" *In IEEE Proceedings of INFOCOM*, pp. 1273-1282, March 2000.

[20] P. Linga, I. Gupta, and K. Birman, "A Churn-Resistant Peer-to-peer Web Caching System" *ACM Workshop on Survivable and Self-Regenerative Systems*, pp. 1-10, October 2003.

[21] H. Luo, R. Ramjee, P. Sinha, L. Li and S. Lu "UCAN: A Unifie Cellular and Ad-Hoc Network Architecture" *In ACM Intl. Conf. on Computing and Networking (Mobicom)*, pp. 353-367, 2003.

[22] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker "Search and replication in unstructured peer-to-peer networks," *In Proceedings of the 16th annual ACM International Conference on Supercomputing*, pp. 84 - 95, June 2002.

[23] P. Nuggehalli, V. Srinivasan, and C. F. Chiasserini "Energy-Efficien Caching Strategies in Ad Hoc Wireless Networks" *In Proceedings of ACM MobiHoc*, pp. 25-34, 2003.

[24] M. Papadopouli, and H. Schulzrinne "Effects of power conservation, wireless coverage and cooperation on data dissemination among mobile devices" *In Proceedings of ACM MobiHoc*, pp. 117-127, 2001.

[25] C. Perkins, and P. Bhagwat,"Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for mobile computers," *ACM SIGCOMM*, pp 234-244, 1994.

[26] C. Perkins, and E.M. Royer,"Ad-hoc On-demand Distance Vector routing," *Proc. of WMCSA 1999*, pp 90-100, 1999.

[27] G.J. Pottie and W.J. Kaiser, "Wireless Integrated Network Sensor," *In Communications of the ACM*, 43(5), pp. 551-558, May 2000.

[28] A. Rowstron, and P. Druschel,"Pastry: Scalable, Decentralized Object Location and Routing for Large-Scale Peer-to-Peer Systems", *Proc. of IFIP/ACM Intl. Conf. on Distributed Systems Platforms (Middleware 2001)*, pp. 329-350. Heidelberg, Germany, November 2001.

[29] A. Rousskov and H. Schulzrinne, "Cache Digests," *Computer Networks and ISDN Systems*, pp. 22-23, 1998.

[30] T. S. Rappaport, "Wireless communications, principles and practice, ", *Prentice Hall*, 1996.

[31] F. Sailhan and V. Issarny, "Cooperative Caching in Ad Hoc Networks" *In Intl. Conf. on Mobile Data Management (MDM)*, pp. 13-28, 2003.

[32] H. Shen, S. K. Das, M. Kumar and Z. Wang, "Cooperative Caching with Optimal Radius in Hybrid Wireless Networks," *Proceedings of Third International IFIP-TC6 Networking Conference (Networking 2004)*, LNCS, Vol. 3042, pp. 841-853, Athens, Greece, May 2004.

[33] J. Shim, P. Scheuermann and R. Vingralek, "Proxy Cache Design: Algorithms, Implementation and Performance," *IEEE Trans. On Knowledge and Data Engineering(TKDE)*, 11(4), pp. 549-562, July/August 1999.

[34] X. Y. Wang, W. S. Ng, B. C. Ooi, K. L. Tan, and A. Y. Zhou, "BuddyWeb: a P2P-based Collaborative Web Caching System" *Proc. International Workshop on Peer-to-Peer Computing*, pp. 247-251, 2002.

[35] D. Wssels, and K. Claffy, "ICP and the Squid Web Cache," *IEEE Journal on Selected Areas in Communication*, pp 345-375, 1998.

[36] H. Wu, C. Qiao, S. De, and O. Tonguz, "Intergrated Cellular and Ad Hoc Relay systems: iCAR" *IEEE Journal on Selected Areas in Communications (JSAC)*, 19(10), pp. 2105-2115, Oct. 2001.

[37] L. Yin, and G. Cao, "Supporting Cooperative Caching in Ad Hoc Networks," *In IEEE Proceedings of INFOCOM*, pp. 2537-2547, March 2004.

[38] IEEE 802.11 Working Group, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications 1999.

[39] http://www.pdos.lcs.mit.edu/roofnet.

[40] ns Notes and Documentation, "http://www.isi.edu/nsnam/ns/".