



# Cooperative Caching in Ad Hoc Networks

Françoise Sailhan, Valérie Issarny

► **To cite this version:**

Françoise Sailhan, Valérie Issarny. Cooperative Caching in Ad Hoc Networks. 4th International Conference on Mobile Data Management : MDM 2003, 2003, Melbourne, Australia. pp.13-28. inria-00414796

**HAL Id: inria-00414796**

**<https://hal.inria.fr/inria-00414796>**

Submitted on 10 Sep 2009

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Cooperative Caching in Ad Hoc Networks

Françoise Sailhan and Valérie Issarny

Inria-Rocquencourt, Domaine de Voluceau, Rocquencourt,  
BP 105, 78153, Le Chesnay Cédex, France  
{Francoise.Sailhan, Valerie.Issarny,}@inria.fr

**Abstract.** Terminal's latency, connectivity, energy and memory are the main characteristics of today's mobile environments whose performance may be improved by caching. In this paper, we present an adaptive scheme for mobile Web data caching, which accounts for congestion of the wireless network and energy limitation of mobile terminals. Our main design objective is to minimize the energy cost of peer-to-peer communication among mobile terminals so as to allow for unexpensive Web access when a fixed access point is not available in the communication range of the mobile terminal. We propose a collaborative cache management strategy among mobile terminals interacting via an ad hoc network. We further provide evaluation of the proposed solution in terms of energy consumption on mobile devices.

## 1 Introduction

The last decade has seen the rapid convergence of two pervasive technologies: wireless communication and the Internet. The resulting mobile Internet a priori enables users to easily access information anytime, anywhere. However, we have not yet reached the point where anywhere, anytime Internet access is actually offered. This paper addresses the above issue, concentrating more specifically on Web caching in a mobile environment to allow for Web access, without requiring availability of an infrastructure in the nearby environment.

There exist two different ways of configuring a mobile network: *infrastructure-based* and *ad-hoc-based*. The former type of network structure is the most prominent, as it is in particular used in both Wireless LANs (e.g., IEEE 802.11) and global wireless networks (e.g., GSM, GPRS, UMTS). An infrastructure-based wireless network uses fixed network access points (known as *base stations*) with which mobile terminals interact for communicating, i.e., a base station forwards messages that are sent/received by mobile terminals. One limitation of the infrastructure-based configuration is that base stations constitute bottlenecks. In addition, it requires that any mobile terminal be in the communication range of a base station. These shortcomings may be solved through the deployment of a sufficient number of base stations. However, this comes at a high cost for network providers. The ad-hoc-based network structure alleviates this problem by enabling mobile terminals to cooperatively form a dynamic network without any pre-existing infrastructure. In general, ad hoc and infrastructure-based networking should be seen as complementary rather than as competitive. Ad hoc

networking is much convenient for accessing information available in the local area, and possibly reaching a WLAN base station, which comes at no cost for users. Ultimately, the user may decide to pay for communication using wireless global networking facility. This paper concentrates on improving the Web latency using a WLAN, exploiting both the ad hoc and infrastructure-based capabilities of the network.

The issue that we are addressing is on setting an ad hoc network of mobile terminals that cooperate to exchange Web pages. One application of our work is, e.g., multimedia guided tour from PDAs. Indeed, several cities and museums have proposed such services over the last few years but the limited number of base stations due to a prohibitive financial cost and the limited communication range of mobile terminals prevent tourists from accessing data anywhere. In addition, the limited memory capacity of PDAs requires constant interaction with the base station to gain access to the rich set of available information. Our proposal allows users to access information when a base station is not reachable in one hop and then offers better scalability at lower financial cost.

In the above context, it is crucial to account for the specifics of mobile terminals. Mobile terminals that will soon be available will embed powerful hardware (e.g., LCD screens, accelerated 3D graphics, high performance processor), and an increasing number of devices (e.g., DVD, CD) [13]. However, the capacity of batteries goes up slowly and all these powerful components reduce battery life. Thus, it is mandatory to devise adequate solutions to energy saving on the mobile terminals, for all the constituents of the mobile environment, i.e., application software, network operating system, and hardware. In particular, communication is one of the major sources of energy consumption, and hence a number of wireless communication protocols have been designed to reduce energy consumption [12, 5, 18]. However, most of the communication protocols that have been proposed for ad hoc networks are addressed in terms of bandwidth usage and not energy consumption [9]. In addition, it is mandatory for these protocols to be coupled with distributed application software that are designed so as to minimize energy consumption.

This paper introduces such a distributed application software, which implements ad hoc cooperative Web caching among mobile terminals<sup>1</sup>. The proposed solution aims at improving the Web latency on mobile terminals while optimizing associated energy consumption. The solution accounts for both the capacities of mobile terminals and the network features; it comprises: (i) a cooperative caching protocol among mobile terminals that builds upon an existing ad hoc routing protocol, and (ii) a local caching strategy for the mobile terminal. The next section discusses related work, addressing background in the area of cooperative Web caching and Web access from mobile terminals. Section 3 then introduces the proposed ad hoc cooperative Web caching protocol, and is followed in Section 4 by the presentation of the caching strategy implemented on terminals. Section 5 provides an evaluation of our proposal, giving the energy

---

<sup>1</sup> In the following we do not address the issue of Web Data conversion for these devices for which a number of base solutions exists.

consumption associated with cooperative caching. Finally, Section 6 concludes with a summary of our contribution, and our current and future work.

## 2 Related Work

The ever growing popularity of the Web and the resulting poor latency for users have given rise to huge effort on improving the Web latency, which mainly lies in the introduction of Web caching protocols. Due to the network topology, the idea of making network caches (also referred to as *proxy caches* or simply *proxies*) cooperate has emerged. The Harvest project [2] introduced the notion of hierarchical caching, which is pioneering and lies in introducing a cache on every network node, the system's hierarchical structure coming from the national networks' hierarchical organization. A cache then locates a missing requested object by issuing a request to the cache at the hierarchy's upper level. The process is iterated until either the object is found or the root cache is reached, which may ultimately involve contacting the object's server. The object is then copied in all the caches contacted as it is returned to the client.

A transversal system enriches the hierarchy by integrating a set of sibling caches that are close in terms of latency time, at each hierarchical level. On a cache miss, a cache not only contacts its ancestral cache but also its siblings. A number of transversal cooperative caching protocols have been proposed in order to minimize the number of messages that are exchanged among sibling caches to retrieve cached objects. Proposed solutions all amount to maintain a partial knowledge of the objects that are cached on siblings. With Summary Cache [8], each proxy keeps a compact summary of the cache content of other sibling proxies. The Scoop protocol [17] proposes to distribute this knowledge among the cache composing the transversal system and the Cache Array Routing Protocol (CARP) [22] partitions the URL space among sibling proxies.

In the context of mobility, proxy caches are used not only to cache and retrieve documents but also to manage user mobility. These proxies are access points to the Internet for mobile terminals. Proxies implement functionalities dedicated to the transfer of data to mobile terminals, such as compression, filtering, format conversion [16][1]. It is further mandatory to account for the disconnection of mobile terminals, as their connectivity is highly dynamic in nature. Typically, a mobile terminal can be in either of the three following connection modes, shifting from one to another depending on the user's location: strongly connected (i.e., use of a fixed LAN), less strongly connected (i.e., use of a wireless network), or disconnected (i.e., due to the absence of connectivity or to save energy). Solutions to the above issue include the one offered by the ARTour Web Express software [4], which queues user requests that are made during an unexpected disconnection. These requests are later processed as a background task when the connection is restored. In addition, the user may act upon local cache management by requesting to keep in the cache the pages that he/she marks using the cache browsing tool. However, not all the disconnections are unexpected, and some may be anticipated by the user. In this case, the user may indicate the

kind of Web pages she will be willing to access (e.g., subject of interest, URLs). The information is then exploited, possibly in combination with the user's profile, to prefetch a number of pages. Similar to our solution, the one of [3] in the area of service discovery and composition or [19] in the area of databases use a decentralized model in an ad hoc environment. Complementing proxy solutions with ad-hoc-based ones will allow for both enhanced connectivity and Web access at low cost. In that context, the caches of the mobile terminals cooperate in a way similar to proxy caches in transversal cooperative cache systems. However, the cooperative scheme must be devised so as to account for: the continuously changing set of cooperating terminals, the heterogeneity of terminals in terms of resource availability, and energy saving.

### 3 Cooperative Caching in Ad Hoc Networks

A number of ad hoc routing protocols are implemented over a base WLAN. Basically, these protocols maintain a *routing table* and differ in the way they manage it (§ 3.1). Using the ad hoc network protocol that offers the best trade-offs in terms of energy consumption and response time, we propose a specialization of the protocol that is specifically aimed at handling remote access to Web pages (§ 3.2). We then introduce our ad hoc cooperative caching protocol, which has primarily been designed to minimize energy consumption (§ 3.3).

#### 3.1 Ad Hoc Networking

The main issue to be addressed in the design of an ad-hoc (network) routing protocol is to compute an optimal *communication path* between any two mobile terminals. This computation must minimize the number of control messages exchanged in order to minimize network congestion and energy consumption [7].

There exist two base types of ad hoc protocols: *proactive* and *reactive*. Proactive protocols (e.g., OLSR [6]) update their routing table periodically. Compared to proactive protocols, reactive protocols (e.g., AODV [20], DSR [15]) check the validity of, and possibly compute, the communication path between any two mobile terminals only when a communication is requested between the two. ZRP [10] is a hybrid protocol that combines the reactive and proactive modes. The design rationale of ZRP is that it is considered advantageous to accurately know the *neighbours* of any mobile terminal (i.e., mobile terminals that are accessible in a fixed number of hops), since they are close. Hence, communicating with neighbours is less expensive and neighbours are most likely to take part in the routing of the messages sent from the terminal. As a result, ZRP implements: (i) a proactive protocol for communication with mobile terminals in the neighbourhood, and (ii) a reactive protocol for communication with the other terminals. With respect to a given mobile terminal, its neighbourhood is referred to as its *zone*. Notice that it would be interesting to dynamically set the number of hops used to define a zone, according to the number of mobile terminals that are present in the perimeter. ZRP is currently among the most efficient ad hoc

routing protocols in terms of network load due to control messages and hence of associated energy consumption on mobile terminals. We thus use ZRP over IEEE 802.11, as the base ad hoc protocol for realizing ad hoc cooperative caching among mobile terminals. Mobile terminals belonging to the zone of a given terminal then form a cooperative cache system for this terminal since the cost for communicating with them is low both in terms of energy consumption and message exchanges [11]. However, cooperative caching must not be restricted to the mobile terminals belonging to the zone: low-cost reachability of a base station must be accounted for as well as knowledge of a terminal that does not belong to the zone but that is likely to store a requested Web document given commonalities in performed Web accesses.

### 3.2 Ad Hoc Communication for Web Caching

A mobile terminal may get Web data that are not cached locally through two communication paths: (i) using the infrastructure-based mode, the terminal may interact with the nearby base station, which forwards the request to the Web, (ii) using the ad-hoc-based mode, the terminal may request for the data to the mobile terminals in its communication range (i.e., accessible in one hop in a base WLAN or in a number of hops using some ad hoc network routing protocol). The former is the most efficient in terms of energy consumption; the mobile terminal sends a single message to a base station that has an infinite energy budget. However, the ad-hoc-based mode must be enabled for the case where a base station is not reachable in one hop. In this case, a base station can still be reachable in a number of hops, thanks to mobile terminals forwarding the requests. For instance, UMTS supports such a mechanism to extend the range of the infrastructure<sup>2</sup>. The number of hops that are necessary to access a base station depends on the terminal's location. Let  $N$  be this number, then any mobile terminal that is at a distance greater or equal to  $N$  is not contacted to get a document. Figure 1 depicts the case where the mobile terminal  $A$  reaches the base station  $D$  in 3 hops, using the mobile terminals  $B$  and  $C$  for routing the request. Then, if either a mobile terminal in the zone of  $A$  (e.g.,  $B$  belonging to the path leading to the base station or any other terminal in the zone) or a known mobile terminal located outside the zone but at a lower distance than the base station  $D$  (e.g.,  $C$  that is in the path leading to  $D$  or  $E$  that does not belong to the path) holds the requested document in its cache, it returns it to  $A$ . Otherwise, the request reaches  $D$ , and  $D$  forwards it to the Web. We get the following ad hoc communication protocol over ZRP, to retrieve a remote Web object  $W$ , with respect to a given mobile terminal  $A$ :

- *In-zone communication:*

If a base station is in the zone of  $A$ , then  $A$  requests for  $W$  through the base station only. Otherwise,  $A$  broadcasts the request message for  $W$  to the mobile terminals in the zone of  $A$ , incurring a low energy cost since the routing table contains the necessary information.

---

<sup>2</sup> <http://www.3gpp.org>

· *Peer-to-peer communication:*

If  $W$  is not cached by any of the mobile terminals in the zone of  $A$  and if there is no base station in the zone of  $A$ , then a peer-to-peer communication (or point to point messaging) scheme is realized with mobile terminals that are known to share interests with  $A$  (see § 3.3) and that are at a distance that is less than the one between  $A$  and the nearest base station. Mobile terminals outside the zone of  $A$  are basically known through two ways: (i) they belong to the path used to reach the nearest base station, (ii) they were previously either in the zone or in the path used to reach the base station. The request for  $W$  is ultimately forwarded to the nearest base station.

Based on the above, the communication cost and hence the energy cost associated with getting a Web object is kept to a minimum: (i) broadcast is within a zone and is thus unexpensive by construction of ZRP, (ii) peer-to-peer communication occurs only with mobile terminals that are both the most likely to store a requested object and closer than a base station.

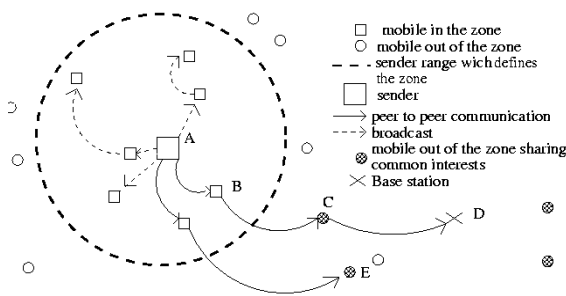


Fig. 1. Getting Web data

### 3.3 Ad Hoc Cooperative Caching

Without a proxy-type architecture that centralizes requests, local statistics are relied on for a mobile terminal  $A$  to identify mobile terminals that are likely to store a Web object requested on  $A$ . Such statistics are maintained on  $A$  through a *terminal profile* for every mobile terminal with which  $A$  interacts. The terminal profile is characterized by a value that counts the number of times the corresponding mobile terminal either is known to cache an object requested by  $A$  or requested for an object to  $A$  that  $A$  had in its cache. The value of the terminal profile is used to identify the mobile terminals with which peer-to-peer communication is undertaken. The list of known mobile terminals outside the zone and that are at a distance less than a base station are then weighed according to the value of  $\mathcal{F} = \text{terminal profile} \div \text{hops}$  where the number of

hops,  $hops$ , is obtained from the routing table. Mobile terminals for which the value of  $\mathcal{F}$  is the greatest are first contacted and the process is iterated until a *hit* message is received or there are no more mobile terminals eligible for the request. In addition to the management of *terminal profiles* to identify mobile terminals that share common interests, we must account for the heterogeneity of the terminals' capacity (i.e., battery, processing, storage, communication). In particular, for two mobile terminals that are equally likely to store a requested object, it is better to contact the one that has the greatest capacity.

Ignoring the case where a base station is accessible in the zone and given the ad hoc communication protocol aimed at Web caching that was discussed in the previous subsection, the request for a Web object  $W$  that is not cached locally, from a mobile terminal  $A$  is handled as follows: (i)  $A$  first broadcasts the requests for  $W$  within the zone; (ii) if  $W$  is not retrieved then the retrieval protocol iterates on sending the request for  $W$  to known mobile terminals outside the zone and at a distance less than the base station according to the maximization of  $\mathcal{F}$ ; (iii) the base station is ultimately contacted. The processing of requests is further handled as follows. A mobile terminal that receives the request for  $W$  and caches it increments its local value of  $A$ 's terminal profile. If the terminal is further willing to cooperate (e.g., absence of energy safeguarding or of security policy enforcement), it returns a *hit* message, which embeds:

- (i) *TTL* that gives the Time To Live field of the document.
- (ii) *Capacity* that characterizes the capacity of the terminal to handle requests, whose value is in the range  $[0..1]$ , 1 denoting the highest capacity. Currently, we use a simple scheme to set the value of *Capacity*; it is equal to the percentage of the energy budget that is left. It is part of our future work to investigate a more accurate way of computing *Capacity*.

For every *hit* message that it receives,  $A$  increments the *terminal profile* of the sender. Among the mobile terminals that replied by a *hit* message,  $A$  selects the terminal from which  $W$  should be obtained, that is the one that maximizes the following function<sup>3</sup>:  $\mathcal{R} = Capacity \times (\lambda \times TTL + \mu \times hops)$  where  $\lambda$  and  $\mu$  are set so as to favour communication with the closest nodes. More precisely, the value of  $\mathcal{R}$  is computed as follows. The metrics used for consistency (duration given by TTL) and for communication cost (number of hops) are distinct and should be made comparable. As a dispersion measure of TTL and hops, we use the standard deviation. Thus, considering that  $A$  received  $N$  hits messages for  $W$ ,  $A$  computes the mean  $m_{TTL}$  (resp.  $m_{hop}$ ) of the TTL (resp. hops) and the standard deviation  $\sigma_{TTL}$  (resp.  $\sigma_{hop}$ ):

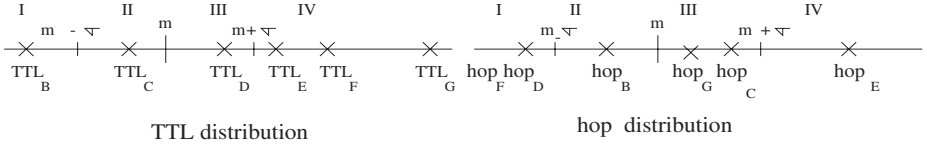
$$m_{TTL} = \frac{1}{N} \sum_{i=1}^N TTL_{mobile_i}, \quad \sigma_{TTL} = \sqrt{\frac{1}{N} \sum_{i=1}^n [TTL_{mobile_i} - m_{TTL}]^2}$$

$$m_{hop} = \frac{1}{N} \sum_{i=1}^N hop_{mobile_i}, \quad \sigma_{hop} = \sqrt{\frac{1}{N} \sum_{i=1}^n [hop_{mobile_i} - m_{hop}]^2}$$

<sup>3</sup> For the case where the selected terminal is no longer accessible, e.g., due to energy safeguarding, the request will be sent to the next eligible terminal and will be so until the page is received.

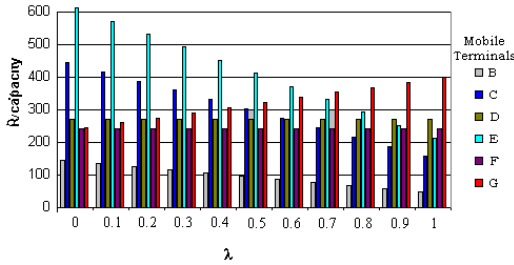


For illustration, Figures 2 gives a set of TTL and hop values considering reception of hit messages from terminals [B,..,G]. We distinguish four intervals for the value of TTL: level  $I_{1_{TTL}} = ]-\infty, m_{TTL}-\sigma]$ , level  $I_{2_{TTL}} = [m_{TTL}-\sigma, m_{TTL}]$ , level  $I_{3_{TTL}} = [m_{TTL}, m_{TTL}+\sigma]$ , level  $I_{4_{TTL}} = [m_{TTL}+\sigma, +\infty]$ .



**Fig. 2.** Energy consumption.

The value of TTL is then mapped into one of the ranges [0..1], [1..2], [2..3] and [3..4], if it belongs to  $I_{1_{TTL}}$ ,  $I_{2_{TTL}}$ ,  $I_{3_{TTL}}$  and  $I_{4_{TTL}}$ , respectively. The value taken in the target range is then set proportionally to the value of the TTL in the initial range,  $I_{j_{TTL}}$ ,  $j \in [1..4]$ , and is the one used in the computation of R. For illustration, Table 1 gives the initial TTL and hops values, and values under the unified metric, considering the distribution given in Figure 2. Then the values of  $\lambda$  and  $\mu$  are set so that  $\lambda + \mu = 1$  and will be chosen according to the respective weighing factors for the consistency and communication cost. Figure 3 gives the value of  $\frac{\mathcal{R}}{\text{capacity}}$  for the mobile terminals B,..,G considered in Figure 2, according to the value of  $\lambda$ , ranging from 1 (i.e.  $\mu=0$  and the communication cost is ignored) to 0 (i.e.  $\mu=1$  and the communication cost is the only selective factor).



**Fig. 3.** R/capacity function.

We do not use *miss* messages for mobile terminals to notify that they do not cache a requested object. This is to minimize both network load and energy consumption. Hence, we use timeouts to detect the absence of a requested object. The value of the timeout is set according to the greatest number of hops that are involved to interact with the mobile terminals to which the mobile object is requested,

**Table 1.** TTL and hop metric

mobile	TTL	TTL	TTL metric	hop	hop metric
b	$\frac{1}{2}$	$(m - \sigma) - \frac{1}{2}\sigma$	$1 - \frac{1}{2} = \frac{1}{2}$	4	$1 + \frac{1}{\sqrt{5}}$
c	$\frac{4}{7}$	$m - \frac{3}{7}\sigma$	$2 - \frac{3}{7} = \frac{11}{7}$	8	$\frac{25}{7} + \frac{2}{\sqrt{5}}$
d	$\frac{27}{10}$	$(m + \sigma) - \frac{3}{10}\sigma$	$3 - \frac{3}{10} = \frac{27}{10}$	2	$\frac{27}{10}$
e	$\frac{13}{4}$	$(m + \sigma) + \frac{1}{4}\sigma$	$\frac{17}{8}$	10	$\frac{49}{8}$
f	$\frac{23}{6}$	$(m + \sigma) + \frac{5}{6}\sigma$	$\frac{29}{12}$	2	$\frac{29}{12}$
g	5	$(m + \sigma) + 2\sigma$	4	6	$2 + \frac{1}{2\sqrt{5}}$

together with the current network load. Upon expiration of the timeout, if *hit* messages have been received, the Web object will be requested to the mobile terminal that maximizes  $\mathcal{R}$ . Otherwise, the next iteration of the cooperative caching protocol is processed (i.e., from the broadcast step to the peer-to-peer iterative step). For the case where a *hit* message is received after timeout expiration, while the object is still not retrieved, the message is accounted for in the current step of the protocol.

## 4 Local Caching

The ad hoc cooperative caching protocol introduced in the previous section is complemented by a local caching strategy, which is adaptive according to the current capacity of the terminal. Local cache management mainly amounts to the implementation of a replacement algorithm that is run when the cache gets full. We weigh every cached document according to both its probability of being accessed in the future and the energy cost associated with getting remotely the document. Documents with the lowest weighs are those that are removed from the cache. The document weigh is computed according to the following criteria:

- **Popularity.** The *Popularity* value serves approximating the probability of future access, both on the terminal and from remote terminals, as enabled by the cooperative caching protocol. The probability is approximated according to the number of times the document has been requested since it has been cached.
- **AccessCost.** The *AccessCost* value gives an estimate of the energy cost associated with getting the document remotely if it is to be removed from the cache. This cost varies depending on whether: a base station is accessible in the zone of the terminal, the document is cached on a mobile in the zone of the terminal, communication out of the zone is required to retrieve the document. In the first two cases, the cost is quite low, while it may be quite high in the third case. The value of the access cost is computed according to the energy consumption associated with intra-zone and inter-zone communication (see § 5). Intra-zone communication holds if a base station is known to be in the zone. It is further assumed if the document

was obtained from a terminal that is still in the zone, as identified using the routing table.

- **Coherency.** A document is valid for a limited lifetime, which is known using the *TTL* field. However, when the energy remaining on the terminal is low, it is better to favour energy saving over the accuracy of the document. Hence, the value of *Coherency* is equal to  $\nu \times TTL$  where  $\nu$  increases as the available energy decreases.

We get the following function to compute the document weigh:

$$\mathcal{W} = \alpha \times Popularity + \beta \times AccessCost + \gamma \times Coherency + \delta \times size$$

The values of  $\alpha$ ,  $\beta$ ,  $\gamma$  and  $\delta$  are set so as make the values of *AccessCost*, *Popularity*, *Size*, and *Coherency*, decreasingly prominent factors for deciding whether a document should be kept in the cache. The metric used to compare the differents parameters is computed in a way similar to the one described in Section 3 for the function  $\mathcal{R}$ . Notice that our  $\mathcal{W}$  function offers similarities with the one used by hybrid replacement algorithms that were proposed in the literature for Web caching on stationary hosts (e.g., [14]). In the same way, we use a function that accounts for a combination of criteria in the replacement decision. However, our function differs in that energy saving is a prominent criterion.

## 5 Evaluation

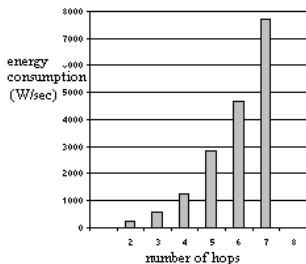
Control messages generated by ZRP and messages associated with ad-hoc cooperative caching affect the network traffic and energy consumption on mobile terminals. In this section, we examine the energy consumption induced by:

- Maintening the routing table on each mobile terminal with ZRP.
- Using the ad-hoc cooperative strategy to identify terminals that are likely to store the requested Web data.
- Retrieving the Web data.

The above includes the energy cost due to both communication and computation. However, we do not consider the latter cost since it is induced by any local cache management, and it is negligible compared to the energy cost of communication: the energy cost of transmitting 1Kb is approximately the same as the energy consumed to execute 3 million intructions [21]. We further use the following wireless interface for our evaluation: 2.4Ghz DSSS lucent IEEE 802.11 WaveLan PC “Bronze” 2Mbps on the terminal.

### 5.1 Energy Consumption of ZRP Control Messages

Performance of ZRP has been evaluated in [11, 10] using event-driven simulation; this evaluation is gauged by considering the control traffic generated by ZRP. Figure 4 gives the energy consumption generated by the reception of ZRP traffic. Messages are used to maintain the zone routing tables with a beacon broadcast period of 0.2/s. To obtain the path to a mobile terminal out of the zone, additional control messages are exchanged only on-demand.



**Fig. 4.** Energy consumption generated by the reception of ZRP traffic in a zone per /s

Given the energy consumption generated by ZRP, we evaluate energy consumption associated with ad-hoc cooperative caching, as the sum of the energy consumption induced for the various mobile terminals that are involved (both in and outside the zone) in the cooperation, which adds to the energy cost induced by ZRP.

## 5.2 Energy Consumption of Ad-Hoc Networking

Focusing on the energy cost associated with communication, the cost associated with the emission of one message is the sum of:

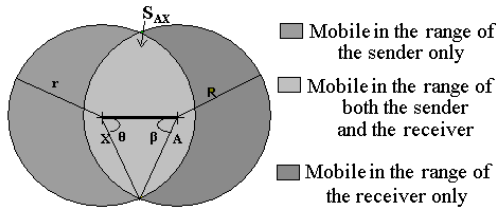
- The cost of emission for the sender node.
- The cost of reception for the destination node.
- The cost of reception and emission for mobile nodes forwarding the message.
- The cost of reception for non-destination nodes (i.e., terminals that receive control messages due to their location with respect to the aforementioned nodes, although they are not involved in the routing of the message).

To minimize collisions in point-to-point traffic, the IEEE 802.11 protocol provides the following collision avoidance mechanism. Prior to any point-to-point transmission, the sender broadcasts a RTS (Request To Send) control message, which specifies the destination node and the data size (for duration). The sender then waits for a CTS (Clear To Send) message from the destination node. Once it receives the CTS, the sender sends the data message. Finally, the destination node sends an ACK message upon the reception of the data message. Therefore, the energy consumed by any mobile terminal for sending, receiving or discarding a message is given by the linear equation [9]:  $\varepsilon = m * size + p$ ; where *size* is the message size, and *m* (resp. *p*) denotes the incremental (resp. fixed) energy cost associated with the message. The value of  $\varepsilon$  is detailed hereafter, for the various mobile terminals affected by the message transmission. Table 2 gives the energy cost, relative to the size of the message, for the sender and destination nodes (i.e., the actual sender and destination nodes and the forwarding nodes that act as both sender and destination nodes). For the sender X, the high value of the

**Table 2.** Energy consumption on nodes for point to point communication

Mobile	Energy consumption ( $\mu W.sec$ )	m ( $\mu W.sec$ )	p ( $\mu W.sec$ )
Sender X	$\varepsilon_{send} = m_{send} * size + p_{send}$	$m_{send} = 1.9$	$p_{send} = 454$
Destination A	$\varepsilon_{dest} = m_{dest} * size + p_{dest}$	$m_{dest} = 0.5$	$p_{dest} = 356$
Non-destination nodes			
in range of sender X	$\varepsilon_{AX} = m_{AX} * size + p_{AX}$	$m_{AX} = -0.22$	$p_{AX} = 210$
in range of sender X	$\varepsilon_X = m_X * size + p_X$	$m_X = -0.04$	$p_X = 90$
in range of destination A	$\varepsilon_A = m_A * size + p_A$	$m_A = 0$	$p_A = 119$

incremental cost  $m_{send}$  is due to the emission of the data message. The fixed cost  $p_{send}$  results from the reception of two control messages (CTS and ACK) and from the emission of the RTS message. For the destination node A, the fixed cost  $p_{dest}$  is due to the emission of two control messages (CTS and ACK) and to the reception of the RTS message. The value of  $m_{dest}$  for A follows from the reception of the data message. Table 2 also gives the energy cost, relative to the size of the message, for non-destination nodes. Non-destination nodes in the range of the sender receive RTS messages and thus enter a reduced energy consumption mode during data emission; this leads to have a negative value for  $m_{AX}$  and  $m_X$  since the energy consumption is less than the one in the idle mode. The fixed cost  $p_{AX}$  for non-destination nodes in the range of both the sender and the receiver is greater than the fixed cost  $p_X$  for non-destination nodes in the range of the sender only, because the latter do not receive the CTS and ACK messages. Finally, non-destination nodes in the range of the destination node but not the sender receive the CTS and ACK messages. On the other hand, they do not receive the RTS message, and thus cannot enter in the reduced energy consumption mode; this leads to have the incremental cost  $m_A$  equal to 0.

**Fig. 5.** Mobile terminals in the range of the sender and of the destination node.

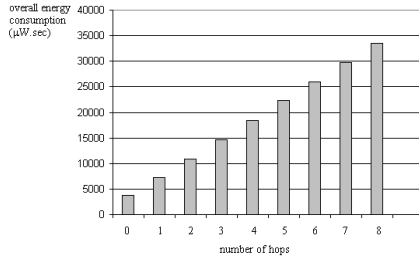
Consider now a network of 500 mobile terminals whose communication range is of about 250m, that is such that the mobile terminals are uniformly distributed in the network surface  $S$  with  $S = 4000[m] \times 4000[m]$ <sup>4</sup>. In a zone (see Figure 5), all the mobile terminals consume the same energy. Thus, the overall energy

<sup>4</sup> This network is taken as an example, as it is used in [11] for the evaluation of ZRP.

**Table 3.** Energy consumption in a zone.

Mobile terminal in range of	Surface	Number of mobile terminals	Total energy for a zone
the sender X & the receiver A	$S_{AX} = \theta(r^2 - a_1^2) + \beta(R^2 - a_2^2)$	$n_{AX} = \frac{NS_{AX}}{S}$	$n_{AX}\epsilon_{AX}$
the sender X	$S_X = \pi r^2 - S_{AX}$	$n_X = \frac{NS_X}{S} = \pi r^2 - \theta r^2 \sin^2 \theta + \beta R^2 \sin^2 \beta$	$n_X \epsilon_X$
the receiver A	$S_A = \pi R^2 - S_{AX}$	$n_A = \frac{NS_A}{S} = \pi R^2 - \theta r^2 \sin^2 \theta + \beta R^2 \sin^2 \beta$	$n_A \epsilon_A$

consumption within a zone is the sum of the energy consumed by every mobile terminal in this zone (see Table 3). Then, the energy consumption of the overall network is the sum of the energy consumed per zone that is traversed. Figure 6 gives the energy consumption associated with the delivery of a message of 1 Kb to the destination node according to the number of hops, which is the sum of the energy consumed on all the nodes involved in the communication. The figure clearly demonstrates that the energy consumption increases with the number of hops. This directly follows from the resulting increase of both mobile terminals forwarding the message and non-destination nodes receiving control messages. Table 4 further evaluates the impact of the network density on energy consumption. For a constant number of hops (=3), we see that increased density of mobile terminals results in additional energy consumption for non-destination nodes. But, the ratio  $\frac{\epsilon_{send} + \epsilon_{dest} + \epsilon_{forwarding\ mobile\ terminals}}{\epsilon_{overall\ non-destination\ mobile\ terminals}}$  highlights the weak impact of message reception on non-destination nodes, on the overall energy consumption. Indeed, for 600 mobile terminals and a destination node at 4 hops of the sender, the energy consumed by non-destination nodes is about 6 times less than the energy consumed by the sender, the receiver and the 3 forwarding mobile terminals.

**Fig. 6.** Energy consumption in the network for the retrieval of Web data.

### 5.3 Energy Consumption of Ad-Hoc Cooperative Caching

Having examined the energy consumption associated with data delivery, we now focus on the energy consumption induced by our ad-hoc cooperative caching

**Table 4.** Energy consumption according to the network density.

Number of Mobile terminals in the network surface	500	600	700
Energy consumption of non-destination mobile terminals ( $\mu W.sec$ )	1389	18959	2136
ratio	9	7	6.11

protocol, which retrieves a requested Web page among mobile terminals. A request message for a Web page includes broadcasting (with mobile terminals in the requester's zone) and peer-to-peer communication (with mobile terminals sharing the same interests). Before broadcasting a message, the sender listens to the channel; if no traffic is detected, the message is broadcasted. Table 5 gives the energy consumption induced by a broadcast while the one associated with peer-to-peer communication is given in Table 2. The difference between the energy consumption associated with broadcasting and the energy consumption associated with peer-to-peer communication is due to the emission of control messages (RTS, CTS and ACK messages) in the case of peer-to-peer communication. Figure 7 gives the energy consumed by cooperative caching according

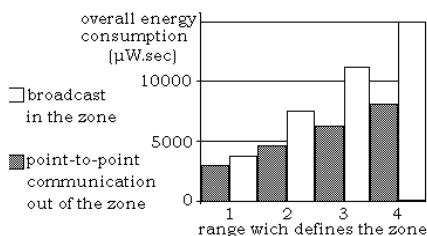
**Table 5.** Energy consumption on the sender and destination nodes for broadcast.

Mobile	Energy consumption ( $\mu W.sec$ )	m ( $\mu W.sec/byte$ )	p ( $\mu W.sec/byte$ )
Sender X	$\varepsilon_{brsend} = m_{brsend} * size + p_{brsend}$	$m_{brsend} = 1.9$	$p_{brsend} = 266$
Destination A	$\varepsilon_{brdest} = m_{brdest} * size + p_{brdest}$	$m_{brdest} = 0.5$	$p_{brdest} = 56$

to the number of hops that defines the zone. Precisely, Figure 7 gives the cost associated with broadcasting and peer to peer communication at a distance that adds one hop to the number of hops that defines the zone. As expected, the overall energy consumption of both broadcast and peer-to-peer communication increases with the number of hops that defines the zone (because more mobiles are in the zone). Comparing energy consumption of broadcasting with the one of peer-to-peer communication, we find that the cost added by broadcasting is weak compared to the number of mobile terminals contacted, and hence supports our approach.

## 6 Conclusion

Mobile technology has reached a stage that enables foreseeing easy access to information technology anywhere, anytime. However, it is necessary to devise



**Fig. 7.** Energy consumption in the network for the broadcast and the peer-to-peer communication.

adequate software and hardware solutions to mask unstable connectivity and limited energy. This paper has addressed one such solution, which focuses on enabling Web accesses from mobile terminals. Currently, Web access is easy to realize if the mobile terminal is in the communication range of a base station of either a WLAN or a global wireless network. However, this cannot always be assumed due to the financial cost associated with the deployment of the underlying infrastructure. In addition, the systematic use of a global wireless network is quite costly for users. Instead of relying on a base station for accessing the Web, an alternative solution is to exploit ad hoc networking, which allows for remote communication at no financial cost and also reaching a base station of a WLAN in a number of hops. In that context, we have proposed a cooperative Web caching system for ad hoc networks, which enables mobile terminals to share Web pages, while minimizing the resulting energy cost. Our system lies in implementing on each mobile terminal: (i) an ad hoc cooperative caching protocol that selects the mobile terminals from which a requested page should be retrieved, in a way that minimizes both energy consumption and network load, (ii) a caching strategy that maintains the local cache in a way that minimizes resource consumption. Such a system is well suited for location-dependent data such as multimedia guided tours. It is further a cheap solution compared to existing ones since it is not based on the infrastructure mode. Preliminary assessment of our proposal has been addressed by providing the energy cost that our system incurs for mobile terminals. We are currently working on further assessment of our proposal, which lies in the implementation of a simulator to thoroughly examine the behaviour of our system. Experiment using the simulator will in particular serves tuning the various weighing functions that we use for cooperative caching. We are also implementing our proposition for multimedia guided tours on the Web.

### Acknowledgments:

This work has been partially funded by the OZONE European IST project<sup>5</sup>.

<sup>5</sup> <http://www.extra.research.philips.com/euprojects/ozone/>



## References

- [1] C. Bennett. *Practical Wap, Developing applications for the Wireless Web*. Cambridge University Press, 2001.
- [2] C.M. Bowman, P.B. Danzig, and D. Hardy. The Harvest information discovery and access system. *Computer Networks and ISDN Systems*, 28(1–2), 1995.
- [3] D. Chakraborty, F. Perich, A. Joshi, T. Finin, and Y. Yesha. A reactive service composition architecture for pervasive computing environments. In *proc. of PWC*, 2002.
- [4] H. Chang, C. Tait, N. Cohen, and M. Shapiro. Web browsing in a wireless environment: disconnected and asynchronous operation in ARTour Web express. In *proc. of MOBICOM*, 1997.
- [5] J.H. Chang and L. Tassiulas. Energy conserving routing in wireless ad-hoc networks. In *proc. of INFOCOM (1)*, 2000.
- [6] T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, A. Qayyum, and L. Viennot. Optimized link state routing protocol. In *proc. of INMIC*, 2001.
- [7] S.R. Das, C.E. Perkins, and E.E. Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In *proc. of INFOCOM (1)*, 2000.
- [8] L. Fan, P. Cao, J. Almeida, and A.Z. Broder. Summary cache: a scalable wide-area Web cache sharing protocol. *IEEE/ACM Transactions on Networking*, 8(3), 2000.
- [9] L. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in ad hoc networking environment. In *proc. of INFOCOM (5)*, 2001.
- [10] Z. Haas. A new routing protocol for the reconfigurable wireless networks. In *proc. of ICUP*, 1997.
- [11] Z. J. Haas and M.R. Pearlman. The performance of query control schemes for the zone routing protocol. *ACM/IEEE Transactions on Networking* 9(4), 2001.
- [12] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *proc. of HICSS*, 2000.
- [13] Intel. Mobile power guide v1.00. <http://developer.intel.com/design/mobile/intelpower/>, 2000.
- [14] Z. Jinag and L. Kleinrock. Web prefetching in a mobile environment. *IEEE Personal Communications*, 5(8), 1998.
- [15] D. Johnson and D. Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*. 1996.
- [16] E. Markatos and C. Chronaki. A top-10 approach to prefetching on the Web. In *proc. of INET*, 1998.
- [17] J.M. Menaud, V. Issarny, and M. Banatre. A scalable and efficient cooperative system for Web caches. In *IEEE Concurrency*. 8(3), 2000.
- [18] K. Nakano and S. Olariu. Energy-efficient randomized routing in radio networks. In *Proc. of DIALM*, 2000.
- [19] F. Perich, S. Avancha, D. Chakraborty, A. Joshi, and Y. Yesha. Profile driven data management for pervasive environments. In *proc. of DEXA*, 2002.
- [20] C. Perkins. Ad-hoc on-demand distance vector routing. In *Proc. of WMCSA*, 1999.
- [21] G.J. Pottie and W.J. Kaiser. Wireless integrated network sensors. *Communications of the ACM*, 2000.
- [22] V. Valloppillil and K.W. Ross. Cache array routing protocol, Internet draft. <http://ircache.nlanr.net/Cache/ICP/carp.txt>, 1998.