Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE DIRECT®

ELSEVIER

Performance Evaluation 53 (2003) 273–294

PERFORMANCE EVALUATION
An International Journal

www.elsevier.com/locate/peva

Performance comparison of power-saving strategies for mobile Web access☆

G. Anastasi [a,*], M. Conti [b], E. Gregori [b], A. Passarella [a]

[a] *Department of Information Engineering, University of Pisa, Via Diotisalvi 2, 56122 Pisa, Italy*
[b] *IIT Institute, CNR Research Area, Via G. Moruzzi, 1-56124 Pisa, Italy*

Abstract

One of the critical issues in mobile Web access is the usage of limited energy resources of mobile computers. Unfortunately, the legacy TCP/IP architecture is very inefficient. This work proposes and analyzes power-saving strategies for mobile Web access. Specifically, in this paper we develop an energy-consumption model for Web transactions and, based on it, we propose and compare four different energy saving strategies: ideal, Indirect-TCP (I-TCP), local and global. The ideal strategy is unfeasible but it is used as a reference bound as it guarantees the lowest energy consumption. The other strategies have been implemented and compared in a real test-bed. The performance comparison is carried out by measuring two main performance figures: the energy spent for downloading a Web page, and the associated transfer-time. Experimental results show that relevant energy saving is achievable and that, among the feasible strategies, the global one gives the best performance: with this strategy we can save (on average) up to 88% of the energy. Furthermore, our results indicate that this power saving is obtained without a significant increase in the transfer-time perceived by the users (on average, 0.2 s). Finally, by comparing the feasible strategies, we observe that the global one is much closer to the ideal case than the other strategies. In detail, the global strategy is about twice more efficient than the local one, and eight times more efficient than the I-TCP strategy.

Keywords: Power saving; Mobile Internet; Indirect-TCP; Web; Performance analysis; Measurements

1. Introduction

In the field of mobile computing, the mobile Internet is one of the most interesting areas. Users are no longer forced to access information at their desktop, since data are available where they are, at *any time* and *any place*. However, several problems must be solved when integrating a mobile device in the

☆ Work carried out under the financial support of the Italian Ministry for Education and Scientific Research (MIUR) in the framework of the projects: "Web Systems with QoS Guarantees", "Internet: Efficiency, Integration and Security" and FIRB-PERF.

* Corresponding author. Tel.: +39-050-568-559; fax: +39-050-568-522.
E-mail addresses: g.anastasi@iet.unipi.it (G. Anastasi), marco.conti@iit.cnr.it (M. Conti), enrico.gregori@iit.cnr.it (E. Gregori), a.passarella@iet.unipi.it (A. Passarella).

0166-5316/03/$ – see front matter
doi:10.1016/S0166-5316(03)00066-X

Internet. As is well known, one of the most critical problems is the *energy consumption* [1–3]. In this work we consider how to introduce power saving in mobile Web access.

Efficient energy management has been approached at different levels of a mobile system architecture: physical transmission [4–6], MAC protocols [7–11], disk and CPU management [12–16], and applications [2,17–22]. Experimental results show that a relevant part of the energy available on a mobile computer is drained by the wireless interface. More precisely, the networking impact on energy consumption varies from about 10% in laptops [23] up to 50% in small-size hand-held devices, such as PDAs [24]. Therefore it is vital to design energy-efficient networking subsystems.

The key point in energy-aware networking is the consumption model of a wireless interface. Specifically, the wireless interface consumes nearly the same amount of energy in the receive, transmit and idle states (see, for example, the 802.11 "Wi-Fi" environment [25]). Therefore, the energy consumption is approximately proportional to the time during which the wireless interface remains switched on. The maximum power saving can therefore be achieved by transmitting data as quick as possible and, immediately after, turning the wireless interface into a power-saving mode. Many researchers have proposed power-saving policies based on this idea [23,24,26–28]. The innovative contribution of the approach presented in this paper is the exploitation of the application semantic to determine the best time instants for switching the wireless interface on and off.

In this paper we refine, and extensively evaluate, the power-saving architecture defined in [29]. Specifically, we compare the performance of four different power-saving strategies aimed at reducing the energy consumed during a Web-page download. The first strategy is a pure Indirect-TCP (I-TCP) architecture [30,31]. With respect to the legacy TCP architecture, this solution improves the throughput achieved by the mobile host, thus reducing the transfer-time. Hence, it indirectly contributes to power saving even though no energy-management mechanism is explicitly introduced in the system. Explicit energy management is included in the other policies we consider, all obtained by enhancing the I-TCP architecture. The *local* strategy switches the wireless interface off when the user is reading the Web page, i.e., it exploits information that are locally available at the client browser. The third approach, referred to as the *global* strategy, in addition to local information, exploits statistical information about Web traffic. Finally, an *ideal* (unfeasible) strategy that guarantees the minimum power consumption is also considered. Throughout this paper, the ideal and I-TCP strategies provide the lower and upper bound for energy consumption, respectively.

We implemented the feasible power-saving strategies and tested them extensively in a real Internet scenario. Our performance study is based on two main performance figures: I_{ps} and I_{pd}. I_{ps} is used as a power-saving index. It measures the energy consumption of a specific strategy expressed as a percentage of the energy consumption related to I-TCP strategy. I_{pd} measures the impact of the power-saving strategy on the user response time (URT), i.e., the time interval elapsed from a user request for a Web page to its rendering on the mobile device.

The experimental results show that the global strategy exhibits the best achievable performance. It saves, on average, 88% of the energy consumed by the I-TCP approach and has a negligible impact on the URT (the URT increase is of 0.2 s on average, and is below 1.8 s with probability 0.9).

The paper is organized as follows. Section 2 models the statistical properties of a Web transaction and its energy consumption. Section 3 defines the power-saving strategies and their energy consumption. Section 4 presents the experimental test-bed, and introduces the performance indexes used in our measurement study. Sections 5 and 6 analyze the performances of the different power-saving strategies. Section 7 concludes the paper.

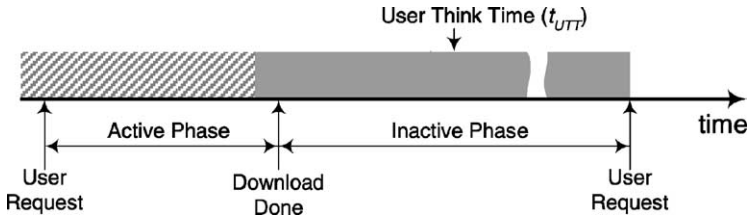

Fig. 1. The Web-page download as an active and an inactive phase.

2. System model

The power-saving strategies evaluated in our system are application-dependent, i.e., they exploit the application semantic to optimize the energy consumption. Hence, as a preliminary step, it is necessary to characterize the traffic profile generated by Web browsing.

Many papers in literature provide mathematical Web traffic characterizations [32–37], and show that, with an appropriate analysis of the Web servers logs, it is possible to model the Web user behavior [34,35].

2.1. Single user's traffic model

The activity of an individual user can be represented as a series of successive requests for Web pages. As shown in Fig. 1, each request causes a two-phase process. During the first phase, the Web page is downloaded from the server to the client while, in the second phase, the user reads the contents. The first phase is typically named active phase because during this time interval data flow on the network. The second phase is referred to as inactive phase because there is no network activity.

The inactive phase is composed by a unique time interval (t_{UTT} in Fig. 1). This time interval is known as the *Inactive OFF Time* or *User Think Time* (UTT), and is typically longer than 30 s (i.e., it is practically much longer than the active phase length). User Think Times are distributed according to a Pareto law [34,35]:

$$p(t_{\mathrm{UTT}}) = \alpha k^{\alpha} t_{\mathrm{UTT}}^{-(\alpha+1)}, \quad t_{\mathrm{UTT}} \geq k,\ \alpha = 1.5,\ k = 30, \tag{1}$$

where k is the *scale parameter* and α the *shape parameter*.

Fig. 2 provides a graphical representation of a typical active phase. A Web page usually consists of a set of files: an HTML *main file* and a number of *embedded files*. Specifically, the main file contains

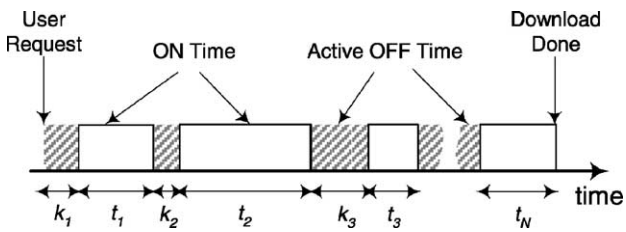

Fig. 2. The active phase as a sequence of ON and OFF Times.

the page textual information, the names of the embedded files and a description of the page layout. The browser transfers the whole set of files and arranges them in the page.

The active phase can be seen as a sequence of N ON Times (t_i in Fig. 2) and N Active OFF Times (k_i in Fig. 2), where N is a random variable. The main file is transferred during the first ON Time. Then, the transfer of each embedded file occurs in subsequent ON times. ON times are usually separated by OFF times. Among the others, an OFF Time includes the time required by the client to prepare HTTP request(s). These OFF Times are typically referred to as *Active OFF Times*, to distinguish them from the User Think Times.

The length of a single ON Time can be described as follows:

$$t_i = \frac{B_i}{\gamma_i} + \delta_i = \frac{D_i + h_i}{\gamma_i} + \delta_i, \tag{2}$$

where B_i is the size (in bytes) of an overall HTTP transaction needed to fetch a file. Specifically, it includes the file size (D_i), and the headers of all packets containing the HTTP request(s) and HTTP response (h_i). γ_i is the throughput experienced during this transaction. δ_i depends on the specific HTTP version, and may include the sum of the network Round Trip Time (RTT), and the time needed by the Web server to process an HTTP request.

It must be pointed out that γ_i and δ_i depend on the network traffic conditions, while h_i can be closely approximated with a constant value. D_i depends on the distribution of the Web file sizes. In the literature, the file size is modeled according to a hybrid distribution [33,35,37]: the tail and the body are modeled according to Pareto (see Eq. (1)) and log-normal distributions (see Eq. (3)), respectively:

$$p(x) = \frac{1}{\alpha x\sqrt{2\pi}}\, \mathrm{e}^{-(\ln x-\mu)^2/2\sigma^2}. \tag{3}$$

The parameters of log-normal (i.e., μ and σ) and Pareto (i.e., α and k) distributions, as well as the cutoff value between the two distributions, depend on the set of files available at the Web server.

Active OFF Times (k_i in Fig. 2), are typically modeled according to a Weibull distribution:

$$p(t) = \frac{bt^{b-1}}{a^b}\, \mathrm{e}^{-(t/a)^b}. \tag{4}$$

The Weibull parameters do not depend on the particular Web site. Typical values are $a = 1.46$, $b = 0.382$ [35].

Finally, N denotes the number of Active OFF and ON Times (see Fig. 2). Obviously, $N = 1 + e$, where e is the number of embedded files, and 1 corresponds to the main file. The number of embedded files, e, is typically modeled according to a Pareto distribution, where α and k parameters depend on the specific Web server [35].

As a final remark, it has been shown in [34] that the Web-page download statistical models are strictly related to the self-similarity property of Web traffic. Since this is a *structural* property of the Web traffic, the characterization provided in this section do not depend either on the Web contents or on the user access patterns.

2.2. Energy-consumption model

In this section we introduce a model for the energy consumption in a mobile Web access scenario. As explained in Section 1, the energy consumption is approximately proportional to the time during which the

Table 1
Summary of the symbols used throughout the paper

N	Number of files in a Web page
k_i	Length of the ith Active OFF Time inside the active phase
t_i	Length of the ith ON Time inside the active phase
t_{UTT}	Length of the User Think Time after the download of a Web page
B_i	Dimension of the HTTP transaction needed to fetch the ith file of a Web page
D_i	Dimension of the ith file of a Web page
h_i	Dimension of the HTTP request and response headers used in the ith HTTP transaction
γ_i	Average throughput experienced by the mobile host during the ith HTTP transaction
γ	Maximum throughput available on the wireless link
δ_i	Sum of the network Round Trip Time and the time needed by the Web server to process the ith HTTP request
β_i	Overhead in bytes introduced in the ith file download
τ_i	Overhead in time introduced in the ith file download
A	Active OFF Times contribution to the energy spent to download a Web page
U	User Think Time contribution to the energy spent to download a Web page
m	Number of times the mobile host wireless interface switches from off to on during the active phase
t_{so}	Time interval needed by the mobile host wireless interface to switch from off to on
g	Number of residual transfer-time estimates provided by the access point during the active phase of a Web-page download
s	Number of residual transfer-time estimates greater than t_{so} ($s \leq g$)

wireless interface remains in the ON state. Therefore, hereafter we will measure the energy consumption as the wireless interface ON time. Eq. (5) provides the energy, C, consumed for downloading a Web page:

$$C = \sum_{i=1}^{N} \left(\frac{D_i + \beta_i}{\gamma_i} + \tau_i \right) + A + U + m \cdot t_{\mathrm{so}}, \tag{5}$$

where β_i measures the overhead in bytes introduced in the ith file download. In addition to the size of the HTTP request and response headers (h_i in Eq. (2)), β_i also includes specific overheads associated with the implemented power-saving strategy (if any). γ_i is the throughput experienced in the file transfer (see Eq. (2)). τ_i is the overhead in time related to the download of the ith file. Specifically, in addition to δ_i (see Eq. (2)), it also includes specific time overheads associated with the implemented power-saving strategy (if any). A is the contribution to the energy consumption due to the Active OFF Times. This contribution is the sum of the Active OFF Times ($A = \sum_{i=1}^{N} k_i$) if no power-saving strategy is implemented. Power-saving strategies typically reduce this quantity. U is the contribution to the energy consumption due to a User Think Time. This exactly corresponds to the User Think Time if no power-saving strategy is implemented. The aim of power-saving strategies is to reduce it. $m \cdot t_{\mathrm{so}}$ is the total contribution to the energy consumption due to the transients caused by the off–on switching of the wireless interface. When the wireless interface is turned on, there is a transient period during which it consumes energy but it cannot be used for data transfer. In Eq. (5) t_{so} denotes the length of the transient period (typically, and throughout this work, 100 ms), while m is the number of off–on transitions during the Web-page transfer.[1] D_i and N define the traffic characteristics (see Table 1) and do not depend on the particular power-saving strategy.

[1] The m value depends on the specific power-saving strategy. Obviously, when no power-saving strategy is implemented, $m = 0$.

For reader convenience, in Table 1 we summarize the symbols that are used throughout the paper.

3. Power-saving strategies for mobile Web access

A typical mobile Internet scenario is depicted in Fig. 3. The communication between a mobile host and a host connected to the Internet (fixed host) is made possible by a third entity (access point) which provides Internet connectivity to the mobile host through a wireless link.

This scenario is becoming more and more relevant with the emerging of the Wi-Fi hotspot business. A hotspot is a critical business area (e.g., airports, stations, hotels) characterized by a set of access points where users can have a broadband access to the Internet by subscribing a contract with a hotspot operator, or a wireless Internet service provider.

In this scenario the legacy TCP/IP protocol stack is typically implemented in the mobile device, and no power-saving strategy is used. Therefore, Eq. (5) instantiates as follows:

$$C_{\mathrm{TCP}} = \sum_{i=1}^{N} \left(\frac{D_i + h_i}{\gamma_i(\mathrm{TCP})} + \delta_i(\mathrm{TCP}) \right) + \sum_{i=1}^{N} k_i + t_{\mathrm{UTT}}, \tag{6}$$

where $\beta_i = h_i$, $A = \sum_{i=1}^{N} k_i$, $U = t_{\mathrm{UTT}}$ and $m = 0$, since no power-saving strategy is used. $\delta_i(\mathrm{TCP})$ represents the δ_i term of Eq. (2) when the legacy TCP/IP architecture is used, and $\gamma_i(\mathrm{TCP})$ is the throughput experienced during this transaction.

Several factors contribute to make the legacy TCP/IP approach inefficient from the power-saving standpoint: $\gamma_i(\mathrm{TCP})$ is typically very low due to the interaction between the wired and wireless environments [23,29,38], U corresponds to the whole User Think Time ($U = t_{\mathrm{UTT}}$), and A is the sum of the Active OFF Times ($A = \sum_{i=1}^{N} k_i$). Therefore, C_{TCP} represents the upper bound for the energy consumption. On the contrary, the ideal strategy introduced in the next section represents the lower bound for the energy consumption.

3.1. Ideal strategy

The minimum possible energy spent for a Web-page download is obtained by assuming that the transfer from the access point to the mobile host is performed in a single phase. Specifically, the wireless interface is turned on, all data are transferred at the maximum throughput allowed by the wireless link, γ, and then the wireless interface remains off until the next Web-page download. Hence, the wireless interface remains on for the minimum amount of time. Accordingly, the ideal energy consumption is given by

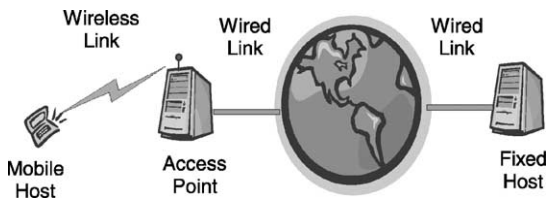

Fig. 3. A typical mobile environment.

Eq. (7):

$$C_{\text{ideal}} = \sum_{i=1}^{N} \left(\frac{D_i + h_i}{\gamma} \right) + t_{\text{so}}. \tag{7}$$

Eq. (7) is immediately obtained from Eq. (5) by considering that

- β_i is equal to h_i;
- γ_i is constantly equal to γ;
- there is no temporal overhead related to the HTTP transaction ($\tau_i = 0$);
- the Active OFF Times and User Think Time contributions are 0 ($A = U = 0$);
- the wireless interface is turned on only once for each Web-page download ($m = 1$).

It is worthwhile to point out that, even though the ideal strategy is unfeasible, C_{ideal} represents a lower bound for any other feasible power-saving strategy. In the next few sections we introduce three feasible power-saving strategies, and compare their performance with the ideal case.

3.2. I-TCP strategy

The I-TCP approach [30] splits the TCP connection (between the mobile client and the remote Web server) in two TCP connections. The former one operates between the mobile client and the access point, while the latter one connects the access point to the Web server. This allows to decouple the wireless and the wired environments. Hence, the I-TCP approach increases the end-to-end throughput [30], and indirectly contributes to reduce the power consumption. This effect is pointed out by Eq. (8) that defines the energy consumption related to the I-TCP strategy:

$$C_{\text{I-TCP}} = \sum_{i=1}^{N} \left(\frac{D_i + h_i}{\gamma_i(\text{I-TCP})} + \delta_i(\text{I-TCP}) \right) + \sum_{i=1}^{N} k_i + t_{\text{UTT}}. \tag{8}$$

The only difference between Eqs. (8) and (6) is γ_i(I-TCP) instead of γ_i(TCP), and δ_i(I-TCP) instead of δ_i(TCP). By considering that $\delta_i(\text{I-TCP}) \approx \delta_i(\text{TCP})$[2] and that the I-TCP approach generally results in an increased throughput (i.e., $\gamma_i(\text{I-TCP}) > \gamma_i(\text{TCP})$), Eq. (8) indicates that $C_{\text{I-TCP}} < C_{\text{TCP}}$.

A bare I-TCP strategy only provides energy saving as a side effect, since it is not essentially aimed at minimizing energy consumption. In particular, this strategy does not provide any contribution to reduce the second and third terms of (6), i.e., A and U. These terms (mainly U) heavily contribute to the energy consumption, since they represent the contributions of idle phases to the energy consumption. To reduce their impact, the wireless interface should remain off as long as possible during idle phases, and hence we expect that a pure I-TCP approach performs poorly. Nevertheless, it constitutes a reference architecture for more efficient strategies. Specifically, in the following we present two strategies that enhance the I-TCP approach. The former one minimizes the contribution of the User Think Time (U) to the energy consumption while the latter one attempts to minimize both A and U. The first strategy only requires information that are local to the mobile host, and hence will be referred throughout as *local strategy*. On the other hand, the second strategy needs a global overview of the system, and will thus be referred to as *global strategy*.

[2] The only difference is the additional processing time at the access point in the I-TCP approach.

3.3. Local strategy

This strategy is *local* in the sense that the wireless interface switching-off decision is taken utilizing only local information. Specifically, the mobile host turns off the wireless interface during the User Think Time. This strategy is very simple to implement. It only requires that the wireless interface is switched off when the active phase is finished, and turned on again upon receiving a new request from the mobile user. As this strategy does not modify the I-TCP behavior during the active phase, the first and second terms of (8) remain unchanged. Furthermore, it eliminates the inactive phase contribution (i.e., $U = 0$), and the wireless interface is switched on just once for each Web-page download (i.e., $m = 1$). Hence, the energy consumption using the local strategy is

$$C_{\text{local}} = \sum_{i=1}^{N} \left(\frac{D_i + h_i}{\gamma_i(\text{I-TCP})} + \delta_i(\text{I-TCP}) \right) + \sum_{i=1}^{N} k_i + t_{\text{so}}. \tag{9}$$

Eq. (9) is obtained from (5) by setting $\beta_i = h_i$, $\tau_i = \delta_i(\text{I-TCP})$, $A = \sum_{i=1}^{N} k_i$, $U = 0$ and $m = 1$.

Since the User Think Time contribution is typically heavy, the energy saving provided by this strategy is expected to be significant. In the next section we will investigate how to further increase the energy saving by switching off the wireless interface even during the active phase.

3.4. Global strategy

The global strategy attempts to approach the ideal energy consumption by exploiting the knowledge of Web traffic statistics (see Section 2.1). The global strategy borrows from the local one the idea of switching off the wireless interface during the inactive phase (i.e., $U = 0$). Moreover, it uses some mechanisms to reduce the energy consumption during the active phase, as well.

In the environment depicted in Fig. 3, the bottleneck between the mobile host and the fixed host is usually represented by the *wired* part of the path and the Web server speed [23,29]. As a consequence, with the previous strategies (see Sections 3.2 and 3.3), the wireless link cannot ever be used at its full available throughput while downloading the files. This problem is tackled in the global strategy by *pre-fetching* the Web page at the access point. Specifically, a software agent at the access point, throughout referred to as the *PS-Daemon*, acts similarly as a Web proxy (a Web proxy with power-saving functionalities), by acting on behalf of the client browser. While the HTML main file is flowing on the wired network, the mobile host maintains the wireless interface switched off. This switch-off time is based on an estimate (provided by the PS-Daemon) of the main file transfer-time.[3] The PS-Daemon stores the main file, and automatically pre-fetches the embedded files, *if any*, from the Web server. When the mobile host reconnects, the PS-Daemon delivers all the available data (i.e., the main file and the embedded files, if available). If more data still need to be downloaded, the PS-Daemon also provides an estimate of the residual transfer-time. If this estimate is greater than t_{so}, the mobile host switches off the wireless interface and reconnects later according to the estimate. More details can be found in [29].

We are now in the position to prove the following proposition.

[3] The mobile host switches the wireless interface off only if the estimated transfer-time is greater than t_{so}.

Proposition 1. *In a system that adopts the global strategy, the energy consumption is*

$$C_{\text{global}} = \sum_{i=1}^{N} \left(\frac{D_i + \beta_i}{\gamma} \right) + A + (s+1) \cdot t_{\text{so}}, \tag{10}$$

where $A \leq (g - s) \cdot t_{\text{so}}$ *and* $\beta_i = h_i + p_i$.

Proof. For ease of reading, we move the proof to Appendix A. □

As shown in the proof of the above proposition (see Appendix A), the global strategy relies upon the algorithm used for estimating the residual transfer-time. Specifically, it requires the estimate of the residual transfer-time for both the HTML main file and the embedded files. The following propositions provide closed formulas for these quantities.

Proposition 2. *By denoting with* est_{m} *the residual transfer-time for the HTML main file, the following equation holds*:

$$\text{est}_{\text{m}} = \begin{cases} \text{RTT}, & \textit{if a connection is available}, \\ 2\text{RTT}, & \textit{otherwise}. \end{cases} \tag{11}$$

Proof. est_{m} can be evaluated by assuming the knowledge of the RTT between the PS-Daemon and the Web server. When the PS-Daemon receives a request from the mobile host, it establishes a TCP connection with the server, or it uses an already opened persistent connection. In the first case, the retrieval of the main file requires, at least, two RTTs (three-way handshake plus HTTP request-response). In the second case (persistent TCP connection), a single RTT may be enough (if the main file fits into a single TCP's window size). □

Proposition 3. *By denoting with* est_{e} *the estimate of the residual transfer-time for the embedded files, the following equation holds*:

$$\text{est}_{\text{e}} = \text{RTT}\, u, \tag{12}$$

where u is the minimum number of RTTs *necessary to transfer all the embedded files on a TCP connection*.

Proof. The residual transfer-time estimate for the embedded files exploits some information contained in the main file, that are already downloaded when the embedded files are requested. Hence, the PS-Daemon knows the number, e, of embedded files that compose the Web page.[4] The total number of bytes to be transferred (throughout referred to as B), can be estimated as follows:

$$\hat{B} = \sum_{i=1}^{e} (\tilde{D}_i + h_i), \tag{13}$$

[4] Throughout the analysis, we assume that all the embedded files reside on the same server of the main file. Very similar mechanisms (although slightly more complex) can be used when the embedded files reside on different servers.

where $\hat{B}$ is the estimate of B, $\tilde{D}_i$ the estimate for the ith embedded file size (i.e., it is a sample from the distribution defined in [35]), and h_i the dimension of the HTTP headers used for downloading the ith embedded file.

The distribution parameters of embedded file sizes may vary with the Web servers' content. For this reason they should be communicated by the Web server to the PS-Daemon. This is complex and unrealistic. This complexity can be avoided by using average values only. Accordingly, $\hat{B}$ becomes

$$\hat{B} = e(\bar{D} + \bar{h}), \tag{14}$$

where $\bar{h}$ is the average of h_i, and $\bar{D}$ the average of $\tilde{D}_i$.

Finally, the residual transfer-time estimate can be evaluated by using $\hat{B}$ and an estimate of RTT. Specifically:

$$\mathrm{est_e} = \mathrm{RTT}\, u, \tag{15}$$

where u is the minimum number of RTTs necessary to transfer the $\hat{B}$ bytes on a TCP connection, given the connection state.

The complete algorithm to evaluate u is presented in [40], and is omitted here due to space reasons.

The estimators of the residual transfer-time require the RTT knowledge. If the PS-Daemon has no information about the RTT, it uses some initial value (as TCP does [39]). □

4. Experimental test-bed

The main objective of our experimental study is to evaluate the power-saving performance of the strategies presented in this paper through an extensive set of measurements on a real Internet test-bed. To this end, we implemented the local and global strategies on top of an I-TCP architecture [30,31]. In this section we present the performance figures that we intend to investigate, and the characteristics of our test-bed.

4.1. Performance indexes

We evaluate the local and the global strategies in terms of *energy consumption* with respect to a *reference I-TCP architecture*. Specifically, in the reference I-TCP architecture we assume on the wireless link a transport protocol optimized for the wireless link characteristics, instead of the legacy TCP protocol. This is a light protocol that only implements mechanisms for error detection and recovery and does not include any congestion control mechanism (see the STP protocol in [29]).

Hence, an important performance measure is the *power-saving index*, defined as

$$I_{\mathrm{ps}} = \frac{C_{\text{power-saving architecture}}}{C_{\text{I-TCP}}}, \tag{16}$$

where $C_{\text{I-TCP}}$ comes from (8) and $C_{\text{power-saving architecture}}$ is one among C_{ideal}, C_{local} and C_{global} from (7) to (10).

I_{ps} is the energy consumption of a specific power-saving strategy expressed as a percentage of the energy consumption of the reference architecture. As it will be explained later, our experimental test-bed

guarantees that values used to compute I_{ps} are measured under the same system conditions (Web server and network traffic conditions).

In addition, we compare the performance of the local and global strategies with those of the ideal one to understand how well feasible strategies approximate the ideal case.

Although power saving is the key factor to evaluate the proposed strategies, we are also interested to analyze the impact of these strategies on the QoS perceived by the user. Hereafter, we use the URT (i.e., the time interval elapsed from the user request till the rendering of the related Web page) as the main QoS index for a Web service. It is worth noting that the global strategy may introduce an additional delay in the Web-page transfer-time. Additional delays occur whenever a residual transfer-time estimate is longer than the real value. To take into consideration this aspect, we introduce the *page delay index* defined as

$$I_{pd} = \text{page transfer-time with global strategy} - \text{page transfer-time with I-TCP strategy}. \tag{17}$$

I_{pd} measures the additional URT delay introduced by the global strategy.

4.2. The test-bed

In our test-bed we use a real Web server located at the University of Texas at Arlington, while the mobile host (and the access point) is located at the CNR in Pisa (Italy). This allows us to evaluate the power-saving strategies over a real, congested, intercontinental path. As far as the wireless link, we adopt the Wi-Fi technology with transmission speed ranging from 2 to 11 Mbps.

At the mobile host we use SURGE to simulate a Web client [32,34,35]. SURGE reproduces the statistical user model presented in Section 2.1. Specifically, SURGE operates in two steps. During the first step, it defines the set of files to be stored in the Web server, guaranteeing that file sizes are distributed as shown in Section 2.1. Moreover, SURGE defines the structure of the Web pages by building groups from the above files.

In the second step, SURGE defines the sequence of client requests to the Web server. To this end it creates

- a trace of Web-page requests to be issued to the server (Active Time trace);
- a trace of Inactive OFF Times.

The traffic generated by using the Active Time and Inactive OFF Time traces meets the statistical characterization given in Section 2.1.

During the experiments, the client requests and User Think Times are extracted from the above traces. Specifically, a client picks up a Web-page request from the Active Time trace and downloads the corresponding page, then picks up a value from the Inactive OFF Time trace, waits for this time interval, and extracts the next Web-page request.

4.3. Experiment methodology

To test our power-saving strategies we ran an extensive set of experiments. In each experiment we have two instances of the same “SURGE client”. The two instances download, in parallel, the same set of Web pages by using the pure I-TCP architecture and the selected power-saving strategy, respectively. This guarantees that the two sets of parallel downloads are performed under the same system

conditions.[5] For each page download we log the URT value, the total length (in bytes) of the Web transaction (i.e., the sum of the page dimension and the HTTP headers dimension), and the network energy consumption (i.e., the total amount of time during which the wireless interface is turned on). From each experiment, we compute the I_{ps} index for the selected strategy (global, local and ideal). Moreover, for each Web-page download performed with the global strategy we evaluate the I_{pd} index.

A final remark is necessary about the length of each experiment. The Web characterization given in Section 2.1 shows that the file size can be modeled according to an hybrid distribution: log-normal in the body and Pareto in the tail. We choose the experiments' length to have—in average—for each experiment at least 10 files' requests coming from the tail of the distribution.[6] From the SURGE documentation, 93% of the requested files comes from the body, while 7% comes from the tail. Therefore, to have (in average) 10 "long" files (i.e., with size belonging to the tail of the distribution), the minimum number of files to be transferred in each experiment is 143. Hence, we decide to stop each experiment after downloading 150 files.[7]

We replicated the experiments sequentially throughout an entire working day. To achieve independent experiments, we modified SURGE in such a way that it can start from a specific point in the page requests trace. Exploiting this feature, each experiment starts requesting the trace item next to the last one used in the previous experiment, and hence I_{ps} and I_{pd} samples from different experiments are independent.

It must be noted that an entire working day of experiments is not sufficient to exhaust the whole trace of Web-page requests. Finally, we replicated an entire day of experiments for 10 working days.[8]

5. Tuning of the experiments

In this section we present some preliminary results collected in the experiments of a single day. These results are used to tune our measurement methodology.

5.1. Comparison between the embedded file size estimators

In Section 3.4 we described two estimators of the total embedded file size (see Eqs. (13) and (14)). The first one uses the file size distribution, while the second one relies upon the average value only. To compare these estimators, we ran two sets of 10 consecutive experiments. The two sets of experiments were performed by downloading the same set of Web pages. Furthermore, we verified that all experiments were performed under the same network conditions. For each experiment, we measure the I_{ps} index for the global strategy. Moreover, for each page, we evaluate the I_{pd} index, and we average the I_{pd} values on the whole 150-file experiment. Hereafter, $\bar{I}_{pd}$ denotes the averaged value. As the experiments are independent

[5] One of the two users may experience some advantages due to the Web server caching. Specifically, if a user requests the pages immediately before the other one, the latter can find the pages in the server's file-system cache, and hence it can experience a lower URT. To overcome this asymmetric behavior, the second user starts 30 s after the first one; moreover, the user that starts first in an experiment will start as the second in the next one.

[6] This constraint ensures that results are not biased by a particular choice in the file size dimensions.

[7] The experiment is stopped when the Web page "on-the-fly" is completely transferred.

[8] The first experiment of a new day begins with the Web-page request successive (in the trace) to the last one used in the previous day. After the trace is exhausted, SURGE wraps-around and requests the first item.

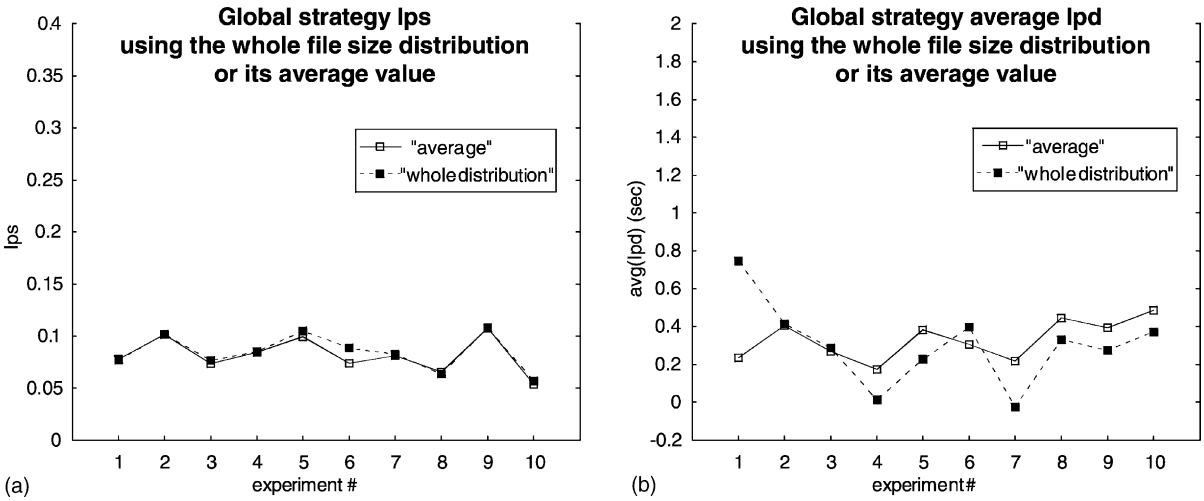

Fig. 4. Embedded files size estimators comparison.

and under the same network conditions, the I_{ps} and $\bar{I}_{pd}$ samples are i.i.d. The data obtained with the two estimators are presented in Fig. 4 and Table 2.

From Fig. 4 it appears that the two estimators provide almost identical results in terms of I_{ps} (see part (a)). In terms of $\bar{I}_{pd}$, the estimator based on the average value appears to be more stable (see part (b)).

These results are confirmed by Table 2 where we report the confidence intervals for I_{ps} and $\bar{I}_{pd}$ (hereafter, the confidence level is 95%). Hence, we can conclude that it is convenient to use the estimator based on the average values.

5.2. Performance over a single day

To give an idea of the power-saving performance of each strategy, we show some snapshots taken from the experiments of 1 day.

Fig. 5(a) shows the plots of C for all strategies under investigation. As expected, the pure I-TCP approach provides the highest power consumption. The gap between $C_{\text{I-TCP}}$ and C_{local} is significant, and this confirms that the User Think Time provides a big contribution to the total energy consumption. In Fig. 5(b) we report the same plots on a different time scale to emphasize the differences among the three strategies. These differences are due to the strategies' behavior during the active phase. As C_{local} is significantly higher than C_{global}, it follows that a wise energy management during the active phase can produce relevant energy savings. Obviously, the ideal strategy is the best one, but the global strategy well approximates the ideal behavior. This observation is confirmed and quantified in Table 3.

Table 2
I_{ps} and $\bar{I}_{pd}$ average values and confidence intervals

	Avg(I_{ps})	Avg($\bar{I}_{pd}$) (s)
Distribution	0.084 ± 0.012	0.303 ± 0.154
Average value	0.081 ± 0.012	0.332 ± 0.075

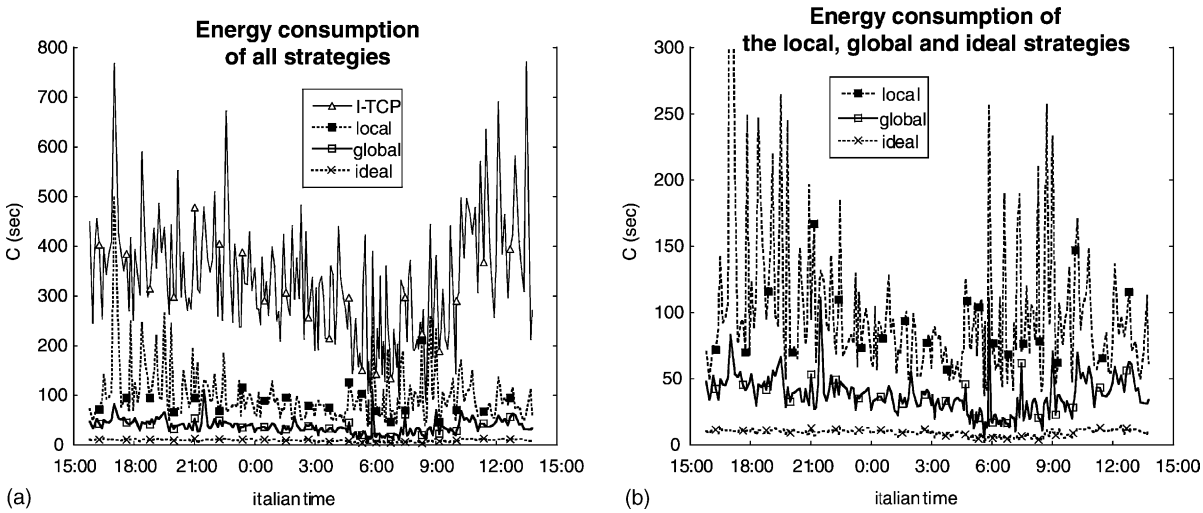

Fig. 5. An experiment day: the energy consumption of the different strategies.

This table shows the values of I_{ps} and $\bar{I}_{pd}$ averaged over the whole experiment day. As it clearly appears from Table 3, the global strategy can save 89% of the energy with respect to the I-TCP approach, and it outperforms the local strategy of more than 50%. Furthermore, the global strategy does not introduce a significant QoS degradation. In detail, this strategy increases (in average) the Web-page download time of 0.24 s that is almost negligible for a Web user.

5.3. Data aggregation

From the plots in Fig. 5 it appears that the energy consumption values (except for the ideal case) are extremely variable during the day. This is mainly due to the variable conditions of the Internet path from Pisa to Arlington. This is confirmed by Fig. 6, that reports the throughput, measured at the application level, averaged on 1 h intervals.

As expected, the throughput varies during the day. Specifically, we can observe that, from the throughput standpoint, a working day can be subdivided into several classes that depend on the status of the Internet in Europe and USA. For instance, in the period 2–7 P.M., we have the minimum throughput due to the overlapping between Europe and USA business hours. On the other hand, in the period 5–9 A.M., we observe the highest throughput due to the overlap of non-business hours in Europe and USA. Furthermore, by repeating the same analysis for several working days we observed the same behavior. Thus, hereafter we can assume that experiments performed within the same hour, even in different days, are identically distributed.

Table 3
I_{ps} and $\bar{I}_{pd}$ average values for the plots in Fig. 5

Strategy	Avg(I_{ps})	Avg($\bar{I}_{pd}$) (s)
Local	0.23	–
Global	0.11	0.24
Ideal	0.03	–

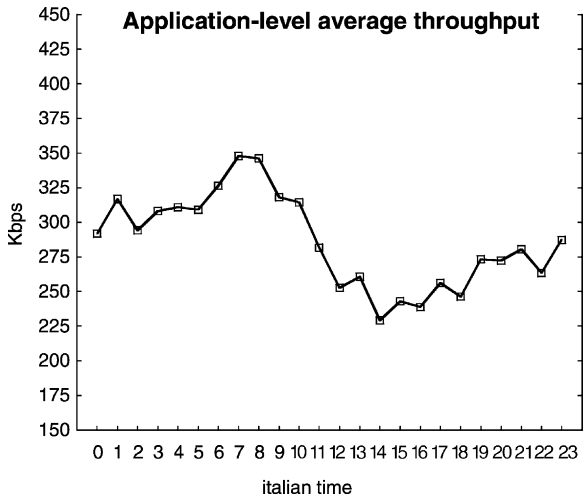

Fig. 6. The average throughput measured at the application level as a function of the day-time.

Based on the above observations we define our data aggregation method as follows. From each 150-file experiment we derive an observation for I_{ps} and one for $\bar{I}_{\text{pd}}$. Our experiments are continuously performed for a whole day and repeated for 10 working days. Samples obtained at the same hour (also in different working days) are i.i.d., and hence from these samples we can compute the hourly confidence intervals of I_{ps} and $\bar{I}_{\text{pd}}$, according to the classical statistical method [41].

6. Performance evaluation

In this section we deepen the previous analysis by providing, for all strategies, accurate estimates of the confidence intervals of I_{ps} and $\bar{I}_{\text{pd}}$ indexes.

6.1. I_{ps} analysis

Fig. 7 shows the I_{ps} index for the local, global and ideal strategies.

The results confirm our preliminary observations. Specifically, by eliminating the power consumption during User Think Times it is possible to achieve a significant energy saving. The local strategy saves about 76% of the I-TCP energy consumption. Moreover, these results also confirm the relevance of the energy management during the active phase. The global strategy saves approximately 88% of the I-TCP energy consumption, and therefore significantly improves the local strategy performance.

Fig. 8 compares the local, global and ideal strategies in more detail. Specifically, plot (a) indicates that, with respect to the local strategy, the global strategy saves 26% more energy in the worst case, 45% on the average, and up to 53% in the best case. Plot (b) shows the performance of the global strategy with respect to the ideal case. On average, the saving of the global strategy is approximately 25% of that achievable in the ideal strategy. It is worth noting that, even if not reported here for space reasons, the local and the I-TCP strategies are very far from the ideal case. Specifically, $C_{\text{local}} = 7 \cdot C_{\text{ideal}}$ and $C_{\text{I-TCP}} = 32 \cdot C_{\text{ideal}}$.

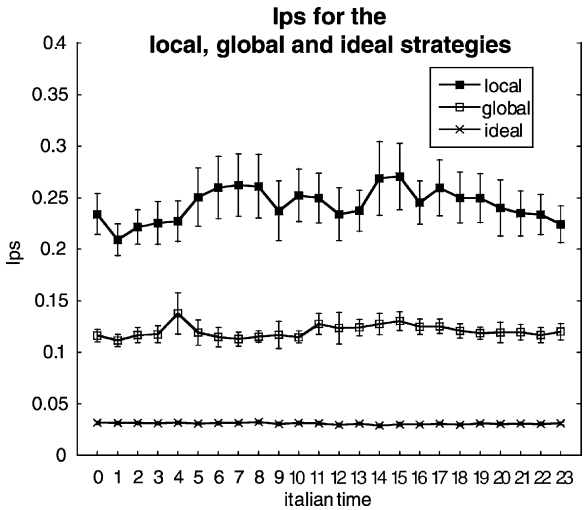

Fig. 7. Power-saving performance of the local, global and ideal strategies.

To summarize, the global strategy is the best approximation of the ideal, *unfeasible*, solution. Therefore, in the following we will focus on the global strategy only. First of all, we analyze the behavior of the I_{ps} index with respect to the throughput on the Internet (γ_i in (8)). To this end we aggregate the samples in three classes of throughput, and we average the samples belonging to the same class, taking the throughput central value as representative of the entire class. More precisely, classes are made up of samples that experienced a throughput below 150 Kbps, between 150 and 300 Kbps, and between 300 and 600 Kbps, respectively.

Fig. 9(a) shows that I_{ps} is not very sensitive to the throughput variation. However, it is slightly higher when the Internet throughput is low. This can be easily explained by recalling the residual transfer-time

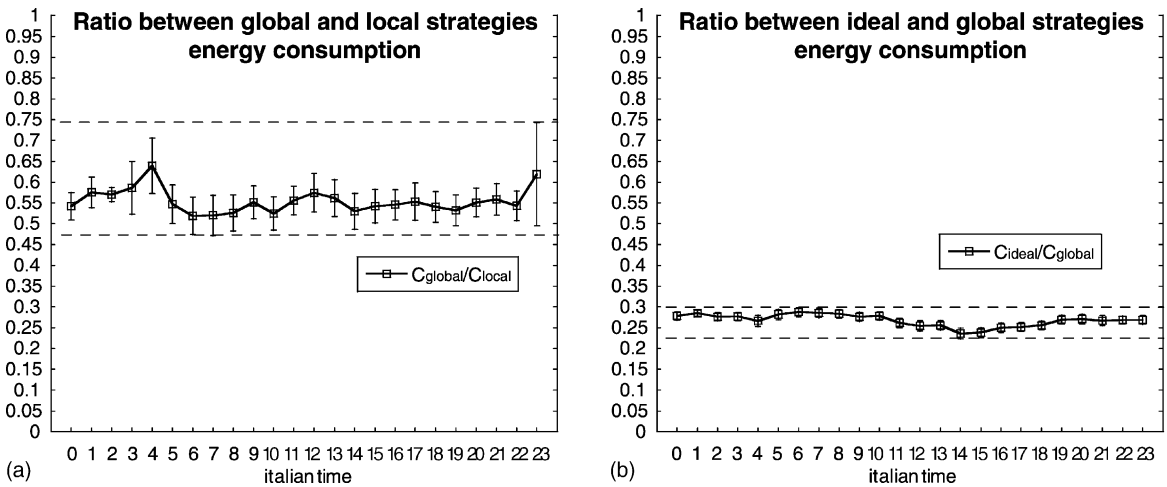

Fig. 8. Comparison between the local, global and ideal strategies.

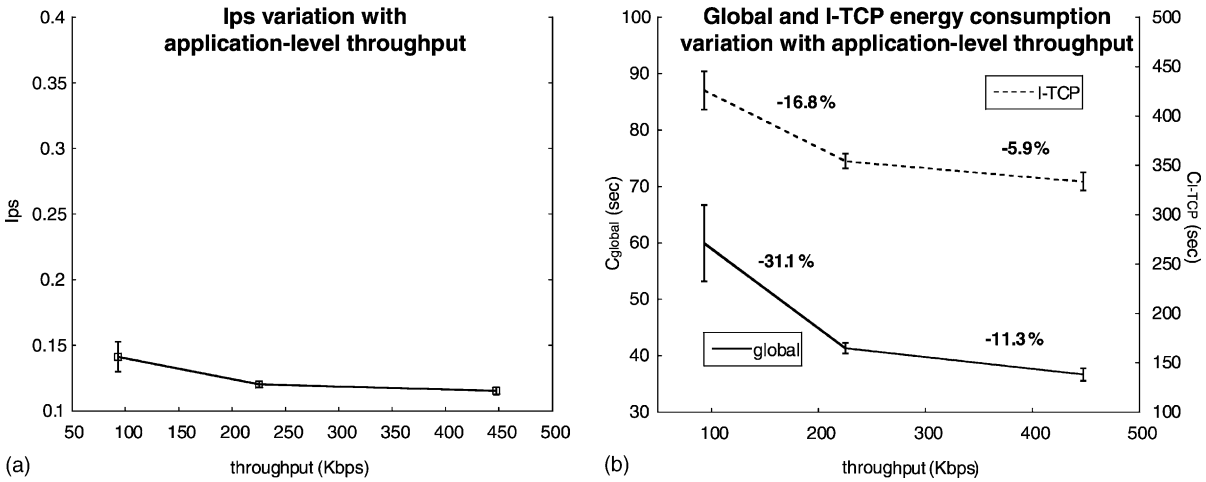

Fig. 9. Analysis of the global strategy I_{ps} as a function of the Internet throughput.

estimator algorithm. The time interval evaluated by (15) is the *minimum* time to transfer the estimated number of bytes [40]. The choice of the minimum time interval reduces the QoS degradation, but slightly increases the energy consumption when the Internet throughput is low. In this case, the mobile host will need more time to complete the data transfer and, hence, it will switch on the wireless interface several times. This behavior is highlighted by Fig. 9(b). By moving from the first class to the second one—due to the γ_i increase (see (8))—C_{global} decreases of more than 30%, while $C_{\text{I-TCP}}$ reduces of 17%. The same trend also occurs in the transition from the second to the third class but the difference is less marked.

Finally, in Fig. 10 we show the dependence of I_{ps} on the *wireless* link throughput. We ran 2 sets of 10 experiments by varying the speed of the wireless link from 11 to 2 Mbps. With the current Internet

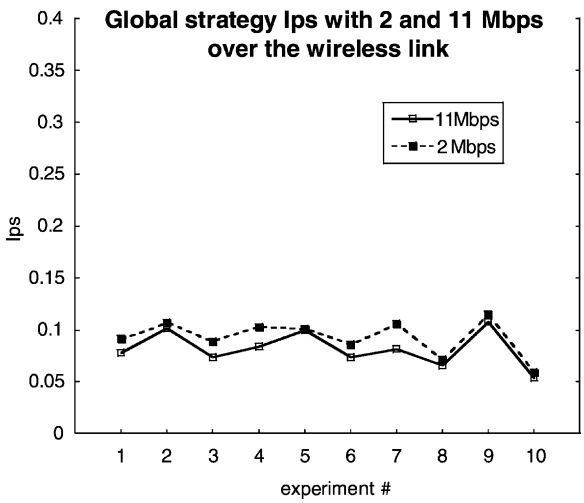

Fig. 10. I_{ps} behavior with varying wireless link throughput.

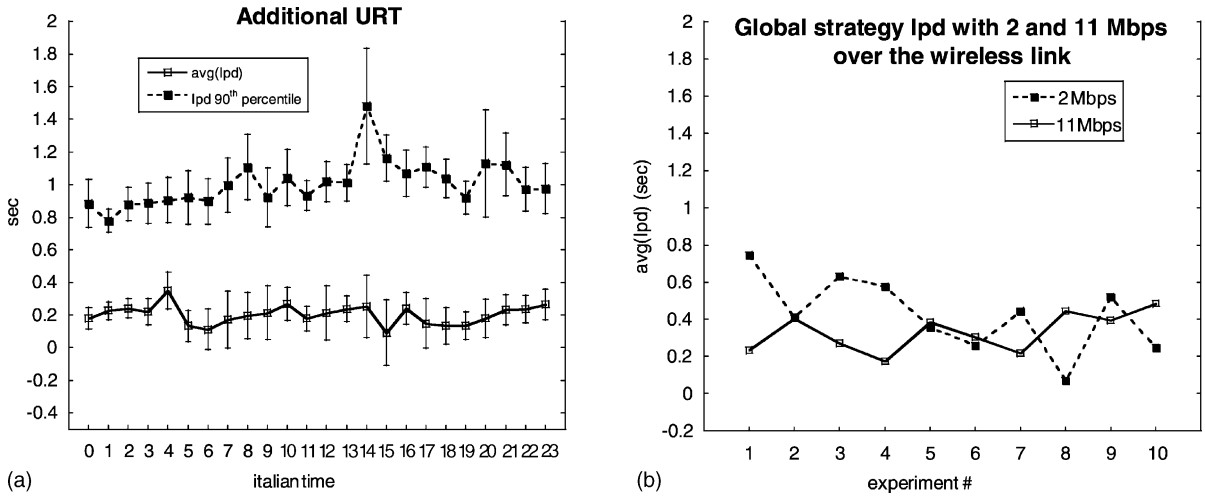

Fig. 11. $\bar{I}_{pd}$ and I_{pd} 90th percentile average values (a); $\bar{I}_{pd}$ values for different wireless throughputs (b).

technologies, we expect that the wired Internet remains the bottleneck also when we use a 2 Mbps WLAN. As it is clear from (10) and (8), when the wireless link throughput decreases, C_{global} increases, while $C_{I\text{-}TCP}$ does not change significantly because it is mainly affected by the wired Internet. However, the results presented in Fig. 10 show that the global strategy exhibits a small sensitiveness to the throughput of wireless link. By decreasing the wireless speed from 11 to 2 Mbps, I_{ps} experience (on average) a 13% increase only.

6.2. I_{pd} analysis

To complete our analysis, we investigate the QoS degradation introduced by the global strategy by studying the I_{pd} index. Specifically, from each experiment we compute the average value of I_{pd} (i.e., $\bar{I}_{pd}$), and its 90th percentile. Then, we average the samples taken within the same hour. Finally, we compute the confidence intervals of the two figures according to the method of Section 5.3. The results obtained are shown in Fig. 11.

From Fig. 11(a), it can be noted that the additional URT introduced by the global strategy does not degrade significantly the QoS perceived by the Web user: the global strategy increases the URT of about 0.2 s on average, and no more than 1.8 s in the 90% of cases.

The analysis of the $\bar{I}_{pd}$ sensitiveness to the wireless link speed confirms the observation done in the I_{ps} analysis (see Fig. 10). As shown in Fig. 11(b), by decreasing the wireless link speed from 11 to 2 Mbps the $\bar{I}_{pd}$ index experiences (on average) a 28% increase.

7. Summary and conclusions

In this work we evaluate the effectiveness of new strategies for reducing the power consumption in mobile Web access. We focus on a Wi-Fi hotspot scenario. This is emerging as the most relevant wireless

Internet scenario and it is interesting from a power-saving point of view, because the bottleneck between the mobile host and the Web server is located on the wired part of the Internet.

Our study starts from the analysis of the impact, on the power consumption, of the different phases of a Web transaction. To this end we characterize the Web transaction phases through statistical distributions (taken from real Web traffic traces), and construct an energy-consumption model. This model highlights possible directions for reducing the energy consumption. In this paper we identify four different power-saving strategies.

We start with a strategy named *ideal strategy*. This strategy guarantees the minimum energy consumption to download a Web page. It is unfeasible but provides a reference for the other strategies we develop. The first feasible strategy we envision is based on a pure I-TCP architecture. The advantage of this strategy is related to the throughput increase (with respect to the legacy TCP/IP architecture) that, indirectly, produces power saving. The second strategy, named *local strategy*, explicitly addresses the power saving. Specifically, by exploiting the semantic of the Web application, it eliminates the waste of power due to User Think Times. The local strategy is further refined by the *global strategy* that performs energy management also during the active phases of a Web transaction. For all these strategies we derive an analytical model of their energy consumption, and we compare their performance through extensive measurements.

The comparison shows that the local strategy saves, on average, 76% of the energy drained by a pure I-TCP solution. The global strategy outperforms the local one by reducing the I-TCP energy consumption of about 88% on average. In addition, with respect to the ideal strategy, the global strategy consumes less than 4 times, while the local and the I-TCP strategies consume 7 and 32 times, respectively. Therefore, among the analyzed strategies, the global one is the best approximation of the ideal—but unfeasible—case. Furthermore, the global strategy introduces a negligible degradation in the QoS perceived by the users. Specifically, the additional URT introduced by the global strategy is about 0.2 s on average, and is below 1.8 s in 90% of cases. Finally, a sensitiveness analysis shows that the performance of the global strategy is almost independent from the throughput of the wireless link, provided that it is greater than the throughput available in the wired Internet.

Acknowledgements

The authors wish to express their gratitude to Paul Barford for providing the SURGE traffic generator. We wish also to thank Mohan Kumar (University of Texas at Arlington) for the fruitful discussions, and for his support to build the distributed test-bed between Pisa and Arlington.

Appendix A. Proof of Proposition 1

Eq. (10) is derived from (5) by setting $\gamma_i = \gamma$, $\tau_i = 0$, $U = 0$ and $m = s + 1$. Hereafter, we proof the above claims.

Firstly, by the definition of global strategy, $U = 0$.

Moreover, in the global strategy, the main file and the embedded files are transferred over the wireless link when they are already stored at the access point. Therefore, the wireless link is used at its full available throughput, and hence γ_i in (5) becomes γ.[9]

[9] As shown in Section 6, the energy savings achieved with the global strategy are not significantly affected by variations of γ.

τ_i is negligible. This result derives from the following considerations: (i) the PS-Daemon includes Web proxy functionalities; (ii) the RTT between the mobile host and the access point is typically negligible; (iii) the overhead related to pre-fetching can be included in β_i, A and s as shown below.

The pre-fetching mechanism forces the mobile host to request a number of residual transfer-time estimates from the PS-Daemon during the active phase. Let g be this number ($g > 0$). When one of these estimates is greater than t_{so} the mobile host switches the wireless interface off. If s is the number of estimates greater than t_{so} then $m = s + 1$. Moreover, the time intervals during which the wireless interface remains on during idle periods within the active phase correspond to the estimates less than t_{so}. Therefore, $A \leq (g - s)t_{so}$. It must be pointed out that g is a random variable. Its distribution is very complex and depends on: (i) the residual transfer-time estimation algorithm; (ii) the throughput between the PS-Daemon and the server; (iii) the number and size of embedded files. Therefore, a closed formula for g is almost impossible to derive, and for this reason, to study the effectiveness of the global strategy, we performed an experimental analysis (see Sections 5 and 6).

Finally, β_i is made up of two components, and can be expressed as $\beta_i = h_i + p_i$, where h_i is the same as in (2), while p_i is the overhead introduced by the residual transfer-time estimation process associated with the ith file of a Web page. Specifically, during the active phase, the mobile host exchanges messages with the PS-Daemon to receive residual transfer-time estimates. The size of these messages is nearly constant, and can be approximated by the average size, $\bar{q}$. Therefore, $\bar{q} \cdot g$ is the overhead (in bytes) of a Web-page download, and $p_i = \bar{q} \cdot g/N$ is the overhead related to the ith file transfer.

This concludes the proof. □

References

[1] G.H. Forman, J. Zahorjan, The challenges of mobile computing, IEEE Comput. 27 (4) (1994) 38–47.

[2] T. Imielinscki, B.R. Badrinath, Wireless computing, Commun. ACM 37 (10) (1994) 18–28.

[3] J.R. Lorch, A.J. Smith, Software strategies for portable computer energy management, IEEE Pers. Commun. 5 (3) (1998) 60–73.

[4] M. Rulnick, N. Bambos, Mobile power management for wireless communication networks, ACM/Baltzer Wireless Netw. 3 (1) (1997) 3–14.

[5] M. Zorzi, R.R. Rao, L.B. Milstein, ARQ error control on fading mobile radio channels, IEEE Trans. Vehicular Tech. 46 (2) (1997) 445–455.

[6] M. Zorzi, R.R. Rao, Energy constrained error control for wireless channels, in: Proceedings of the IEEE GLOBECOM'96, 1996, pp. 1411–1416.

[7] L. Bononi, M. Conti, M. Donatiello, A distributed mechanism for power saving in IEEE 802.11 wireless LANs, Mobile Netw. Appl. (MONET) 6 (2001) 211–222.

[8] R. Bruno, M. Conti, E. Gregori, Optimization of efficiency and energy consumption in p-persistent CSMA-based wireless LANs, IEEE Trans. Mobile Comput. 1 (2002) 10–31.

[9] J.P. Ebert, B. Stremmel, E. Wiederhold, A. Wolisz, An energy-efficient power control approach for WLANs, J. Commun. Netw. 2 (2000) 197–206.

[10] R. Krashinsky, H. Balakrishnan, Minimizing energy for wireless web access with bounded slowdown, in: Proceedings of the Eighth Annual International Conference on Mobile Computing and Networking (Mobicom 2002), Atlanta, GA, September 23–28, 2002.

[11] H. Woesner, J.P. Ebert, M. Schlaeger, A. Wolisz, Power-saving mechanisms in emerging standards for wireless LAN's: the MAC level perspective, IEEE Pers. Commun. 5 (1998) 40–48 (special edition on power saving).

[12] J.R. Lorch, A.J. Smith, Scheduling techniques for reducing processor energy use in MacOS, ACM/Baltzer Wireless Netw. 3 (5) (1997) 311–324.

[13] M. Weiser, B. Welch, A. Demers, S. Shenker, Scheduling for reducing CPU energy, in: Proceedings of the First Symposium of USENIX Association on Operating System Design and Implementation, Monterey, CA, November 1994.

[14] C. Poellabauer, K. Schwan, Power-aware video decoding using real-time event handlers, in: Proceedings of the Fifth ACM International Workshop on Wireless Mobile Multimedia (WoWMoM2002), Atlanta, GA, September 28, 2002.
[15] D.P. Helmbold, D.E. Long, B. Sherrod, A dynamic disk spin-down technique for mobile computing, in: Proceedings of the Second Annual ACM International Conference on Mobile Computing and Networking, New York, November 1996, pp. 130–142.
[16] K. Li, R. Kumpf, P. Horton, T. Anderson, A quantitative analysis of disk drive power management in portable computers, in: USENIX Conference Proceedings, 1994.
[17] S.H. Phatak, V. Esakki, B.R. Badrinath, L. Iftode, Web&: an architecture for non-interactive web, in: Proceedings of the Second IEEE Workshop on Internet Applications, WIAPP'01, July 2001.
[18] J. Flinn, S. Park, M. Satyanarayanan, Balancing performance, energy, and quality in pervasive computing, in: Proceedings of the 22nd IEEE International Conference on Distributed Computing Systems (ICDCS), July 2002, pp. 217–226.
[19] A. Joshi, On proxy agents, mobility, and web access, ACM/Baltzer Mobile Netw. Appl. 5 (2000) 233–241.
[20] T. Imielinski, S. Vishwanathan, B.R. Badrinath, Power efficient filtering of data on air, in: Proceedings of the EDBT, Cambridge, UK, March 1994.
[21] A. Rudenko, P. Reiher, G.J. Popek, G.H. Kuenning, Saving portable computer battery power through remote process execution, ACM Mobile Comput. Commun. Rev. 2 (1) (1998) 19–26.
[22] M. Othman, S. Hailes, Power conservation strategy for mobile computers using load balancing, ACM Mobile Comput. Commun. Rev. 2 (1) (1998) 44–50.
[23] G. Anastasi, M. Conti, W. Lapenna, A power-saving network architecture for accessing the internet from mobile computers: design, implementation and measurements, The Comput. J. 46 (1) (2003) 3–15.
[24] R. Kravets, P. Krishnan, Power management techniques for mobile communication, in: Proceedings of the Fourth Annual ACME/IEEE International Conference on Mobile Computing and Networking (MOBICOM'98).
[25] L.M. Feeney, M. Nilsson, Investigating the energy consumption of a wireless network interface in an ad hoc networking environment, in: Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2001), 2001.
[26] M. Stemm, R.H. Katz, Measuring and reducing energy consumption of network interfaces in handheld devices, IEICE Trans. Fund. Electron Commun. Comp. Sci. 80 (1997) 1125–1131 (special issue on mobile computing).
[27] IEEE Standard for Wireless LAN—Medium Access Control and Physical Layer Specification, P802.11, November 1997.
[28] G. Anastasi, M. Conti, E. Gregori, A. Passarella, Balancing energy saving and QoS in the mobile internet: an application-independent approach, in: Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS-36), Big Island, HI, January 2003.
[29] G. Anastasi, M. Conti, E. Gregori, A. Passarella, A power saving architecture for web access from mobile computers, in: Proceedings of the IFIP Networking Conference (Networking'02), Pisa, Italy, May 2002, Lecture Notes in Computer Science, Vol. 2345, Springer, Berlin, 2002, pp. 240–251.
[30] A. Bakre, B.R. Badrinath, Implementation and performance evaluation of indirect TCP, IEEE Trans. Comput. 46 (3) (1997).
[31] H. Balakrishnan, V.N. Padmanabhan, S. Seshan, R.H. Katz, A comparison of mechanisms for improving TCP performance over wireless links, IEEE/ACM Trans. Netw. 5 (6) (1997) 756–769.
[32] C. Cunha, A. Bestavros, M. Crovella, Characteristics of WWW client-based traces, Technical Report TR-95-010, Department of Computer Science, Boston University, April 1995.
[33] M. Arlitt, C. Williamson, Internet web servers: workload characterization and performance implication, IEEE/ACM Trans. Netw. 5 (5) (1997) 631–645.
[34] M. Crovella, A. Bestavros, Self-similarity in world wide web traffic: evidence and possible causes, IEEE Trans. Netw. 5 (6) (1997) 835–846.
[35] P. Barford, M. Crovella, Generating representative web workloads for network and server performance evaluation, in: Proceedings of the ACM SIGMETRICS'98, Madison, WI, June 1998, pp. 151–160.
[36] P. Barford, A. Bestavros, A. Bradley, M. Crovella, Changes in Web client access patterns: characteristics and caching implications, World Wide Web (special issue on characterization and performance evaluation), 1999.
[37] M. Arlitt, T. Jin, Workload Characterization of the 1998 World Cup Web Site, HP Laboratories Palo Alto, HPL-1999-35(R.1), September 1999.
[38] S. Agrawal, S. Singh, An experimental study of TCP's energy consumption over a wireless link, in: Proceedings of the Fourth European Personal Mobile Communications Conference, Vienna, Austria, February 20–22, 2001.
[39] W.R. Stevens, TCP/IP Illustrated, vol. 1, The Protocols, Addison-Wesley, Reading, MA, 1994.

[40] A. Passarella, Un'architettura power saving per l'accesso al Web da computer mobile, Laurea Thesis, University of Pisa, October 2001 (in Italian).
[41] A.M. Law, D. Kelton, Simulation Modeling and Analysis, 2nd ed., McGraw-Hill, New York, 1991.

G. Anastasi received the Laurea degree in Electronics Engineering and the Ph.D. degree in Computer Engineering both from the University of Pisa, Italy, in 1990 and 1995, respectively. He is currently an Associate Professor of Computer Engineering at the Department of Information Engineering of the University of Pisa. His research interests include architectures and protocols for mobile computing, energy management, QoS in mobile networks, and ad hoc networks. He was a co-editor of the book *Advanced Lectures in Networking* (LNCS 2497, Springer, 2002), and published more than 40 papers, both in international journals and conference proceedings, in the area of computer networking. He served in the TPC of several international conferences including *IFIP Networking 2002* and *IEEE PerCom 2003*. He is a Member of the IEEE Computer Society.

M. Conti received the Laurea degree in Computer Science from the University of Pisa, Italy, in 1987. In 1987 he joined the Italian National Research Council (CNR). He is currently a Senior Researcher at CNR-IIT. His research interests include Internet architecture and protocols, wireless networks and ad hoc networking, mobile computing, and QoS in packet switching networks. He co-authored the book "Metropolitan Area Networks" (Springer, London, 1997), and published in journal and conference proceedings more than 100 research papers related to design, modeling, and performance evaluation of computer-network architectures and protocols. He served as the Technical Program Committee Chair of The Second IFIP-TC6 Networking Conference "Networking2002", and technical program committee co-chair of *ACM WoWMoM 2002*. He is serving as Technical Program Committee Chair of The Eighth International Conference on Personal Wireless Communications (PWC2003). He served as Guest Editor for the *Cluster Computing Journal* (special issue on "Mobile Ad Hoc Networking"), *IEEE Transactions on Computers* (special issue on "Quality of Service issues in Internet Web Services"), and *ACM/Kluwer Mobile Networks and Applications* Journal (special issue on "Mobile Ad hoc Networks"). He is Member of IFIP WGs 6.2, 6.3 and 6.8.

E. Gregori received the Laurea degree in Electronic Engineering from the University of Pisa in 1980. He joined CNUCE, an institute of the Italian National Research Council (CNR) in 1981. He is currently a CNR Research Director. In 1986 he held a visiting position in the IBM Research Center in Zurich working on network software engineering and on heterogeneous networking. He has contributed to several national and international projects on computer networking. He has authored more than 100 papers in the area of computer networks and has published in international journals and conference proceedings and is co-author of the book "Metropolitan Area Networks" (Springer, London, 1997). He was the General Chair of The Second IFIP-TC6 Networking Conference "Networking2002". His current research interests include: wireless access to Internet, wireless LANs, quality of service in packet-switching networks, energy saving protocols, evolution of TCP/IP protocols. He is on the editorial board of the *Cluster Computing* Journal.

A. Passarella received the degree (cum laude) in Computer Engineering from the University of Pisa, Italy, in 2001. Since 2002 he is at the Department of Information Engineering of the University of Pisa, where he is currently pursuing his Ph.D. degree. His research interests include architectures and protocols for mobile computing, energy management for small-size mobile computers and ad hoc networks.