CAPS: A Peer Data Sharing System for Load Mitigation in Cellular Data Networks

Young-Bae Ko
Ajou University
Suwon, South Korea.
youngko@ajou.ac.kr

Kang-won Lee
IBM T. J. Watson Research Center
Hawthorne, NY 10532. USA.
kangwon@us.ibm.com

Thyaga Nandagopal
Bell Labs, Lucent Technologies
Holmdel, NJ 07748. USA.
thyaga@lucent.com

Abstract— **The exponential growth of mobile data users and services places a heavy burden on the limited wireless bandwidth of cellular data networks. The situation will be exacerbated with the advent of high bandwidth multimedia applications for mobile devices. In this paper, we propose an architecture called the *Cellular-based Ad hoc Peer Data Sharing system (CAPS)* to reduce the load on the cellular network while improving request response times. In CAPS, mobile hosts in a cellular network form an overlay multi-hop wireless network. This ad hoc network acts as a 'virtual cache' that enables data sharing among mobile hosts. Participating mobiles share the contents of their local caches with other mobiles. A subset of mobile hosts keep track of the location of objects with minimal overhead. Using CAPS, popular objects can be obtained over the ad hoc network without accessing the cellular infrastructure, thereby reducing the load on the cellular network and improving data access latency. We have extensively evaluated the performance of this architecture through simulations in *ns-2* across a wide range of scenarios, and find that CAPS reduces the timeouts of the user requests by 70 – 90% and reduces the load at the base station by up to 60%.**

I. Introduction

The popularity and growth of wireless data has led to various wireless technologies, ranging from wide-area cellular data networks (e.g. GPRS and 3G/4G networks) to decentralized ad hoc networks, and wireless LANs based on IEEE 802.11 [1] and HIPERLAN [2] standards. The average data rates supported by these systems vary widely, from 100 Kbps for GPRS to up to 54 Mbps for wireless LANs. The transmission range also varies widely depending on the technology by a couple of orders of magnitude. This diversity is a fundamental characteristic of the future wireless networks because no single system is best suited for all environments. Consequently, mobile users are likely to access data using multiple wireless interfaces, e.g., using an IEEE 802.11 network at home and at work, and a 3G/4G cellular data network while on the move.

In the emerging wireless architecture, access to the Internet backbone is controlled either by some form of a cellular network or by wireless local area networks (WLANs) called 'WLAN hotspots.' Since it is not possible to have ubiquitous coverage through WLAN hotspots, mobile users requiring continuous connectivity will have to connect through the cellular data network. With the exponential growth in wireless data users and the projected linear-scale growth in cellular data bandwidth, the cellular channel is expected to become overloaded and consequently data access latency may increase without bound. This situation will be further exacerbated with the introduction of multimedia applications on mobile devices.

In this paper, we propose a novel solution to alleviate the wide-area wireless bandwidth scarcity problem by integrating ad hoc networks within a cellular data network architecture. The key idea behind the proposed scheme, that we call the *Cellular-based Ad hoc Peer Data Sharing system* (CAPS), is to establish an overlay multi-hop wireless network among mobile hosts on a separate channel and then try to retrieve popular objects from the local cache of peer mobiles, using a distributed directory. When the requested object cannot be obtained from a peer mobile, the request is sent over the backbone via the cellular channel. Thus, peer mobiles participating in CAPS can be collectively viewed as a 'virtual proxy cache' or a 'file sharing system.'

The CAPS system consists of simple low overhead protocols for establishing the ad hoc peer network and for efficient data access among peer hosts in the ad hoc network, based on distributed object location information. We expect that CAPS will reduce the load on the cellular system, thereby reducing the object download time observed by wireless users.

The rest of the paper is organized as follows. Section II describes related work, and Section III presents the network service model considered in this paper. Section IV provides an overview of CAPS and Section V quantifies the performance of CAPS through simulations. Finally, we conclude the paper in Section VI.

II. Related Work

There exists a wide body of related work aiming to utilize ad hoc networks to augment some facet of cellular systems. The Multi-hop Cellular Network (MCN) system [3] effectively extends the reachability of a cellular network by transmitting data over multi-hop paths consisting of mobile devices. In iCAR [4], special ad-hoc relay stations are deployed to relay traffic from a congested cell through a neighboring uncongested cell. Unlike the related work where the ad hoc network is used for extended reachability, CAPS uses the ad hoc network to alleviate the bandwidth scarcity problem.

In [5], the authors proposed a data sharing system for mobile devices, called 7DS. The 7DS relies on data exchange between directly reachable hosts. 7DS is similar to CAPS in that both of them provide mechanisms to enable peer data sharing. However, CAPS allows data retrieval from a peer that is not directly reachable while a 7DS mobile can only talk to

 0-7803-7974-8/03/$17.00

peers within the range. Also, the main emphasis of CAPS is on the throughput enhancement to the cellular network, while 7DS aims to provide extended data availability when the infrastructural support is not available.

On a different note, cellular service providers are investigating techniques to provide seamless roaming between Wi-Fi networks and cellular data networks The Wireless Internet Service Provider Roaming (WISPr), a subcommittee of the Wireless Ethernet Compatibility Alliance [6], has been investigating this issue. The focus is on establishing an unified billing and location tracking standard across these heterogenous networks. To the best of our knowledge, there has been no prior work relating to the use of ad hoc networks to supplement data services for load mitigation in cellular networks.

Recently, a lot of attention has been drawn to peer-to-peer data sharing systems in the Internet, and new architectures [7], [8] have been proposed. In such systems, the object location information is organized into a distributed hash structure using a sophisticated algorithm to optimize the search cost. However, such distributed location management may not be viable in a highly dynamic environment as in mobile ad hoc networks where the group of participants and the network topology can change continuously in a relatively small time scale. In this context, we strive to provide a simpler yet more practical solution to the peer data sharing problem in ad hoc networks.

III. Network Model

Cellular data network: We consider a macro cellular network environment with multiple channels per cell. The channel is slotted in time for uplink and downlink communications as in GSM and GPRS. Mobile hosts access the backbone through a local base station. The base station communicates with mobile hosts using a control channel and data channels. The control channel is used by hosts to transmit requests for sending data, and by the base station to announce the schedules. The base station can also use control channel to coordinate CAPS operation such as directory node election if it supports CAPS (see Section IV-C).

Ad hoc network: We assume each mobile host has at least two wireless interfaces, of which one interface is used to communicate with the cellular base station, and the other interface is used to communicate with other mobile users for seamless mobility. We assume that if a mobile host can reach a peer through the ad hoc network, then there exist suitable protocols to handle connection and session management. For our simulations, we used TCP/IP over the DSDV routing protocol [9] with 802.11 MAC. The maximum data rate of the ad hoc network is typically higher than that of the cellular network, and the ad hoc data range is shorter than that of the cellular network. For example, the peak data rate is up to 11 Mbps for 802.11b covering about 250 meters [1], compared to the data rates of 400 Kbps outdoors and 2 Mbps indoors for 3G cellular networks, whose cell radius is typically around 1 km. The operational frequency of the ad hoc network is separate from that of the cellular network hence there is no interference, e.g., GPRS operates in 900 MHz band whereas 802.11 operates in either 2.4 GHz or 5 GHz range.

IV. Architecture Overview

The main idea of CAPS is to locate and retrieve popular objects from peer mobile hosts via ad hoc communication whenever possible, instead of downloading them from the backbone network through the base station. If a substantial amount of data requests can be served by ad hoc peer-to-peer (P2P) operation, it will reduce congestion at the base station, and subsequently lower the response time observed by the end users.

A. Ad hoc object discovery

When designing a peer data sharing over ad hoc networks, we must address the fundamental challenge posed by the peer-to-peer principal, namely "How to determine the location of a target object?" One simple solution is to initiate a scoped broadcast of object location query. Whoever has the knowledge of the object location can respond to the query. While this approach is easily implementable with no overhead of maintaining object location, it may introduce a large volume of control traffic, and therefore may not scale well.

Another way to address this problem can be using active snooping: Each mobile host actively snoops downlink communications to learn of object delivery information. In theory, mobiles can snoop the downlink communication by listening to the cellular channel. However, in practice active snooping cannot be readily facilitated due to technical and administrative constraints. Even if active snooping is allowed, we find that energy consumption is excessively high for battery-operated mobile devices (Section V).

The approach taken by CAPS tries to make the best out of the above schemes. In CAPS architecture, only a small set of mobiles that have enough battery power actively maintains the object location information. These mobiles are called the *directory nodes* (or *dnodes*). The id's of the dnodes are well-known and when a non-directory mobile wants to find an object from a peer, it simply contacts one of the dnodes to get location information.

As previously mentioned, CAPS can operate with and without support from the cellular service provider. When the cellular network is oblivious of CAPS, each mobile voluntarily reports its local cache entry to a neighboring dnode. In addition, dnodes passively snoop the ad hoc traffic to discover object location information similarly to route snooping in some ad hoc routing protocols [9].

When the cellular network is cooperative, base station periodically sends down the summary of object transmission during the last interval on a broadcast channel. This information is used to update the location directory of the dnodes. Additionally, dnodes can snoop ongoing transmission on the ad hoc channel. In the remainder of this paper, we describe the CAPS architecture with an assumption that the cellular service provider is cooperative.

B. Electing directory nodes

The dnode needs to listen to the location information from the base station, snoop traffic on the ad hoc channel, and serve requests for object information in addition to its normal functions as a mobile host. Consequently, dnodes expend more energy than non-dnodes. In CAPS, new dnodes are elected periodically to avoid depleting the battery power of particular hosts by running a directory election algorithm among hosts.

To reduce the latency to locate target objects, every mobile host must be able to access a dnode with the lowest possible delay. To ensure this, we can try to identify dnodes in the cell such that every mobile host has at least one dnode in its neighborhood. In CAPS, this goal is achieved by a distributed algorithm to find a dominating set[1] in the connectivity graph of the ad hoc network of mobile hosts.[2] A detailed dnode election algorithm is presented in the extended version of the paper [11].

C. Directory node operation

The dnode performs three main functions: (1) obtain location information from the base station, (2) serve object location information to requesting peer hosts, and (3) snoop ad hoc communication to update object location information. To maintain a location directory of the objects, each dnode constructs a simple hash table: {`object ID` → list of (`mobile address, download time`)}, and responds to a location query with the corresponding (host address, download time) entries. Upon reception of the response from the dnode, the mobile can choose to contact any peer node from the entry. In our scheme, the mobile host first tries to contact a peer node in its wireless range. If it fails, then it tries to contact a peer node that has cached the object most recently.

D. Object location directory data structure

Each dnode maintains the location information using the MD5 digest values of the URLs. In this way, we can store and look up cache entries using the 128 bit digest values instead of variable length URL strings. In the location directory, each hash entry consists of a linked list of the location data structure. Each location data structure consists of three components: (a) the IP address of the mobile host that stores the object (4 bytes), (b) the object download time represented in seconds (2 bytes), and (c) a checksum of URL (2 bytes). With the 16 bit 'time' field, the entry does not overlap within 18 hours, which is a large enough period for our purpose. The URL checksum calculates the 16 bit one's complement of the digest value to resolve potential collisions in the MD5 hash. Using this data structure, 10,000 location data structure can be stored in about 100KB of memory. Note that for a single object there can be more than one location data structure if the object has been downloaded by multiple mobile hosts.

[1] Given a graph $G = (V, E)$ consisting of a set of vertices V and a set of edges E, a dominating set $S \subset V$ is a set such that every node in V is either in S or is a neighbor of a node in S. A dominating set with minimum cardinality is called a minimum dominating set. Finding a minimum dominating set is an NP-complete problem [10].

[2] The connectivity graph for the ad hoc network is vastly different from the connectivity graph of the cellular network, since the radio range of these two interfaces vary.

E. Updating location directory

As described in the previous section, dnodes periodically receive object location information from the base station. In addition, dnodes can snoop the communications on the ad hoc channel to discover up-to-date location information. For instance, if a dnode overhears that mobile host A is downloading object x from mobile host B, then it adds a directory entry specifying that mobile A has cached object x. In addition, when a dnode receives a location query for object y from mobile host C, then it can infer that mobile host C will cache object y without overhearing the actual transmission.

It is evident that these partial updates may create inconsistent view among dnodes, since an ad hoc transmission overheard by some dnode is not necessarily heard by other dnodes. Therefore, dnodes need to periodically exchange messages to update each other with local changes. For this purpose, each dnode send out the incremental changes that it made in the last time period T_{update}. In this way, the location information of popular objects, that are cached by a large number of mobile hosts in the cell, gets propagated to all dnodes.

F. Handing off location directory

To ensure non-disruptive services when a new set of dnodes commences in a new period, the retiring dnodes must transfer their location directory to the new dnodes. To reduce the control traffic during this *hand-off* phase, the predecessor transfers the location directory entries for the most recently accessed objects. In other words, they transfer only the directory entries that has been created within the last $T_{history}$ seconds, where $T_{history}$ is a period during which the location information is considered valid. Ideally, the time period $T_{history}$ must be adaptive to the dynamics of the ad hoc network: if the ad hoc network is highly dynamic then the value of $T_{history}$ must be small since the corresponding mobile host may have left the ad hoc network already. In practice, however, it is hard to determine the dynamics of ad hoc networks. Hence, we employ a conservative value of $T_{history}$, estimated from a long term measurement.

V. PERFORMANCE EVALUATION

In this section, we evaluate the proposed scheme via a simulation study using the network simulator, *ns 2*. We extended the ns simulator to implement the Web access capability on the mobile nodes, ad hoc mode operation among peers including directory query and response, and dnode management scheme.

A. Simulation Model

In our simulation, initial locations (X and Y coordinates) of the mobile nodes are set using a uniform distribution. We test with 10 – 40 mobiles, moving around in a rectangular region of size 1000 m x 1000 m according to the random waypoint mobility model. The mobility of the random waypoint model is determined by the maximum speed of the mobiles and

parameter	*range*	*default value*
arrival rate (req/sec)	0.2 – 0.6	0.4
max speed (m/sec)	0 – 20	5
number of mobile hosts	10 – 40	10
Zipf parameter	0.6 – 1.4	1.0

TABLE I

RANGES OF THE SIMULATION PARAMETERS AND THEIR DEFAULT VALUES

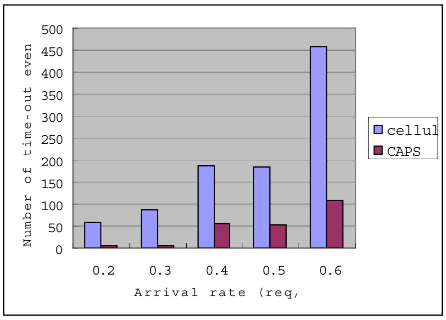

Fig. 1. Impact of the request arrival rate on the latency to download objects

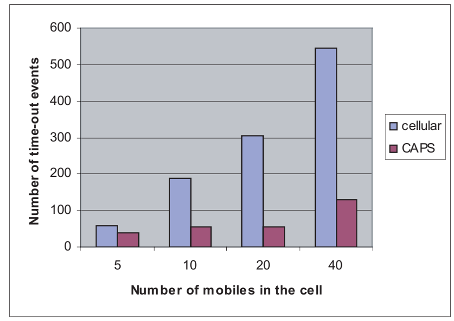

Fig. 2. Impact of the number of mobile hosts in a cell on the latency to download objects

	Tx	Rx
Ad hoc	1.6 * S / 2e06	1.2 * S / 2e06
Cellular	1.6 * S / 4e05	1.2 * S / 4e05

TABLE II

ENERGY CONSUMPTION MODEL WHERE S REPRESENTS DATA SIZE IN BITS

the pause time after reaching the position. The *maximum* speeds of the nodes were set to 1 – 20 m/sec with various pause time values. In general, we find that mobility does not fundamentally change the performance characteristics of CAPS [11]. Thus we present the case with maximum speed of 10 m/sec and 0 sec pause time. The wireless link bandwidth between mobile hosts is set to 2 Mbps to model WLAN speed. The cellular link bandwidth is set to 400 Kbps modeling the 3G wireless networks. The wired connection between the base station and the origin server is set to 10 Mbps with 50 ms delay.

To model the user access pattern, we use ProWGen synthetic trace generator [12] to generate Web access traces. ProWGen effectively models key parameters that determine the user behavior, e.g., object popularity, object size distribution, correlation between object size and popularity, and temporal locality. In our traces, the object size distribution follows log-normal distribution with the mean of 7 KB to simulate the Web environment.

B. Simulation Results

We now present the performance of CAPS in comparison to the basic cellular network under various simulation scenarios. For the performance metric, we use the number of time-out events of user requests since it directly translates into the quality of Web access that user experiences. We call a request has been *timed out*, when the desired object could not be downloaded within a certain time value $t_{timeout}$. In our simulation, we set $t_{timeout} = 100$ seconds.

1) User request arrival rate: We first consider how each scheme performs under various load conditions by changing the user request arrival rate. Figure 1 presents the number of timeout events with respect to the request arrival rate. From the figure we observe that CAPS reduces the number of timeout events significantly. In particular we observe 70 - 90 % reduction in timeouts with CAPS compared to the basic cellular case.

The performance improvement of CAPS results from 'cache hits' in other mobile hosts' caches on the ad hoc network. From the trace log, we find that about 40% of the requests have been served by the local cache in both cases across all load conditions. In addition to the local hits, about 35% of the requests have been served in ad hoc mode without consuming cellular bandwidth in CAPS. Since mobile hosts always try to download using the ad mode first, these objects are never sent on the cellular network. This means that CAPS effectively reduces the traffic on the cellular network by about 60%.

2) Number of mobile hosts: In this section, we study the impact of the population of mobile hosts in a cell on the performance of the cellular network and CAPS. Figure 2 presents the results with varying number of mobile hosts in the cell. As the number of mobiles increases in a cell, so does the level of contention. Contentions on the wireless channels result in packet loss and link level retransmission which subsequently affect the end user performance in an adverse manner. From the figure, we observe that CAPS is highly effective in maintaining the level of contention low. In particular, the number of timeouts increases linearly in the case of the cellular network. With CAPS however the increase is much more graceful and therefore allows better scalability.

3) Energy consumption: Battery power is a limited resource on the mobile devices and has to be managed efficiently. In this section, we summarize the implication of peer data sharing on the energy consumption at mobile hosts. We employ the simple energy consumption model implemented in *ns* (see Table II).

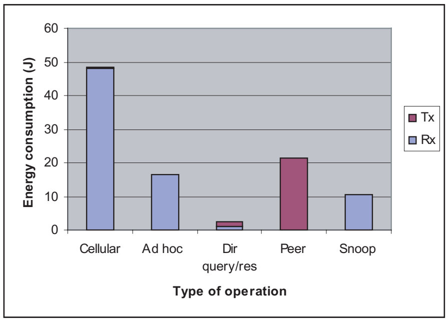

Fig. 3. Energy consumption on various types of operation

Figure 3 presents the energy consumption for various types of operations at the dnode. The y-axis represents the consumed energy in J. The x-axis represents each type of operation. The first two columns "Cellular" and "Ad hoc" represent the case of downloading objects *for its own consumption* from the cellular network and the ad hoc network, respectively. We observe that downloading objects from the cellular network is the dominant source of energy consumption expending about 50% of the total energy. The "Dir query/res" column corresponds to the operations to handle object location query and response from the other mobile hosts. Note that as we increase the number of dnodes, the energy consumption in this category will decrease correspondingly. From the figure, we observe that handling directory queries is not a major overhead for the dnode even in this worst case scenario. The "Peer" column represents the case of peer-to-peer data transfer to serve other mobile hosts. Note that non-dnode mobiles will also observe this overhead. Overall, 20% of the total energy consumption is utilized to serve the data requests from the other mobile hosts. The "Snoop" column represents the energy consumed while receiving summary information from the base station. We observe that this extra energy consumption is moderate; about 10% of total energy consumption. However, when there is no support from the base station and if the dnode has to snoop all downlink communication in a TDMA network, the energy consumption is excessively high (120 J per 5 min) prohibiting battery-operated nodes to work as dnodes.

VI. CONCLUSION

The exponential growth of mobile data users and services places a heavy burden on the limited wireless bandwidth of cellular data networks. On the other hand, the diversity in the wireless data access technology, ranging from 3G/4G wide-area cellular data network to 802.11/HIPERLAN, will likely require mobile users to equip multiple network interface for seamless always-on data access.

In this paper, we proposed an overlay peer data sharing architecture called the *Cellular-based Ad hoc Peer Data Sharing system (CAPS)* to reduce the load on the cellular network while improving request response times experienced by mobile users. In CAPS, mobile hosts in the cellular network form an overlay multi-hop wireless network. This ad hoc network acts as a 'virtual cache' or a 'file sharing system' that enables data sharing among peer mobile hosts. Participating mobiles share the contents of their local storage with other mobiles. A subset of mobile hosts keep track of the location of objects with minimal overhead. Using CAPS, popular objects can be obtained over the ad hoc network without accessing the cellular infrastructure, thereby reducing the load on the cellular network. At the same time, mobile users can enjoy peer data sharing and download without paying for air time to the wireless data service provider.

We have extensively evaluated the performance of this architecture through simulations in *ns-2* varying user load levels, population in the cell, and user access patterns. From the simulation study, we find that CAPS reduces the load on the cellular network by up to 60%, and also reduces the timeouts of the user requests by 70 – 90% on the average. We also find that about 20% of total energy is consumed for serving other mobiles in peer-to-peer manner and about 10% of total energy is consumed for maintaining object location in the cell.

In this paper, our evaluation primarily focused on the case where the cellular network is cooperative. When the cellular network is not supportive, however, we expect that CAPS can still provide performance enhancement. In such a scenario, the location of the target object has to be queried via scoped broadcast unless the information is already stored in the local cache. The range of the query should be carefully determined (usually limited to a few hops) to avoid long delays in locating objects and reduce control traffic on the ad hoc network. We plan to investigate the CAPS performance in such environment as part of our future work.

REFERENCES

[1] "IEEE P802.11, The Working Group for Wireless LANs." On-line document, Jan. 2002. http://grouper.ieee.org/groups/802/11.
[2] "HiperLAN2 Global Forum." On-line document, Mar. 2002. http://www.hiperlan2.com/.
[3] Y.-D. Lin and Y.-C. Hsu, "Multihop Cellular: A New Architecture for Wireless Communications," in *Proceedings of IEEE INFOCOM*, Mar. 2000.
[4] C. Qiao and H. Wu, "iCAR: an Intelligent Cellular and Ad-hoc Relay System," in *Proceedings of IC3N*, Oct. 2000.
[5] M. Papadopouli and H. Shulzrinne, "Effects of power conservation, wireless coverage and cooperation on data dissemination among mobile devices," in *Proceedings of ACM MobiHoc*, Oct. 2001.
[6] "Wireless Ethernet Compatibility Alliance." On-line document, May 2002. http://www.weca.net.
[7] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-Addressable Network," in *Proceedings of ACM SIGCOMM*, Aug. 2001.
[8] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan, "Chord: A Scalable Peer-To-Peer Lookup Service for Internet Applications," in *Proceedings of ACM SIGCOMM*, Aug. 2001.
[9] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Computing*, pp. 153–181, 1996.
[10] P. Crescenzi and V. Kann, "A Compendium of NP Optimization Problems." http://www.nada.kth.se/ viggo/wwwcompendium/wwwcompendium.html, March 2000.
[11] Y.-B. Ko, K.-W. Lee, and T. Nandagopal, "Caps: A peer data sharing system for load mitigation in cellular data networks," *IBM Research Report RC22468*, May 2002.
[12] M. Busari and C. Williamson, "On the Sensitivity of Web Proxy Cache Performance to Workload Characteristics," in *Proceedings of IEEE INFOCOM*, Apr. 2001.