WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2008; **8**:213–230
Published online 12 September 2006 in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/wcm.456

WILEY InterScience® DISCOVER SOMETHING GREAT

Web browsing optimization over 2.5G and 3G: end-to-end mechanisms vs. usage of performance enhancing proxies

C. Gomez*,†, M. Catalan, D. Viamonte, J. Paradells and A. Calveras
Wireless Networks Group (WNG), Entel Department, Universitat Politècnica de Catalunya (UPC), Campus del Baix Llobregat, Edifici C4, Avda. del Canal Olímpic s/n, 08860 Castelldefels, Spain

2.5 Generation (2.5G) and Third Generation (3G) cellular wireless networks allow mobile Internet access with bearers specifically designed for data communications. However, Internet protocols under-utilize wireless wide area network (WWAN) link resources, mainly due to large round trip times (RTTs) and request–reply protocol patterns. Web browsing is a popular service that suffers significant performance degradation over 2.5G and 3G. In this paper, we review and compare the two main approaches for improving web browsing performance over wireless links: (i) using adequate end-to-end parameters and mechanisms and (ii) interposing a performance enhancing proxy (PEP) between the wireless and wired parts. We conclude that PEPs are currently the only feasible way for significantly optimizing web browsing behavior over 2.5G and 3G. In addition, we evaluate the two main current commercial PEPs over live general packet radio service (GPRS) and universal mobile telecommunications system (UMTS) networks. The results show that PEPs can lead to near-ideal web browsing performance in certain scenarios.

KEY WORDS: UMTS; GPRS; TCP/IP; performance optimization; end-to-end; PEP

1. Introduction

Over the last few years, wireless technologies have been providing core elements to clear the way for a ubiquitous Internet. In the wide area, 2.5 Generation (2.5G) systems such as the general packet radio service (GPRS) [1] have been deployed all over the world. 2.5G networks are an important step in migrating towards third generation (3G) networks. Furthermore, the universal mobile telecommunications system (UMTS) [2] is an example of 3G technology that is now commercially available in many countries.

It is a widely known problem that Internet protocols, which were designed under the assumption that they would be used over wired links, underperform over wireless wide area networks (WWANs). Researchers have devoted a great deal of effort to optimizing the mobile user experience for over a decade. Their efforts have focused on developing solutions to cope with

*Correspondence to: C. Gomez, Wireless Networks Group (WNG), Entel Department, Universitat Politècnica de Catalunya (UPC), Campus del Baix Llobregat, Edifici C4, Avda. del Canal Olímpic s/n, 08860 Castelldefels, Spain.
†E-mail: carlesgo@entel.upc.es

Contract/grant sponsors: Spanish Government; FEDER; DURSI; FSE.

WWAN characteristics such as low data rates, high round trip times (RTTs), jitter, bit error rate (BER), coverage outages, and handovers. In particular, mitigating suboptimal performance of the Transmission control protocol (TCP) over lossy environments has been a major concern in research circles for many years. Several procedures have been proposed for avoiding unnecessary TCP congestion control activation after losses in the radio link [3–5]. However, recent studies have shown that 2.5G and 3G link layer mechanisms are able to ensure low loss probability to upper layers, especially when the network is properly tuned by the operator [6–8]. Actually, the main problem in WWANs is the performance degradation, which is a result of long RTTs in these networks [6–11]. Long RTTs have detrimental effects on request/reply protocol patterns. This in turn significantly increases the times of several mechanisms such as: (i) the TCP three-way handshake and slow start at the transport layer, (ii) the domain name system (DNS) lookup at the session layer, and (iii) application layer protocols themselves if there is a severe mismatch between the TCP and such protocols. As a result, transmission delays are longer.

Web browsing is one of the Internet's killer applications, which also applies to mobile users [12]. However, the hypertext transfer protocol (HTTP) [13,14] misuses the TCP/IP data delivery service as configured by default in current browsers [15–17]. Moreover, many popular websites contain objects hosted under different domain names, thus making a significant number of DNS lookups necessary that add to the total download time [6,17]. High RTTs in 2.5G and 3G networks heighten these effects. The result is a very different web browsing experience from that attainable with fixed Internet access.

The solutions that have been proposed for compensating performance degradation experienced on the Internet over WWAN can be divided into two main groups. The first is the use of appropriate end-to-end mechanisms and/or protocol configuration parameters [18]. The second is the introduction of a proxy between the endpoints of the communication. This kind of element, which can either be centralized or distributed, is known as a performance enhancing proxy (PEP) [19]. There is some opposition to the use of PEPs, since in many cases, they break the Internet principle based on keeping end-to-end semantics [20]. However, as we attempt to justify in this work, we believe that PEPs are the only feasible way to significantly improve access to the Internet via WWAN networks.

In this paper, we focus on web browsing optimization over 2.5G and 3G. The remainder of the article is organized as follows. In Section 2, we present an IP level characterization of GPRS and UMTS, indicating how they are related to web performance. We then review and discuss a comprehensive range of empirically validated optimizations, including both end-to-end techniques and the use of PEPs. In Section 3, we identify a set of relevant transport and application layer end-to-end options for web browsing. We conclude that standard end-to-end mechanisms are not feasible solutions for improving performance over WWANs. In Section 4, we examine the main PEP techniques available in the literature. We go on to evaluate the two main current commercial PEP products (i.e., FlashNetworks NettGain [21] and Bytemobile Macara [22]) over live GPRS and UMTS networks. Section 5 describes the scenarios in which our commercial PEP evaluation is undertaken. In Section 6, we present the main results of our experiments, showing that PEPs may lead to a performance close to that attained with an ideal web page download protocol. Finally, Section 7 summarizes the concluding remarks derived from this work.

2. IP Characterization of GPRS and UMTS

In this section, we highlight the main features of GPRS and UMTS that influence web browsing, namely: throughput and latency. These characteristics determine how relatively long an RTT is. As will be shown, RTTs are long in both GPRS and UMTS. This leads to inefficient web browsing downloads with long periods of inactivity. In addition, there are other phenomena in WWANs, such as cell reselection in GPRS [9], handover in UMTS [10], link outages and delay jitter [8,23], which may yield undesired interactions with the TCP Retransmission TimeOut (RTO) such as serial timeouts and spurious retransmissions.

2.1. GPRS

2.1.1. Throughput

The GPRS system offers a number of access speeds through the radio interface. An average user may expect around 34–35 kbps in the downlink at the IP level [8]. Empirical results show that using large IP packets, that is, large maximum transmission unit

DOI: 10.1002/wcm

(MTU) values, balances header overhead against packet error probability, thus optimizing IP level throughput. For example, achieved throughput using 128-byte IP packets is 17% lower than that achieved with 1500-byte packets [9].

2.1.2. Latency

Many published works provide results on RTT measurements over live GPRS networks [6,8,9,16]. These works reach the conclusion that the GPRS link significantly contributes to the end-to-end delay of a communication. The authors in Reference [8] report that 80% of the RTT values in GPRS fall within the range between 650 ms and 1 s. As a consequence, interactive applications are subject to significant performance degradation over GPRS.

2.2. UMTS

2.2.1. Throughput

A UMTS network must be able to provide adequate bit rates within its coverage area, offering a peak bit rate of 64 kbps for basic coverage, 144 kbps for extended coverage, 384 kbps in hot spot areas and a theoretical potential peak bit rate of 2 Mbps. Up to 57 kbps uplink and up to 371 kbps downlink IP level available bit rates have been reported from live network measurements in urban environments [10]. Maximum throughput is obtained when large packets are used [10,11].

2.2.2. Latency

Current UMTS deployments offer dedicated channel bearers. Under these conditions, UMTS RTTs are close to fixed Internet RTT values. Measured RTTs fall in the range between 200 and 450 ms depending on the packet sizes [10]. A significant conclusion is that interactive applications will perform better over UMTS than over 2.5G technologies such as GPRS. However, UMTS latency measurements show that there is still room for performance improvement. Transmission delays in request–reply patterns are expensive if we take into consideration the available data rates offered by the system.

2.3. Discussion

Figure 1 illustrates how throughput and latency in GPRS and UMTS lead to significant idle times in HTTP. This figure depicts an HTTP Get/Response transaction for the download of an object of an IP level size equal to 1500 bytes, which fits into a single IP packet. Figure 2 represents a TCP connection including a three-way handshake and the first three rounds of Slow Start. Each figure has been plotted to scale to allow the comparison of GPRS and UMTS. We worked with the RTT values and IP-level bit rates measured in References [8–10]. As can be seen, a large amount of time is wasted in waiting for both the TCP and HTTP. In the next section, the interaction between the TCP and HTTP will be discussed in terms of HTTP end-to-end options.

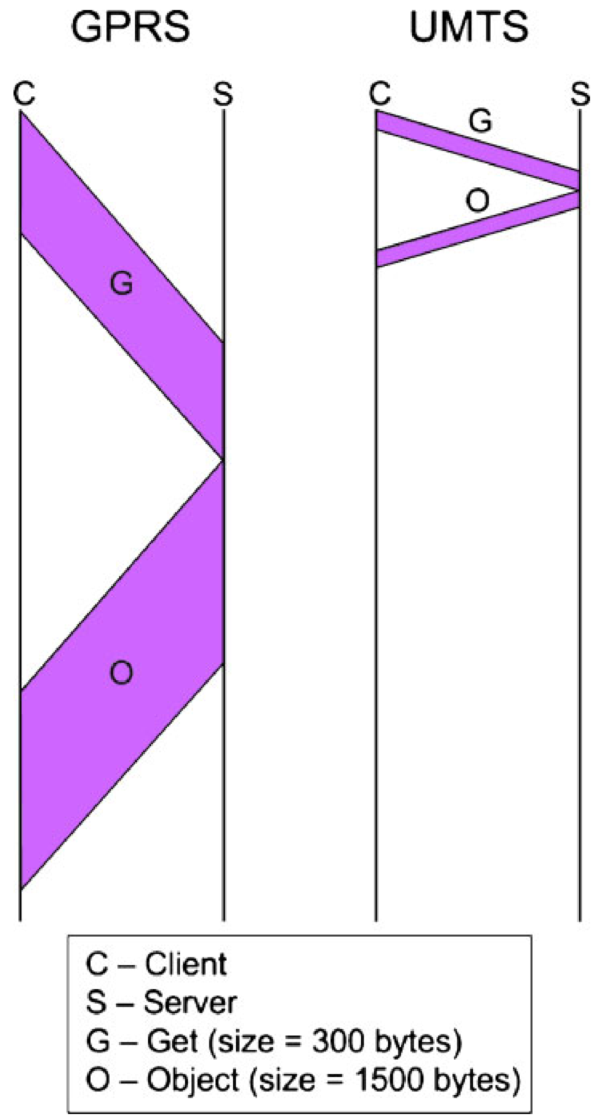


Fig. 1. Influence of GPRS and UMTS parameters (latency and throughput) on an HTTP transaction.

DOI: 10.1002/wcm

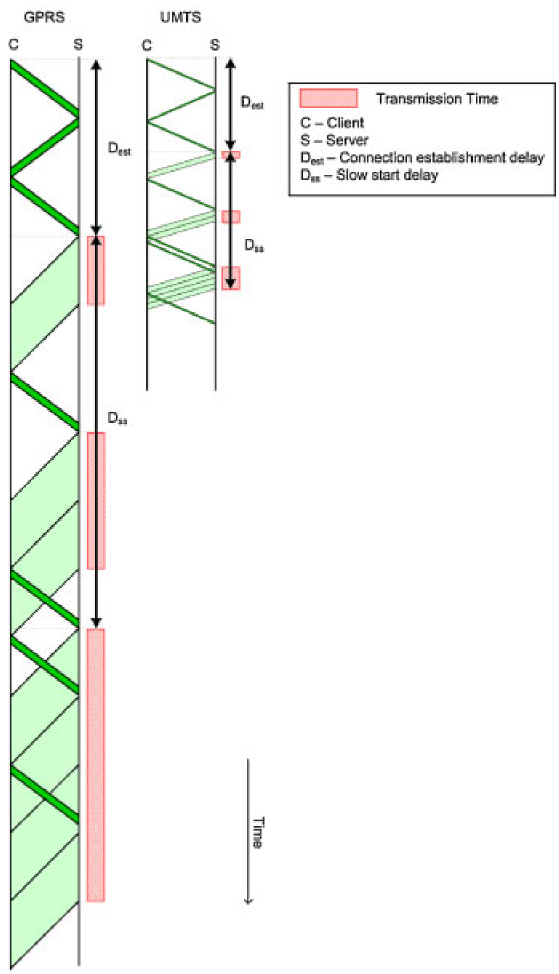


Fig. 2. Influence of GPRS and UMTS parameters (latency and throughput) on a TCP connection.

3. End-to-End Options and Improvement Techniques

An initial approach for enhancing web performance over WWAN is to use end-to-end techniques. In this section, we survey the most relevant available options in standard TCP and HTTP for minimizing 2.5G and 3G link under-utilization. Note that there exist other end-to-end solutions that improve performance over wireless links. Some of them are based on modifying standard protocols (e.g., Freeze-TCP [24]), while others propose the use of an alternative protocol (e.g., WTCP [25]). However, we prefer to focus on mechanisms that can be configured on operating systems and web browsers/servers that have already been deployed. We finally discuss the advantages and disadvantages of the techniques presented.

3.1. TCP End-to-End Techniques

Organizations such as the IETF and the WAP Forum have already produced TCP parameter configuration proposals for use over 2.5G and 3G networks [18,26], which have been empirically validated in several studies [8–10]. We then present the following relevant TCP parameters: maximum segment size (MSS), maximum transmission window, selective acknowledgements (SACKs), and timestamps. We describe and quantify how they affect performance over GPRS and UMTS.

3.1.1. Maximum segment size (MSS)

The MSS value must fit into the IP MTU in order to avoid fragmentation at the TCP layer. Therefore, the choice of an appropriate MSS depends on the MTU. However, the MSS may interact with some TCP mechanisms. For example, since the congestion window is counted in units of segments, high MSS values allow TCP congestion windows to increase faster. Empirical results obtained in GPRS and UMTS prove that high MSS values (e.g., 1460 bytes) are recommended in order to optimize TCP throughput in such environments. According to measurements on live GPRS networks [8], TCP throughput using a 1500 byte MSS is between 15% and 20% higher than for an MSS equivalent to 552 bytes. Similar figures are obtained in live UMTS networks [10].

3.1.2. Maximum transmission window

The maximum transmission window of a TCP sender is defined by the Receive WINdow (RWIN) advertised by the receiver in each TCP segment. The maximum transmission window must be at least equal to the bandwidth delay product (BDP) of a path to fully utilize its capacity. At present, the authors in [8,16,27] suggest using larger windows over 2.5G and 3G networks in order to ensure maximum link utilization. Another aspect to be considered is that WWANs, especially GPRS networks, are usually the cause of a bottleneck on a TCP connection due to their low bandwidth. Thus, window values that are too high may contribute to excessive queuing at the gateway to the wireless network. Consequently, RTTs may suffer significant inflation, leading to severe performance degradation [23]. Finally, BDP is also related to the MSS, since the number of segments in the transmission window affects TCP congestion mechanisms. A window size of at least four segments is needed for applying Fast Retransmit and Fast Recovery mechanisms. Assuming

DOI: 10.1002/wcm

that 1460-byte MSSs are used, the BDP is roughly equal to 6 kB (i.e., 4 segments) and 21 kB (i.e., 14 segments) in GPRS and UMTS respectively [9,10].

3.1.3. Selective acknowledgements (SACKs)

This option is useful when multiple losses occur in a single window, allowing the recovery of the lost segments in a single RTT. It may also help to improve TCP throughput since retransmissions may be performed earlier than the RTO expiration. Published results [3] aligned with IETF and WAP Forum recommendations state that the SACK option leads to a 10% improvement over GPRS networks.

3.1.4. Timestamps Option

Timestamps may be useful in 3G networks according to IETF recommendations [18]. This option enables a more accurate RTO calculation than by default. This should help to avoid spurious retransmissions and the activation of congestion avoidance mechanisms. However, timestamps add a 12-byte overhead to TCP headers. Measurements over GPRS networks show that a very minor improvement (around 3%) is achieved when timestamps are enabled [8].

3.2. Application Layer

Web browsing performance over WWAN is highly dependent on the HTTP version and the options used, which determine how TCP services are used by HTTP. In this section, we focus on the main tunable features and parameters of HTTP: versions HTTP 1.0 and HTTP 1.1, the keep-alive option, use of parallel persistent connections, and pipelining. We also consider the impact of object sizes on performance and content compression as supported by HTTP.

3.2.1. HTTP 1.0

HTTP 1.0 [13] is the first version of the protocol. In this case, the client opens a TCP connection for each object on the page. Once a connection is open, the client sends a GET message requesting the object to be downloaded. Therefore, a three-way handshake procedure for establishing connection, followed by a Slow Start phase, is performed for each object. Moreover, every GET request adds roughly one RTT to the total web page download time. This RTT wasting pattern results in a significant decrease in data transmission efficiency. Figure 3(a) shows an example of a three-object web page download with HTTP 1.0.

3.2.2. HTTP 1.1

Current browsers and servers mainly implement HTTP 1.1 [14] by default. This version addresses several issues that were somewhat neglected in HTTP 1.0 and contemplates a number of options that may reduce HTTP- and TCP-induced transmission stalls in a long RTT link:

(1) *Persistent connections*. In HTTP 1.1, one or more objects may be downloaded using the same TCP connection, therefore reducing the number of connection establishment and Slow Start phases. Some HTTP 1.0 clients support persistent connections by using the Keep-Alive option. Figure 3(b) shows the improvement obtained from avoiding HTTP 1.0 behavior.

(2) *Parallel connections*. HTTP 1.1 makes it possible to use a number of parallel connections in a web page download. This option may result in significant benefits over 2.5G or 3G since data may be transmitted through one connection, thus taking advantage of periods of inactivity in other concurrent connections. According to our observations, current browsers like Microsoft Internet Explorer and Netscape Navigator allow a maximum of two simultaneous connections, while other browsers like Opera allow up to four concurrent connections. Authors in Reference [7] obtain a 35–40% download time reduction over GPRS using six parallel connections. In our experiments, a 24–37% improvement was achieved over live UMTS networks, in which the optimal number of connections depends on the web page characteristics in terms of total size and object sizes. Figure 3(b) and (c) illustrates how using parallel connections help to reduce inactivity times.

(3) *Pipelining*. Pipelining is an option defined with the goal of avoiding the HTTP stop-and-wait effect derived from the object request–reply sequence within a TCP connection. When pipelining is used, a client may perform a GET request before the previous object is completely received. Ideally, pipelining should lead to a higher number of outstanding GET requests than object downloads at any time during the connection. Therefore, in such cases, HTTP-induced waiting times would be minimized (see Figure 3(d). Note that pipelining may be used regardless of the number of parallel TCP connections. An improvement of as much as 56% was measured using pipelining over GPRS [7]. Unfortunately, many web servers and proxies do not

DOI: 10.1002/wcm

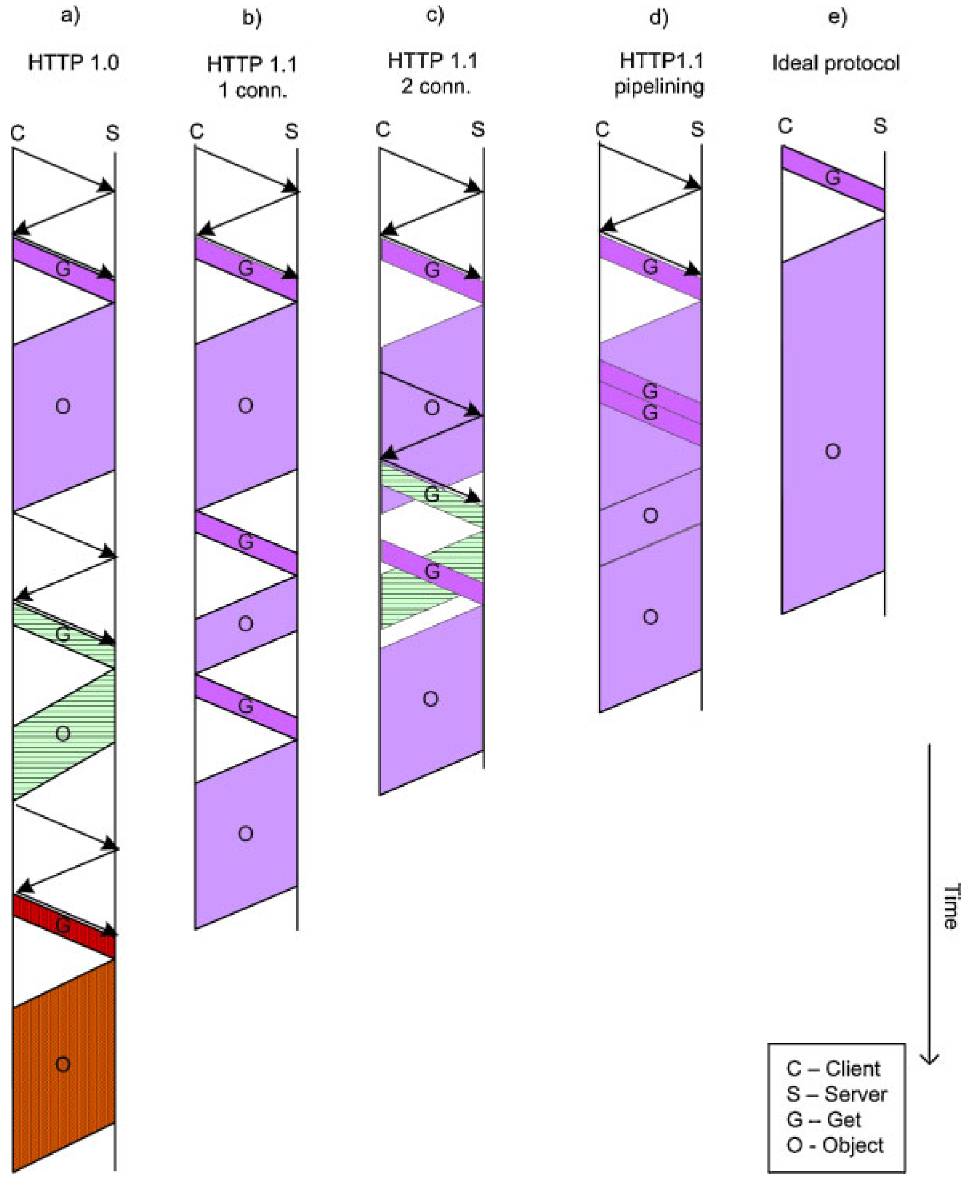


Fig. 3. A comparison of end-to-end HTTP options, together with an ideal web page download protocol. (a) using HTTP 1.0; (b) using HTTP 1.1 and a single TCP connection; (c) using HTTP 1.1 and two simultaneous TCP connections; (d) using HTTP 1.1 and pipelining; (e) using an ideal web page download protocol (see Section 6.3).

respect the pipelining feature. For this reason, many browsers disable this option by default or do not support it.

3.2.3. HTTP 1.1 and server pre-emptive FIN

Some web servers use a technique based on sending a pre-emptive TCP FIN segment that imposes a premature end of a TCP connection in order to reduce server load, which improves server performance [29]. Therefore, even if HTTP 1.1 is used, in such cases, the web download behavior may be close to that of an HTTP 1.0 communication. Hence, HTTP 1.0 performance is still relevant for worst case analysis.

3.2.4. Web page object sizes

Web pages are not built to keep the pipe full of data. The web browser client typically generates GET messages as uniform resource locator objects (URLs) appear on

DOI: 10.1002/wcm

the downloaded HTML text. The client does not know in advance the size of each object. This is a problem, since consecutive GET messages may require a series of very small objects. This pattern may lead to idle periods that could be avoided if small and large objects were interleaved with the web page download, minimizing transmission stalls. Thus, shorter download times could be achieved if the URL objects in the HTML text were suitably located.

3.2.5. Content compression

One way of improving performance over links with relatively low data rates (such as 2.5G and 3G) is by performing content compression in order to reduce the amount of application data to be transmitted, and the corresponding lower layer header overhead. There are two main kinds of compression techniques depending on the nature of the content: (i) lossless compression, which applies to text- or binary-based content (such as HTML, Javascript, CSS, etc) and (ii) lossy compression, which can be used for image-based content (JPEG, GIF, bitmap, etc). In the latter case, image quality is sacrificed in order to obtain smaller sized image objects.

HTTP 1.0 web servers support lossless compression coding techniques by using default formats such as gzip and compress [13]. Additional lossless encoding formats such as deflate are supported in HTTP 1.1 [14]. Note that negligible gain may be achieved when lossless compression is applied to content already compressed (e.g., image formats). Lossy compression techniques are commonly used by PEPs.

3.3. Discussion

There are two main reasons in favor of using end-to-end mechanisms for improving performance of the web over 2.5G and 3G:

(1) End-to-end mechanisms are based on standard options and maintain end-to-end semantics. Thus, they are fully compliant with the Internet architecture.
(2) Empirical results demonstrate a significant improvement, especially when adequate HTTP settings are used.

However, the following drawbacks can be pointed out:

(1) The design criteria of Internet servers aim to optimize server throughput by minimizing the amount of open connections [25,29]. This is actually the reason why TCP connections are pre-emptively closed even if HTTP 1.1 is used. Thus, this philosophy inherently limits the applicability of using a number of parallel connections for web downloads.
(2) Certain relevant parameters cannot be optimized at the same time for different access technologies. For example, as mentioned above, BDP in UMTS is more than three times that of GPRS. Moreover, servers are by default unaware of the access technology used by a client. Additional intelligence could be provided to a server for selecting adequate protocol settings in each case, but this approach is not used for practical reasons.
(3) At best, at least one TCP Slow Start phase will still take place during a web page download.
(4) Should multiple objects be hosted under different domain names, DNS lookup overhead cannot be avoided or reduced using end-to-end options.
(5) End-to-end mechanisms cannot cope with connectivity gaps suffered when link outages or cell reselection/handovers occur. The reason for this is that a server is unable to distinguish between congestion and radio link losses. This gives rise to the unnecessary activation of TCP congestion control mechanisms.

Considering the issues discussed above, we conclude that end-to-end optimization mechanisms cannot provide optimal performance. An alternative solution is the use of PEPs. This approach has in fact been adopted by a large number of mobile operators. In the next section, we examine the main published PEP techniques available in the literature.

4. PEP Improvement Techniques

A PEP is an entity placed between the endpoints of a communication, which is designed for improving degraded performance caused by the nature of specific link environments. Several PEP solutions have been proposed for wireless environments, such as satellite links, WWANs and wireless local area networks (WLANs) [13]. PEPs may allow two operation modes, namely: client-server optimization (C/S), a distributed system that includes additional software on the client side, and clientless optimization (CLS), which is a centralized proxy approach. Other works refer to C/S and

DOI: 10.1002/wcm

CLS PEPs as explicit and transparent PEPs respectively. PEP operation may be held at the transport layer, session layer and/or application layer. Note that optimizations at different layers can be combined to operate simultaneously. We refer to such PEPs as multilayer PEPs. We will now go on to review the main PEP techniques available in the literature. The section ends with a discussion on the use of PEPs.

4.1. Transport Layer PEPs

Research on transport layer PEPs has focused on solving the impact of the wireless link on the TCP in several ways. We can classify TCP PEPs into three categories depending on the degree of change they bring about in the communication: (i) PEPs that respect the end-to-end argument, which is one of the architectural principles of the Internet, (ii) proposals that split the TCP connection into two parts, in an attempt to isolate mobility and wireless related problems from the existing network protocols, and (iii) procedures that define custom protocols over the wireless link.

4.1.1. PEPs that keep the end-to-end argument

Snoop [3] is a well-known example of a PEP that does not break the end-to-end principle. It is a link layer TCP-aware PEP designed for WLAN environments that performs local retransmissions. Thus, Snoop avoids the activation of sender congestion control mechanisms, while maintaining the end-to-end semantics. The ACK regulator [30] is another PEP in the same category that monitors and controls ACKs on the uplink channel, in order to regulate the flow on the downlink channel to avoid buffer overflow and the activation of congestion control mechanisms.

4.1.2. Split connection TCP PEPs

I-TCP [4] and M-TCP [5] are some of the first examples based on a split connection approach that include additional features such as support for dealing with disconnection periods. Another split connection solution is W-TCP [31], which performs local retransmissions to the mobile user. This is a unique solution in which, despite splitting the end-to-end connection, W-TCP authors claim that end-to-end semantics are maintained. Authors argue that an ACK is sent to the fixed host only if an ACK from the mobile user has reached the base station.

Note that, as mentioned above, recent work does not aim to handle packet losses induced by wireless link errors. Instead, more recent split connection approaches such as TCP WWAN [32] or the proxy presented in Reference [33] attempt to use optimized protocol settings (e.g., an appropriate window size) and/or avoid Slow Start (e.g., by clamping the TCP window to a BDP estimate of the link).

4.1.3. Custom transport protocol PEPs

Custom protocol approaches are designed to avoid the TCP's connection setup and Slow Start costs. One example is GPRSWeb [23], a protocol that operates on top of the UDP, which takes advantage of GPRS reliability for using an error recovery strategy based on selective repeats with negative acknowledgement (NACK). The connection start-up method used by GPRSWeb is very similar to the TCP Accelerated Open (TAO) developed for T/TCP [34]. Thus, no handshake is needed. Another custom transport protocol is the wireless application protocol (WAP) [35], in which a WAP gateway splits the communication with a new protocol for use over the wireless link.

4.2. Session Layer PEPs

Two main session layer PEPs have been presented in the literature: (i) DNS- and URL-rewriting and (ii) parse and push.

4.2.1. DNS- and URL-rewriting

Web page content may be distributed among a number of servers. In such cases, several DNS lookups must be performed by a client. DNS re-writing [17] is a technique whereby a proxy may force the client to point to one single proxy server where the web content is assumed to be already downloaded, thus reducing the number of DNS lookups and TCP connection establishments. Similarly, uniform resource locators URLs on a web page can be replaced by the IP address of a proxy server. Note that more than one proxy server may be used. Using the appropriate number of proxy servers may improve performance, leading to behavior analogous to that of using an optimal number of simultaneous TCP connections.

4.2.2. Parse and push

Parse and push proxies aim to push objects towards the mobile user before such objects are actually requested.

DOI: 10.1002/wcm

According to published results, this scheme helps to improve wireless link utilization by between 5 and 12% [7].

4.3. Application Layer PEPs

Below, several application layer PEP techniques are summarized, namely: extended caching, compression and content format adaptation, delta encoding, and client prefetching.

4.3.1. Extended caching

A distributed extended caching procedure is presented in Reference [23]. This solution implements a client-side cache that replaces the browser's permanent cache. The client communicates through a custom protocol with another caching entity close to the base station (i.e., PEP server). Fingerprints of objects mapped to URLs are used to determine whether or not objects have changed with respect to previous web page downloads. When a URL mapping expires on the client side cache, the client PEP asks the PEP server to check whether a more recent mapping exists. If the object has not changed, then the PEP server simply retransmits the URL to fingerprint mapping to the client PEP. Thus, unnecessary object downloading through the wireless link is avoided, reducing web page download times and bandwidth requirements.

4.3.2. Compression and content format adaptation

A PEP may apply compression techniques (see Section 3.2.5) in order to reduce the total amount of data sent to the wireless link. In particular, lossy compression, which is not supported by default by servers, is a good candidate technique to be used by a PEP. Note that the achievable compression degree for lossless compression may be statistically characterized, while lossy compression gain is not known *a priori* and may be tuned by the user (when possible) or selected by the proxy administrator.

Additionally, content format adaptation mechanisms may be supported by a PEP in order to deliver information in a suitable format, depending on the device's hardware features and also on what the user wants to display on his/her terminal. Efficient formats are needed in order to save bandwidth and device power consumption. Existing content adaptation mechanisms based on an identification of the terminal capabilities have been defined. Some examples of this approach are the use of HTTP headers and the set of procedures developed by the W3C and the WAP Forum for describing the client's capabilities contained in the Composite Capabilities/Preferences Profile (CC/PP) and User Agent Profile (UAProf) specifications [36].

4.3.3. Delta encoding

Delta encoding is a technique based on sending differences between new and old versions of a document [41]. This strategy avoids the need for downloading a web page every time a client wishes to access its content. Thus, it is generally successful since updated documents are often significantly similar to their predecessors [23]. Significant reductions in the amount of data transmitted (between 61 and 99%) have been measured when delta encoding is applied to web page downloads [37].

4.3.4. Client prefetching

Client prefetching allows early fetches to be performed on web pages linked to pages that have already been read. The rationale behind this mechanism is as follows. Inactive link periods may be used so that when a user clicks on a link on a displayed web page, the new one appears instantaneously on the client browser [38]. However, applying client prefetching to WWANs has been questioned, since charging models in WWANs are volume-based in many cases [23].

4.4. Multilayer PEPs

Significant performance enhancement can be achieved by combining optimizations at more than one layer [7]. A number of PEPs provide multilayer optimization. Mowgli [28,40] uses a custom transport protocol for the wireless link, together with application layer techniques such as compression and content format adaptation. GPRSWeb [22] uses a UDP-based custom transport protocol, together with session layer optimizations such as DNS rewriting and 'parse and push,' and application layer solutions including extended caching, data compression, content format adaptation, and delta encoding.

Remarkably, FlashNetworks NettGain [21] and Bytemobile Macara [22], the two main current commercial PEPs on the WWAN market, also belong to the multilayer PEP category. The latter performs protocol acceleration by reducing the number of transactions (RTTs) needed for data transfer with the TCP and HTTP. HTTP is replaced with the

DOI: 10.1002/wcm

Macara Dynamic Interleaving technology, which is intended to minimize the effect of latency on web page downloads. Macara imposes the use of multiple TCP connections and merges multiple data requests and responses. Techniques for improving the efficiency of HTML, JavaScript, and CSS are applied before data compression and content format adaptation is performed. Finally, extended caching is another application layer procedure used by Macara. Regarding NettGain [39], an early version of this PEP supported several application layer mechanisms such as client prefetching, data compression, and content format adaptation. Exact details about the methods used by each proxy provider are not public. In consequence, the authors have preferred not to provide this information, since the purpose of the paper is to show the benefits of proxies, rather than comparing both commercial solutions.

4.5. Discussion

The key advantage that PEPs offer is that they are fully aware of the radio technology of the mobile terminal. In addition, they are specifically developed for optimizing mobile user performance, which is not a design criterion for an all-purpose Internet server. The main argument against using PEPs is that optimum performance is usually achieved by using mechanisms that break the end-to-end principle. Some additional related issues include PEP server scalability and security [23].

Given the various PEP optimizations described so far, our purpose is to investigate the web download performance currently provided by commercial PEPs, that is, performance as perceived by a significant number of mobile users.

5. Test Scenarios and Description of the Methodology for Commercial PEP Evaluation Over GPRS and UMTS

In this section, we describe the experimental setup and methodology we use to quantify the web browsing optimization provided by NettGain and Macara over live GPRS and UMTS networks. The trials reported in this paper were held during the last week of January 2004.

5.1. Test Scenarios

We will now present the scenarios in which tests were performed. The evaluation process was carried out in three different scenarios, which we will refer to as Scenario 1, Scenario 2, and Scenario 3. In the first two, a public GPRS access was used. In the third one, a pre-commercial UMTS network was used. All trials were performed in a static indoor scenario. The Received Signal Strength Indication (RSSI) ranges measured during the trials were between −95 dBm and −80 dBm in GPRS, and between 107 dBm and −93 dBm in UMTS. A description for each test scenario now follows.

5.1.1. Scenario 1 description

Scenario 1, depicted in Figure 4, consists of a procedure in which the mobile access was performed using the public GPRS network of a Spanish operator. The purpose of this testbed was to evaluate a Beta version of NettGain. Three test cases were considered in Scenario 1: using the NettGain C/S system, using the NettGain CLS system, and finally, the default communication without any optimization system.

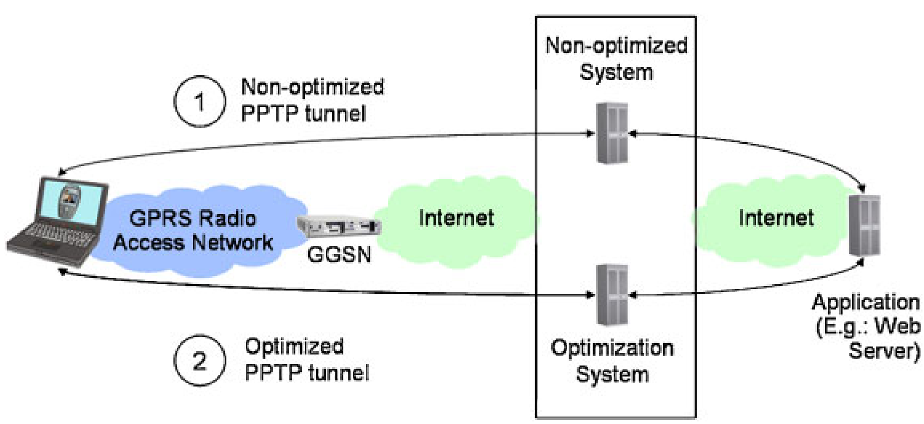


Fig. 4. Scenario 1 overview.

DOI: 10.1002/wcm

Two different paths were defined for this scenario. The path labeled in Figure 4 as (1) did not include an optimization system. A point-to-point tunneling protocol (PPTP) tunnel was established between the Mobile Station (MS) and a PPTP Access Concentrator (PAC), in which PPTP de-encapsulation and network address translation (NAT) was carried out. This scenario served as a reference for comparison with optimized communications, for which the tunnel was necessary for carrying the traffic to the proxy element. The second path, indicated in Figure 4 as (2), was used for testing NettGain in its two modes of operation. The path is analogous to the previous one, but in this case, the tunnel ends at the NettGain proxy element, which, in addition to the de-encapsulation and NAT services, also performs optimization operations on the end-to-end traffic. Note that tunneling leads to a reduction in performance of between 1.9 and 2.5% due to header overhead.

When NettGain C/S was used, the NettGain client ran on the MS. Communication between the NettGain client and server took place over the PPTP connection using a set of optimized protocols. At the tunnel endpoint, the optimization server acted as a standard client making requests to the remote end of the communication on behalf of the mobile user, so optimization occurred over the NettGain client–server link (thus including the GPRS radio and core network path). Figure 5 shows the protocol architecture involved in the C/S environment. In the CLS mode, communication over the PPTP tunnel is based on the use of standard protocols.

5.1.2. Scenario 2 description

Scenario 2 is a GPRS connectivity procedure for evaluating the performance of a version of Macara, which was being used by a German GPRS mobile operator at the time of the trials. Two test cases are considered in Scenario 2: use of Macara C/S and use of Macara CLS.

Figure 6 shows the architecture of this scenario. In this case, the MS was subscribed to the German GPRS mobile operator network, but radio access was performed from the above-mentioned Spanish GPRS network through a GPRS Roaming eXchange (GRX). The end-to-end communication crossed the Macara optimization system.

Scenarios 1 and 2 can be considered to be approximately the same, although Scenario 1 was expected to perform slightly worse due to PPTP tunneling and the presence of two public Internet paths in the end-to-end communication. The protocol architecture related to the C/S mode implemented by Macara is equivalent to that illustrated in Figure 5 without the additional stack due to tunneling (i.e., IP/PPP/GRE additional stack). In the CLS mode, communications are based on standard protocols.

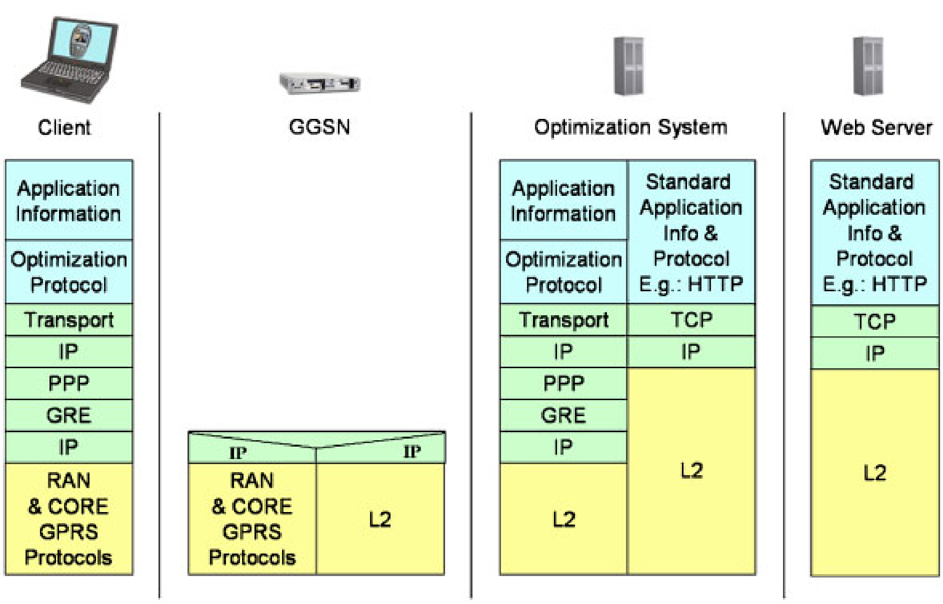


Fig. 5. Scenario 1. Protocol architecture for NettGain C/S optimization, including the PPTP related mechanisms used in the trials.

DOI: 10.1002/wcm

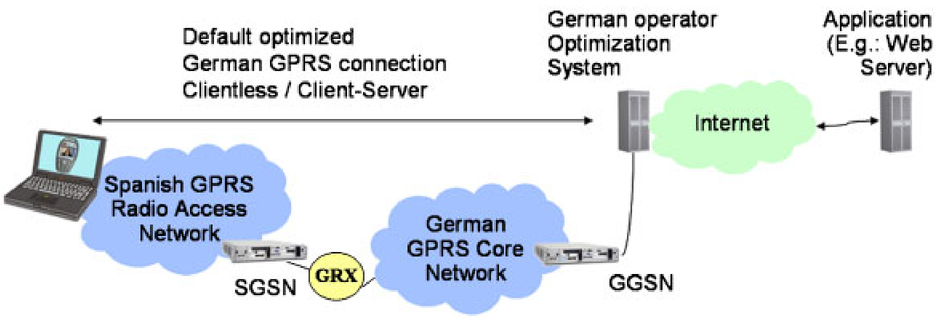


Fig. 6. Scenario 2. German GPRS operator connectivity paths.

5.1.3. Scenario 3 description

Scenario 3 is a UMTS connectivity network that used the pre-commercial network of a Spanish mobile operator. This testbed was set up for evaluating PEP performance over a live 3G network. It was the same as Scenario 1 except that it used UMTS mobile access. Only NettGain was tested since we did not have access to Macara in a UMTS environment. NettGain C/S and CLS modes were tested and compared with a non-optimized communication.

5.2. System Configuration

This section describes the features of both hardware and software elements involved in the tests.

5.2.1. Hardware configuration

All tests were performed using a set of IBM R32 Pentium IV 1.7 GHz processor, 256 MB RAM laptop computers. The mobile device used in GPRS trials was a Nokia 6650 phone Class 6 terminal supporting GSM/GPRS/UMTS access—Software v13.89. The network interface used in UMTS trials was a Novatel Wireless Merlin U530 PCMCIA card, while the UMTS service used was a 384 kbps/64 kbps dedicated data packet bearer with the Background QoS class.

5.2.2. Software configuration

Three different laptops with the same hardware features were used as clients. A different software setup was used on each laptop (i.e., trials were performed using a different laptop with NettGain, Macara, and a non-optimized system respectively). This strategy prevents the settings for each test affecting the performance of the other ones. The operating system and browser used in the trials were Windows XP Professional v5.1 and Internet Explorer 6.0.28 respectively. The choice was based on market share criteria in order to ensure that the results were truly representative. It must be noted that the default Internet Explorer parameter configuration was used, which means that HTTP 1.1 and up to two TCP simultaneous connections were used for a web page download. PEP client versions tested were the Bytemobile Macara v1.3.0.2090 Client and the Flash Networks NettGain 1100 Client 3.0. Finally, NettGain allows the user to configure a set of image quality levels, some of which were tested. Macara does not offer image quality configuration options to the user.

5.3. Description of Reference Web Pages

A set of 10 reference web pages was selected according to the following two criteria: (i) possibility of evaluating different types of contents and (ii) popularity of web pages. The URL, the number of objects, and the size of each web page are shown in Table I. Figure 7 illustrates the composition of the reference web pages used in our trials.

Table I. Size and number of objects of the selected reference web pages.

Web page URL	Size (kB)	Number of objects
www.cnn.com	285.58	73
www.bahn.de	256.03	35
www.ebay.com	148.03	66
www.alps.com	431.70	33
www.t-mobile.co.uk	144.16	12
www.lastminute.com	232.39	32
www.starwars.com	368.48	90
www.samsung.com	226.96	47
www.snoopy.com	173.97	63
www.nasa.gov/home/index.html	330.43	54

DOI: 10.1002/wcm

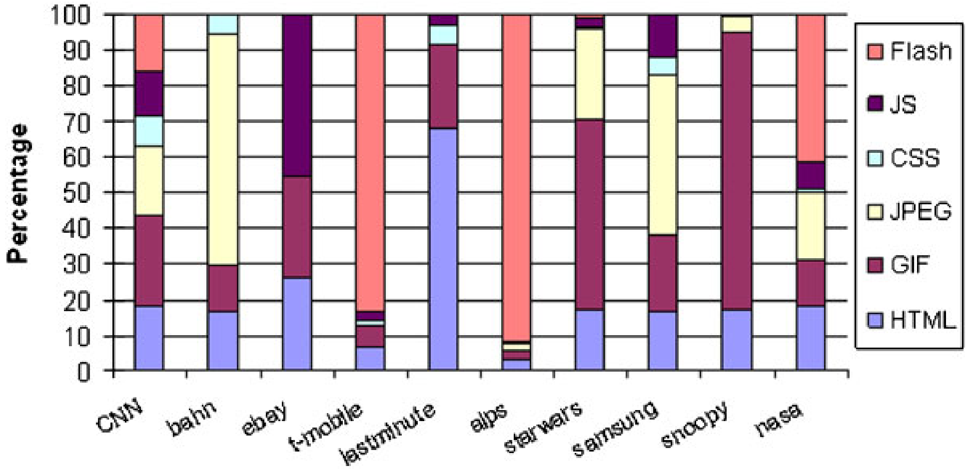


Fig. 7. Percentage of content type for each test web page.

5.4. Test Methodology

Over 300 individual trials were performed at different times during the working day. Since cellular networks tend to vary for several reasons (e.g., system load, user mobility, radio propagation impairments, etc.), we ran all modes of a particular test at the same time interval. Additionally, each individual trial was performed at least five times. Our measurements were performed with the following tools: the Ethereal 0.10.0 with WinPCap 3.0, the Microsoft Network Monitor 2.0—v5.00.943, the Hagel DuMeter 3.03 build 110, and the HDD Serial Monitor v.3.10. The main performance metrics we considered were download time (probably the one to which the user is most sensitive), the amount of IP level information exchanged (relevant for billing) and image quality (when applicable).

6. Test Results

6.1. Download Times and Exchanged Traffic Measurements

Figure 8 shows the average web page download times over GPRS scenarios (i.e., Scenario 1 and 2) for all the test cases: without the optimization system (labeled as 'Non-opt'), using NettGain in both CLS and C/S modes (labeled as 'NettGain CLS' and 'NettGain C/S' respectively) and using Macara in the same operational modes (labeled as 'Macara CLS' and 'Macara C/S' respectively). NettGain was configured with low quality image settings, while Macara was tested with the default image quality level provided, which leads to a higher subjective perception of image quality. We thus expected to obtain the range of improvement values that PEPs can achieve.

Table II provides a summary of results through a set of parameters that measure the optimization factor resulting from the use of the aforementioned PEPs in terms of download time, downlink amount of information, uplink amount of information, and perceived bandwidth.

A general result is that using PEPs in all their operational modes leads to a significant improvement in performance in all cases. As expected, the C/S approach provides higher optimization values, since it allows non-standard mechanisms specifically developed for optimization in last-hop wireless links to be used.

Trials were then performed in Scenario 3. They were designed to compare the use of NettGain in CLS and C/S modes over a UMTS link. Table III and Figure 9 summarize the results obtained, showing that communications over UMTS can be significantly optimized.

6.2. Image Quality Considerations

We performed measurements on the image quality degradation held by NettGain and Macara to reduce the amount of information to be downloaded. The results are based on a parameter known as peak signal to noise ratio (PSNR) and the comparison of original and compressed images. PSNR is a function of the mean squared error (MSE) between original and compressed images, which is computed on the luminance signal only.

DOI: 10.1002/wcm

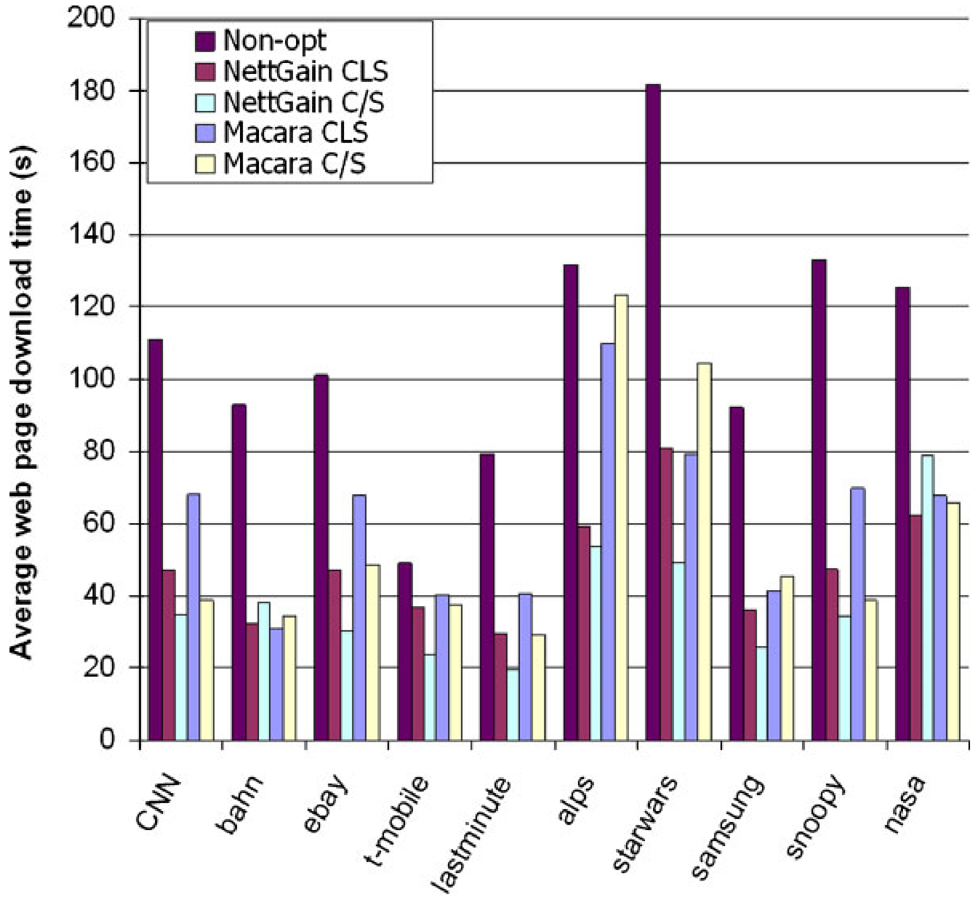


Fig. 8. Test results: average web page download times over GPRS with and without the usage of PEPs.

It must be noted that the PSNR is an objective measurement of quality, while the user's perception is subjective. The PSNR is widely accepted as a reliable means of estimating the performance of a compression mechanism. Table IV shows the results after calculation of the PSNR for a set of six sample images and a comparison of the original-sized image with compressed images for the two optimization

Table II. Measurement results of the improvement achieved by the tested PEPs on web page downloads over GPRS.

	Download time optimization factor	Downlink information optimization factor	Uplink information optimization factor	Perceived bandwidth optimization factor
NettGain CLS	2.77	3.77	1.71	2.74
NettGain C/S	4.28	5.31	4.21	4.23
Macara CLS	1.88	2.04	1.49	1.80
Macara C/S	2.18	2.36	2.83	2.05

Table III. Average web page download optimization test results with NettGain over a UMTS link.

	Download time optimization factor	Downlink information optimization factor	Uplink information optimization factor	Perceived Bandwidth optimization factor
NettGain CLS	1.81	3.58	1.62	1.91
NettGain C/S	2.41	5.02	3.82	2.43

DOI: 10.1002/wcm

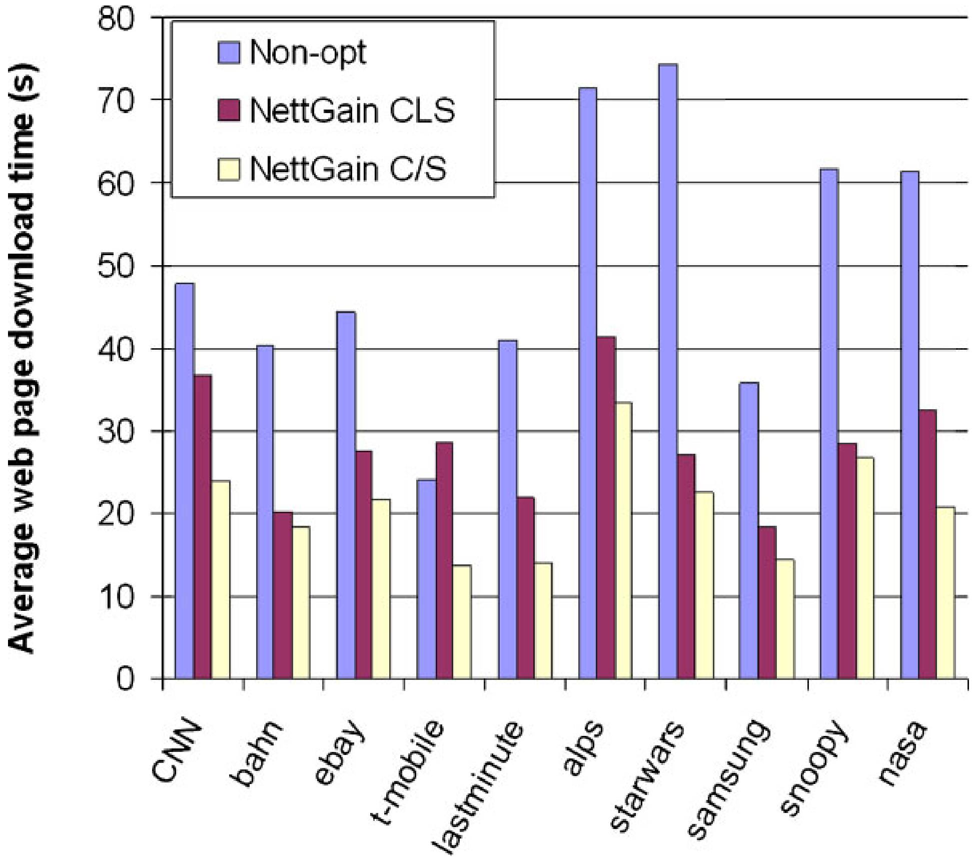


Fig. 9. Test results: average web page download times over UMTS with NettGain CLS mode, C/S mode, and without optimization system.

systems: NettGain and Macara. In this case, NettGain was configured with the high image quality option so that subjective image quality perception would be similar to that offered by Macara.

As is shown in Table IV, both systems significantly reduce image sizes and provide quite similar PSNR values (moderately higher for NettGain). PSNR and image size appear to be correlated, although different techniques may be used for compressing image content. Therefore, this rule may not always apply, as happens with image2.gif. Finally, the lossy compression factors measured were significantly higher than those published in Reference [6], in which image degradation is reported to be barely perceptible to the human eye.

6.3. Optimal Web Page Download Protocol

Once the benefits of using PEPs had been quantified, our purpose was to investigate the extent to which PEPs provide optimal performance. Our approach was to compare PEP performance with that of an ideal web

Table IV. Image quality degradation measurements for both NettGain and Macara PEPs.

Image Name	Original image size (kB)	PSNR (dB)		Compressed image size (kB)	
		Macara	NettGain	Macara	NettGain
image1.jpg	48.4	29.07	30.29	12.9	16.3
image2.gif	4.75	36.80	39.86	2.87	2.75
image3.jpg	16.0	27.98	29.38	2.01	2.47
image4.jpeg	16.0	27.74	28.48	1.56	1.99
image5.jpeg	10.0	29.07	30.10	0.75	0.91
image6.jpeg	10.3	28.84	29.58	1.30	1.55

DOI: 10.1002/wcm

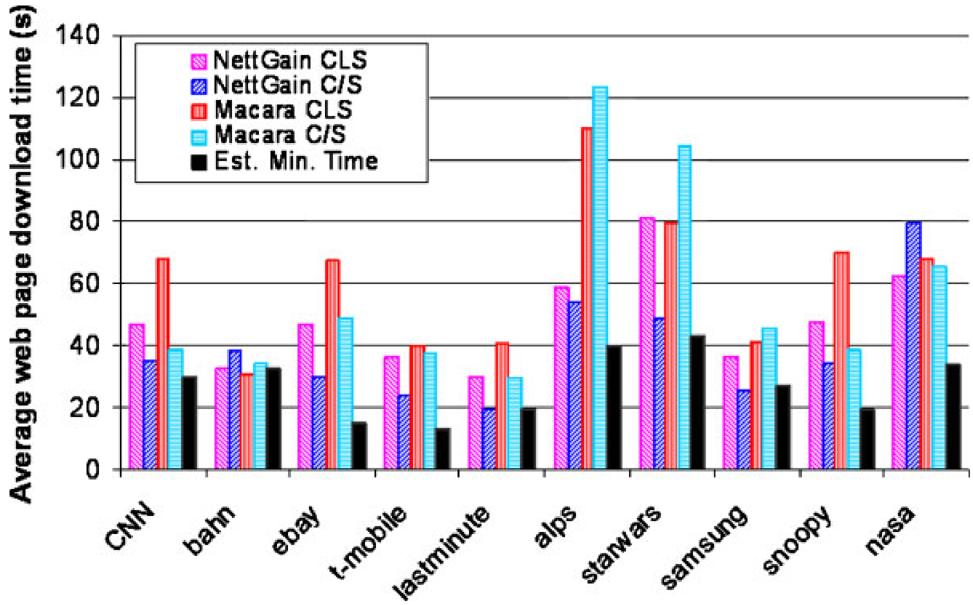


Fig. 10. A comparison of performance provided by commercial PEPs over GPRS and usage of an ideal web page download protocol.

page download protocol over 2.5G and 3G. We defined such a protocol drawing from the issues described for a standard web page download (see Section 3). This ideal protocol would consist of three phases: (i) a single DNS lookup, (ii) a request download sent by the client, and (iii) transmission of the whole compressed web page as a single object using a reliable transport protocol without congestion control mechanisms (i.e., Slow Start, Fast Recovery, etc.). Figure 3(e) shows an example of a web page download using this protocol (note that the initial DNS lookup has been removed from Figure 3 for the sake of clarity). This definition is useful for purposes of comparison, since it sets a lower bound on web page download time. Note that lossy compression gain, when applicable, is tunable.

We then compared NettGain and Macara with the above-mentioned ideal web page download protocol. We calculated web page download times by adopting the ideal web page download protocol presented, and by applying the compression factors shown in Reference [6] and a GPRS access with the downlink speed measured for 100 kB FTP transfers. Note that the initial Slow Start effect is not considered to be relevant for such a file size over GPRS, in which the Bandwidth-Delay Product (BDP) is approximately equal to 6 kB [9]. We chose the GPRS environment since trials could be performed with both PEPs in this case. Figure 10 shows a comparison of the estimated minimum web page download time with download times obtained with NettGain and Macara in all of their modes. In this case, NettGain was configured in its high quality mode for better comparison conditions, since the compression factors quoted in Reference [6] also provide high quality image compression.

As can be seen in Figure 10, NettGain C/S gives the best performance in most of the trials. Such performance is very close to that obtained with an ideal download protocol. The main result is that using a PEP may give rise to an almost optimum web browsing experience, since download times are reduced to values very close to those obtained by using an ideal webpage download protocol.

7. Conclusions

WWAN networks such as GPRS and UMTS may cause a sharp deterioration in the performance of Internet applications such as web browsing. When the TCP/IP stack was built, it was not anticipated that it would be used over wireless links, which have large RTTs that lead to a significant number of transmission delays. There are two main solutions to this problem: (i) using adequate end-to-end protocol parameters and mechanisms and (ii) interposing a PEP between the wired and wireless parts. The first approach fully respects the end-to-end argument, which is one of the premises of the Internet architecture. However, servers are by default unaware of the access technology used by a client. Therefore, optimizing communications for

DOI: 10.1002/wcm

each particular last hop technology is not possible. In addition, server design principles attempt to optimize server performance rather than user experience. Thus, end-to-end techniques cannot ensure that the best possible improvements are made. Contrarily, a PEP can be fully optimized for the specific features of a given wireless link. In many cases, this means using custom, non-standard, data delivery protocols, which break the end-to-end semantics of the communication. However, in this paper, we have shown through a set of experiments over live GPRS and UMTS networks that PEP performance may be very close to that of using an ideal web page download protocol. Finally, since current end-to-end protocols result in significant under-performance, and PEPs break the end-to-end semantics, the possibility of developing standard, high-performing end-to-end mechanisms over WWAN should be considered.

Acknowledegement

This work was supported in part by FEDER, the Spanish Government through project TIC2003-01748, DURSI, and FSE.

References

1. Brasche G, Walke B. Concepts, services and protocols of the new GSM phase 2+ general packet radio service. *IEEE Communications Magazine*, August 1997.
2. Holma H. *WCDMA for UMTS: Radio Access for Third Generation Mobile Communications*, 3rd edn. John Wiley & Sons, Inc: UK, 2004.
3. Balakrishnan H, Seshan S, Amir E, Katz R. Improving TCP/IP performance over Wireless Networks. In *Proceedings of the ACM 1st Annual International Conference on Mobile Computing and Networking (MOBICOM)*, Berkeley, USA, November 1995.
4. Bakre A, Badrinath B. Implementation and performance evaluation of indirect TCP. *IEEE Transactions on Computers* 1997; **46**(3): 260–278.
5. Brown K, Singh S. M-TCP: TCP for mobile cellular networks. *Computer Communication Review* 1997; pp. 19–43.
6. Rodriguez P, Fridman V. Performance of PEPs in Cellular Wireless Networks. In *Proceedings of the 8th International Workshop on Web Content Caching and Distribution*, Hawthorne, USA, October 2003.
7. Chakravorty R, Banerjee S, Rodriguez P, Chesterfield J, Pratt I. Performance optimizations for wireless wide-area networks: comparative study and experimental evaluation. In *Proceedings of the ACM 10th Annual International Conference on Mobile Computing and Networking (MOBICOM)*, Philadelphia, USA, October 2004.
8. Benko P, Malicsko G, Veres A. A large-scale, passive analysis of end-to-end TCP performance over GPRS. In *Proceedings of the 23rd IEEE INFOCOM'2004 Conference*. Hong Kong, March 2004.
9. Calveras A, Paradells J, Gomez C, Catalan M, Valles JC. Optimizing TCP/IP parameters over GPRS and WLAN networks. In *Proceedings of the IEEE Semiannual Vehicular Technology Conference (VTC) 2003-Fall*, Orlando, USA, October 2003.
10. Catalan M, Gomez C, Viamonte D, Paradells J, Calveras A, Barcelo F. TCP/IP analysis and optimization over a precommercial live UMTS network. In *Proceedings of the IEEE Wireless Communications and Networking Conference 2005 (WCNC'05)*. New Orleans, USA, March 2005.
11. Kohlwes M, Riihijarvi J, Mahonen P. Measurements of TCP performance over UMTS networks in near-ideal conditions. In *Proceedings of the IEEE Semiannual Vehicular Technology Conference (VTC) 2005-Spring*, Stockholm, Sweden, June 2005.
12. Kalden R, Varga T, Wouters B, Sanders B. Wireless service usage and traffic characteristics in GPRS networks. In *Proceedings of the 18th International Teletraffic Congress (ITC-18)*, 2003, Berlin, pp 981–990.
13. Berners-Lee T, Fielding R, Frystyk H. Hypertext transfer protocol—HTTP/1.0. May 1996.
14. Fielding R, *et al*. Hypertext transfer protocol—HTTP/1.1. RFC 2616, June 1999.
15. Heidemann J, Obraczka K, Touch J. Modeling the performance of HTTP over several transport protocols. *IEEE/ACM Transactions on Networking* **5**(5), October 1997.
16. Chakravorty R, Pratt I. WWW performance over GPRS. In *Proceedings of the IEEE International Conference in Mobile and Wireless Communications Networks (MWCN) 2002*. Stockholm, Sweden, September 2002.
17. Rodriguez P, Mukherjee S, Rangarajan S. Session-level techniques to improve web-browsing performance over wide-area wireless links. In *Proceedings of the IW3C2/ACM WWW Conference*, New York, USA, May 2004.
18. Inamura H, Montenegro G, Ludwig R, Gurtov A, Khafizov F. TCP over Second (2.5G) and Third (3G) Generation Wireless Networks, RFC3481, February 2003.
19. Border J, Kojo M, Griner J, Montenegro G, Shelby Z. Performance enhancing proxies intended to mitigate link-related degradations. RFC 3135, June 2001.
20. Saltzer JH, Reed DP, Clark DD. End-to-end arguments in system design. *ACM Transaction on Computer Systems* 1984; **2**(4): 277–288.
21. FlashNetworks web page: http://www.flashnetworks.com (date of access: February 2006).
22. Bytemobile web page: http://www.bytemobile.com (date of access: February 2006).
23. Chakravorty R, Clark A, Pratt I. Optimizing web delivery over wireless links: design, implementation and experiences. *IEEE Journal on Selected Areas in Communications* 2005; **23**(2): 402.
24. Go T, Moronski J, Phatak DS, Gupta V. Freeze-TCP: a true end-to-end enhancement mechanism for mobile environments. In *Proceedings of IEEE INFOCOM, Tel-Aviv*, Israel, March 2000.
25. Sinha P, Venkitaraman N, Sivakumar R, Gharghavan V. WTCP: a reliable transport protocol for wireless wide-area networks. In *Proceedings of ACM Mobicom, Seattle*, USA, August 1999.
26. Wireless profiled TCP. Version 31-March-2001. Wireless Application Protocol. WAP-225-TCP-20010331-a.
27. Sagfors M, Ludwig R, Meyer M, Peisa J. Queue management for TCP traffic over 3G links. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC) 2003*, New Orleans, USA, March 2003.
28. Liljeberg M, Alanko T, Kojo M, Laamanen H, Raatikainen K. Optimizing world-wide web for weekly-connected mobile workstations: an indirect approach. In *Proceedings of the International Workshop on Services in Distributed and Networked Environments (SDNE), Whistler, Canada, 1995*.
29. Rabinovich M, Wang H. DHTTP: an efficient and cache-friendly transfer protocol for the web. *IEEE/ACM Transaction Network* 2004; **12**(6): 1007–1020.
30. Chan MC, Ramjee R. TCP/IP performance over 3G wireless links with rate and delay variation. In *Proceedings of ACM Mobicom*, Atlanta, USA, September 2002.

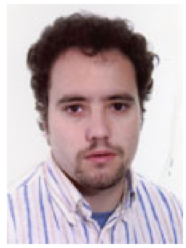
DOI: 10.1002/wcm

31. Ratnam K, Matta I. W-TCP: an efficient transmission control protocol for networks with wireless links. In *Proceedings of Third IEEE Symposium on Computers and Communications*, Athens, Greece, July 1998.
32. Chakravorty R, Katti S, Pratt I, Crowcroft J. Flow aggregation for enhanced TCP over wide-area wireless. In *Proceedings of the IEEE INFOCOM'03 Conference, San Francisco*, USA, San Francisco, USA, March 2003.
33. Meyer M, Sachs J, Holzke M. Performance evaluation of a TCP proxy in WCDMA Networks. *IEEE Wireless Communications*, October 2003.
34. Braden R. T/TCP—TCP Extensions for Transactions. RFC 1644, July 1994.
35. WAP Forum/Open Mobile Alliance web page: http://www.wapforum.org (date of access: February 2006).
36. Composite Capabilities/Preferences Profile Public home page: http://www.w3.org/Mobile/CCPP/
37. Housel BC, Samaras G, Lindquist DB. WebExpress: a client/intercept based system for optimizing Web browsing in a wireless environment. *Mobile networks and applications*. 1999; **3**: 419–431.
38. Fleming TB, Midkiff SF, Davis NJI. Improving the performance of the World Wide Web over wireless networks. In *Proceedings of Global Telecommunications Conference 1997* (GLOBECOM '97), Phoenix, USA, November 1997.
39. NettGain—Mobile Data Access Platform. In *Proceedings of the Workshop on Internet Usage over 2.5G and 3G*. Barcelona, Spain, March 2003.
40. Liljeberg M, Helin H, Kojo M, Raatikainen K. Mowgli WWW software: improved usability of WWW in mobile WAN environments. In *Proceedings of Global Telecommunications Conference 1996, (GLOBECOM'96)*, November, Las Vegas, 1996.
41. Mogul JC, Douglis F, Feldmann A, Krishnamurthy B. Potential benefits of delta encoding and data compression for HTTP. In *Proceedings of ACM Sigcomm, Cannes, France*, September 1997.

Authors' Biographies

Carles Gomez received his M.Sc. degree from the Technical University of Catalonia in 2002. He is an Assistant Professor at the same university and a member of the Wireless Networks Group (WNG). He is currently working towards his Ph.D. He has worked in several public-funded research projects. His research interests include TCP/IP optimization over WWANs and routing in ad-hoc and sensor networks.

Marisa Catalan received her M.Sc. degree from the Technical University of Catalonia in 2003. In the same year, she joined the Wireless Networks Group (WNG). She is currently pursuing her Ph.D. at the Telematics Engineering Department, at the mentioned university. Her research interests are the performance of wireless communications and the optimization of Internet protocols in 2.5G, 3G, and wireless multi-hop networks.



David Viamonte obtained his M.Sc. degree from the Technical University of Catalonia. He is currently Ph.D. candidate in the Telematics Engineering Department of the same university. His research interests include real-time and multimedia services over wireless networks. Between 2001 and 2006, he occupied several research and technical positions at Nokia and Vodafone respectively. He is currently with Genaker, involved in Research and Development of advanced SIP- based solutions.

Josep Paradells is professor at the Technical University of Catalonia. He is the head of the Wireless Networks Group (WNG). His expertise areas are network convergence and ambient intelligence, combining theoretical studies with real implementations. He has been participating in national and European public-funded research projects and collaborating with main Spanish telecommunications companies. He has published his research results in conferences and journals.

Anna Calveras-Augé obtained her Ph.D. degree from the Technical University of Catalonia in 2000. She is an Associate Professor at the mentioned university, at the Telemathics Department, and a member of the Wireless Networks Group (WNG). Her expertise area is the optimization of Internet application and transport protocols on heterogeneous network environments. She has been involved in several national and international projects. She has several communications in international and national conferences.

DOI: 10.1002/wcm