TCP Performance Issues over Wireless Links

George Xylomenos, George C. Polyzos, Athens University of Economics and Business, Greece

Petri Mähönen, University of Oulu, Finland, and VTT, Technical Research Center of Finland

Mika Saaranen, University of Oulu, Finland

Corresponding Author: George C. Polyzos, Athens University of Economics and Business, Department of Informatics, Patission 76, Athens 104 34, Greece, Tel: +30-1-6000683

Abstract

This article discusses the problems arising when the TCP/IP protocol suite is used to provide Internet connectivity over existing and emerging wireless links. Due to the strong drive towards wireless Internet access through mobile terminals, these problems must be carefully studied in order to build improved systems. We review wireless link characteristics using Wireless LANs and Cellular Communications systems as examples. We then outline the performance problems of the TCP/IP protocol suite when employed over those links, such as degraded TCP performance due to mistaking wireless errors for congestion. We present various proposals for solving these problems and examine their benefits and limitations. Finally, we consider the future evolution of wireless systems and the challenges that emerging systems will impose on the Internet protocol suite.

Introduction

The ubiquity of the Internet is, at least partly, owed to the network technology independent design of IP, the network layer protocol of the Internet, which seamlessly interconnects diverse networks into a global one. The current strong drive towards Internet access via mobile terminals, makes the inclusion of wireless systems such as *Cellular Communications* (CC) and *Wireless Local Area Networks* (WLAN) into the mainstream Internet very desirable. These systems share characteristics with both traditional wireless systems (satellite and terrestrial microwave), such as high error rates, and wired systems, such as low physical layer propagation delays. Although Internet protocol development has almost exclusively been based on wired media with decreasing error rates and increasing bandwidths, the simple services offered by IP can be easily provided even over wireless links. CC and WLAN systems however raise a multitude of performance issues, since environmental conditions and terrestrial obstructions and reflections lead to high and unpredictable error rates. In addition, cellular systems suffer from long communication pauses whenever mobile devices move between adjacent cells. In order to solve these problems, a synthesis of techniques for enhancing the performance of both wired and wireless links is required, that will also take into account the requirements of the TCP/IP protocol suite. This article presents the characteristics and performance limitations of various existing and emerging wireless systems and surveys a wide range of approaches for enhancing Internet performance over such links.

Wireless Systems

Generic Characteristics

The *delivery delay* for a link layer frame consists of *transmission delay*, i.e. frame size divided by link speed, *propagation delay*, i.e. the time the signal takes to cross the link, and *processing delay* at the sender and receiver. WLAN and CC links have similar propagation delays to wired ones, which are much lower than those of satellite links. Unlike wired links though, WLAN and CC links suffer from severe error rates, due to external interference. CC links are affected by atmospheric conditions and multipath fading due to terrestrial obstructions, while indoor WLAN links suffer from multipath fading due to furniture and people. With mobility constantly changing the error characteristics of a link, WLAN and CC error behavior can vary in a faster and more unpredictable manner than that of satellite links.

Depending on the intended application of a system, the link layer may offer either a private switched circuit service, or a shared best effort connectionless service. In order to support TCP/IP, the link layer must (at least) encapsulate IP datagrams into link frames, thus isolating higher layers from low level details. Minimalistic link layers however may be insufficient for wireless links. In voice telephony, random frame losses of 1-2% are considered reasonable as they do not cause audible speech degradation [1]. Since physical layer errors are usually clustered, randomization is achieved by interleaving and coding across several frames. Most Internet applications are not error tolerant though, thus wireless losses impose additional error recovery requirements.

The traditional Internet approach is to delegate issues such as congestion and error control to higher (end to end) layers, so as to avoid imposing the corresponding recovery overhead on all applications. While this is adequate for reliable wired links, in error prone wireless links local (link layer) error recovery can be faster and more adaptable to link characteristics. For error intolerant applications, voice oriented systems offer a *non-transparent mode* that incorporates link layer error recovery, in addition to their native *transparent mode*. Packet oriented WLAN systems may similarly provide error recovery to reduce their error rates. Non-transparent services are not a panacea though, since each application may require a different level of reliability. Furthermore, Internet protocols implementing their own error recovery schemes may interact adversely with link layer mechanisms. For example, the transport layer may retransmit delayed packets in parallel with the link layer, thus wasting wireless link bandwidth [2].

Wireless Local Area Networks

A characteristic example of WLAN systems is the Lucent WaveLAN. The original system employed either *direct sequence* (DS) or *frequency hopping* (FH) spread spectrum radios, at the 900 MHz or 2.4 GHz frequency bands. While the original bit rate was 2 Mbps, more recent WLANs offer 5.5 Mbps and 11 Mbps bit rates, with 50 Mbps versions in the design phase. The WaveLAN hardware offers an Ethernet compatible interface to higher layers, i.e. the same headers, CRCs and frame sizes are used, and a connectionless best effort service is provided. WaveLAN networks are broadcast based, using CSMA/CA to share the channel, instead of Ethernet's CSMA/CD. *Collision detection*(CD) is difficult to implement in wireless networks as it requires simultaneous transmission and reception at the same band, hence *collision avoidance* (CA) is employed instead.

Transmission and propagation delays are low due to the small coverage area and high system bandwidth. The system is robust in the presence of narrowband interference and obstructions within its operating range. Typical frame loss rates are less than 2.5% using maximum sized frames. Due to timing differences between desktop and laptop cards, their throughput is not symmetric [3]. Host processing power also affects throughput and frame loss between heterogeneous hosts. Synchronization may lead to excessive collisions during bidirectional communication [4]. A receive threshold mechanism is offered to isolate adjacent WaveLAN networks, but no power control is provided. Newer WLANs support multiple frequency bands, to avoid interference between adjacent networks.

To achieve interoperability between WLAN devices supplied by different vendors, the IEEE designed the 802.11 standard. Enhancements over the WaveLAN include support for acknowledgments and retransmissions, contention free transmission using reservations, and an operating mode where a master host provides WLAN co-ordination. The original standard specified radios working in the 2.4 GHz frequency band with 1 or 2 Mbps bit rates, in both DS and FH spread spectrum versions. Subsequently, two new standardization projects were initiated to provide higher speeds. 802.11a uses a high speed (OFDM) physical layer in the 5 GHz frequency band, providing bit rates ranging between 6 and 54 Mbps. 802.11b was developed to increase bit rates over the existing physical layer. Commercial 802.11b solutions provide either 5.5 Mbps or 11 Mbps bit rates, using the 2.4 GHz frequency band.

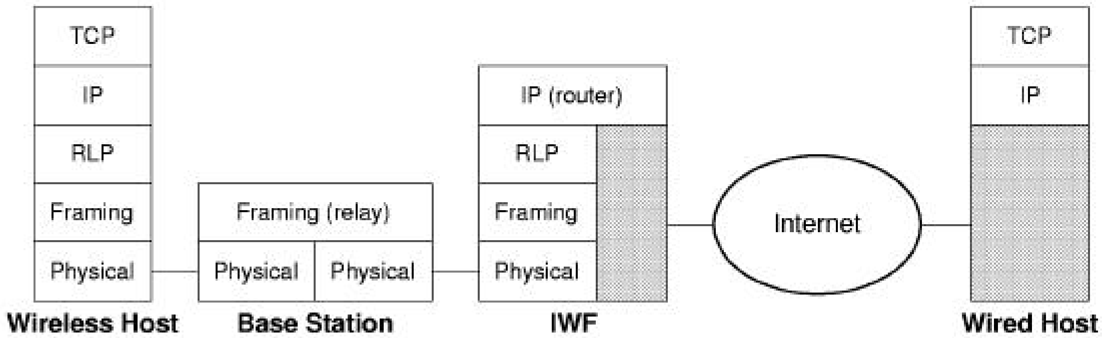

Figure 1: Connectivity between CC systems and the Internet

Cellular Communications Networks

Current CC systems are characterized by modest bit rates, small frames, and circuit mode operation. They use either TDMA (GSM and IS-54) or CDMA (IS-95) to share the medium. Frames may carry either encoded voice or higher layer data. Compared to WLANs, CC systems exhibit higher delays due to the lower bit rates and longer distances involved. The outdoor CC environment is also harsher, with multipath fading caused by buildings and hills. Frame loss rates of 1-2% [1] are not detrimental to voice quality as long as they seem random. This is achieved by bit interleaving, which considerably increases processing delay. CC systems are interconnected to other networks using an *Interworking Function* (IWF) [5]. The IWF provides digital to analog conversions to interface with analog networks and rate adaptation/frame conversions to interface with ISDN. In order to interoperate with packet networks, the IWF uses a *Radio Link Protocol* (RLP) to communicate with the mobile. The RLP may support IP datagram segmentation and reassembly [1], thus providing transparent Internet connectivity, and error recovery, thus hiding wireless losses from the Internet [6]. Figure 1 shows the part of the IWF which serves as an Internet gateway, located between the CC system and the Internet.

GSM (TDMA) offers 9.6 Kbps full rate channels. The non-transparent mode RLP uses 240 bit frames. It employs *Selective Repeat* (SR) ARQ, causing the native bit error rate of 10^{-3} to be reduced to 10^{-8}, at the expense of variable throughput and delay due to retransmissions [5]. IS-54 (TDMA) supports 9.6 Kbps full rate channels. The non-transparent mode RLP uses an advanced ARQ scheme with 256 bit frames. Each frame separately acknowledges multiple consecutive frames. The sender keeps track of the order of frame (re)transmissions, so that when a frame is acknowledged, all unacknowledged frames transmitted before it can be assumed lost, since the link preserves the transmission sequence [6]. IS-95 (CDMA) supports 8.6 Kbps full rate channels. The non-transparent mode RLP uses 172 bit frames [1]. Network layer packets are first encapsulated into variable size PPP frames and then segmented into fixed size RLP frames. This combines the convenience of variable sized packets with the efficient error recovery of fixed size frames. Only negative acknowledgments are used to reduce control overhead. Frames not received after a few retransmissions are dropped, thus trading off reliability for limited delay variance. The residual packet loss rate thus becomes 10^{-4}.

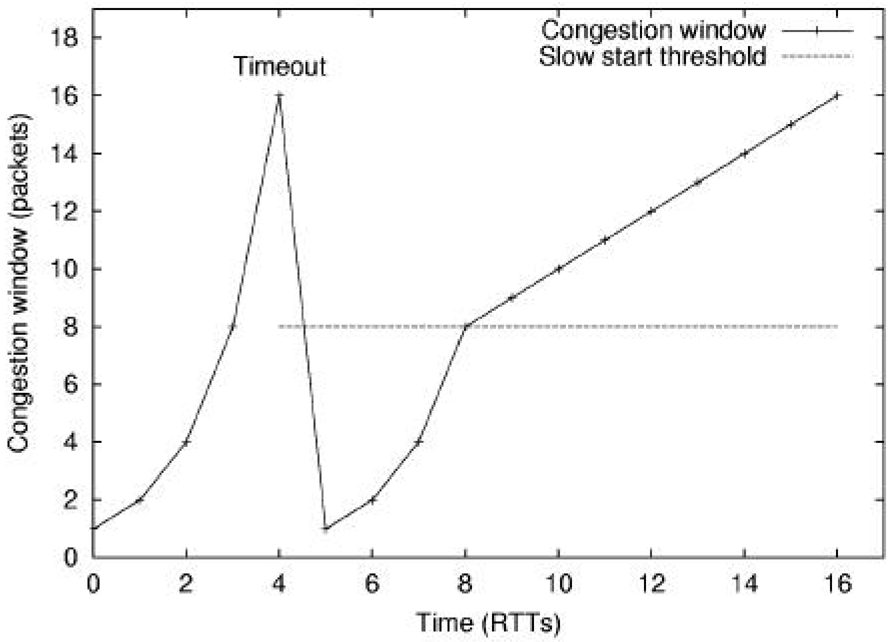

Figure 2: TCP congestion window dynamics

TCP Performance over Wireless Links

TCP Fundamentals

The most popular transport layer protocol on the Internet is TCP, which offers a reliable byte stream service. TCP provides transparent segmentation and reassembly of user data and handles flow and congestion control. TCP packets are cumulatively acknowledged as they arrive in sequence, with out of sequence packets causing duplicate acknowledgments to be generated. The sender detects a loss when multiple duplicate acknowledgments (usually 3) arrive, implying that the next packet was lost. IP may reorder datagrams, thus TCP cannot immediately assume that all gaps in the packet sequence signify losses. When the session becomes idle or acknowledgments are lost, TCP detects losses using timeouts. Retransmission timers are continuously updated based on a weighted average of previous round trip time (RTT) measurements. Accuracy is critical, since delayed timeouts slow down recovery, while early ones may lead to redundant retransmissions. A prime concern for TCP is congestion. Congestion occurs when routers are overloaded with traffic that causes their queues to build up and eventually overflow, leading to high delays and packet losses. Since most Internet traffic is carried by extremely reliable wired links, TCP assumes that *all* losses indicate congestion. Therefore, when losses are detected, besides retransmitting the lost packet, TCP also reduces its transmission rate, allowing router queues to drain. Subsequently, it gradually increases its transmission rate so as to gently probe the network's capacity.

TCP maintains a *congestion window*, which is an estimate of the number of packets that can be in transit without causing congestion. New packets are only sent if allowed by both this window and the receiver's *advertised window*. The congestion window starts at one packet, with new acknowledgments causing it to be incremented by one, thus doubling after each RTT. This is the *slow start* phase (exponential increase). In Figure 2 slow start stops after 4 RTTs when a loss is detected by a timeout. A *slow start threshold* is then set to half the value of the congestion window, the congestion window is reset to one packet, and the lost packet is retransmitted. Slow start is repeated until the threshold is reached after 3 RTTs, allowing routers to drain their queues. Subsequently, the congestion window is incremented by one packet per RTT. This is the *congestion avoidance* phase (linear increase). When losses are detected by duplicate acknowledgments, indicating that subsequent packets have been received, TCP retransmits the lost packet, halves the congestion window, and restarts with the congestion avoidance phase. This description is based on TCP Reno, see [7] for more details on the various TCP variants. Multiple losses may repeatedly reduce the slow start threshold, causing the slower congestion avoidance phase to take over immediately, leading to large throughput degradations.

TCP Performance

The TCP assumption that all losses are due to congestion becomes quite problematic over wireless links. The WaveLAN suffers from a *frame error rate* (FER) of 1.55% when transmitting 1400 byte frames over an 85 ft distance, with clustered losses [3]. Reducing the frame size by 300 bytes halves FER, but increases framing overhead. In shared medium WLANs, forward TCP traffic (data) contends with reverse traffic (acknowledgments). In the WaveLAN this can lead to collisions that dramatically increase FER [4]. Mobility also increases FER for the WaveLAN by about 30% [3]. File transfer tests over a WaveLAN with a nominal bandwidth of 1.6 Mbps achieved a throughput of only 1.25 Mbps [3]. This 22% throughput reduction due to a FER of only 1.55% is caused by the frequent invocations of congestion control mechanisms which repeatedly reduce TCP's transmission rate. If errors were uniformly distributed rather than clustered, throughput would increase to 1.51 Mbps [3].

CC links in transparent (voice) mode suffer from a residual FER of 1-2% (after low level error recovery), despite their short frames [1]. A full rate IS-95 link would segment a 1400 byte IP datagram into 68 frames. Assuming independent frame errors, the probability of a successful packet transmission is 50.49% at a FER of 1%. Frame errors are less bursty than bit errors, because multiple frames are bit interleaved before transmission. Coding and interleaving reduce the loss rate and randomize frame errors, thus avoiding audible speech degradation, but increase processing delay due to de-interleaving after reception. Shorter IP datagrams face fewer errors but suffer from increased header overhead. TCP/IP header compression may be used over slow CC links, shrinking TCP/IP headers to 3-5 bytes. Unfortunately, header compression is incompatible with network layer encryption and may adversely interact with TCP error recovery and link layer resets, causing entire windows of TCP data to be dropped [8].

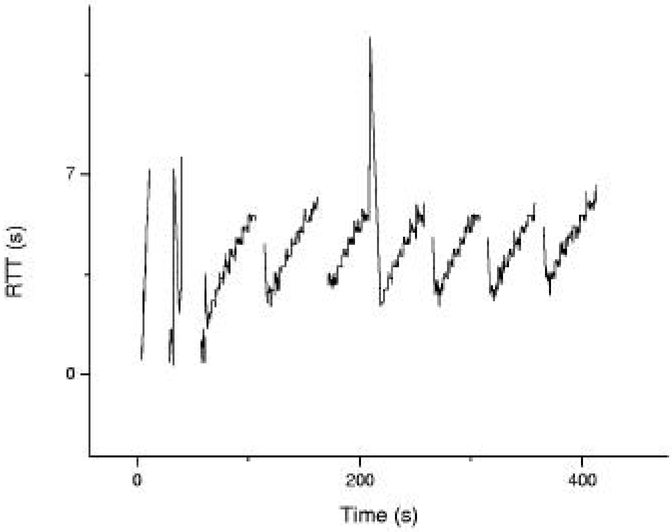

Figure 3: TCP RTT behavior over GSM

While the non-transparent mode RLP used by GSM usually recovers from losses before TCP timers expire [8], it suffers from high and widely varying RTT values. Measurements using `ping` over a GSM network in San Francisco showed that 95% of the RTT values were around 600 ms with a standard deviation of 20 ms [9]. Our measurements with `ping` over GSM networks in Oulu, Helsinki and Berlin, produced similar results with higher standard deviations. Large file transfer experiments however, reveal that RTT can be much higher (up to 12 seconds) with real applications over operational networks. Figure 3 shows RTT measurements from a commercial GSM network in Oulu, Finland, during a file transfer. RTT values consist of processing time, the 2×150 ms delay of the GSM channel, plus 250-1250 ms and 35 ms to transmit a packet and its acknowledgement, respectively. This high latency is due to interleaving, rate adaptation, buffering and interfacing between GSM network elements [9].

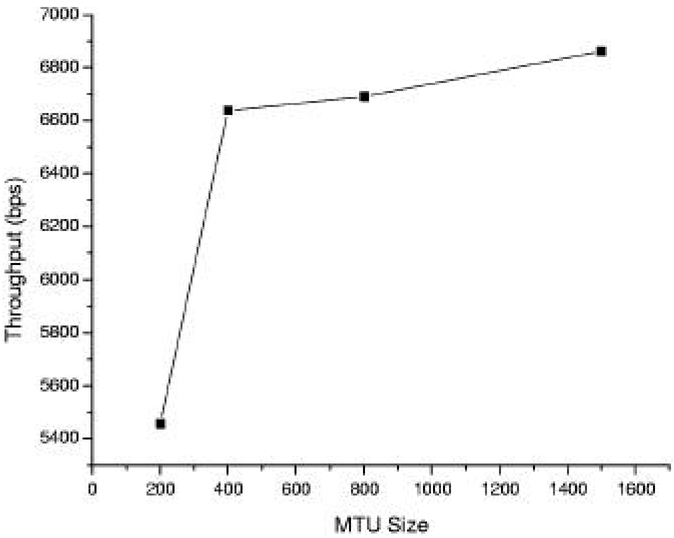

Figure 4: TCP performance against MTU size over GSM

Increasing the TCP *maximum transfer unit* (MTU) size reduces TCP/IP header overhead, thus improving throughput, but also increasing the interactive response time. Figure 4 shows typical throughput as a function of TCP MTU size in an operational GSM network. TCP throughput is maximized for a MTU size of 720 bytes in our experiments. Our measurements also show that TCP over GSM suffers from occasional disruptions of 6-12 seconds, which are due to RLP level disruptions that last for a couple of seconds. Analysis of this problem suggests that some IP datagrams are buffered and later released out of sequence, a phenomenon that appears in operational GSM networks but is rarely simulated or encountered in small test networks. Disruptions are also caused by link resets which occur when a RLP frame cannot be transmitted after a few retries, or when a serious protocol violation occurs. This causes the sender and receiver sequence numbers to be reset and flushes all buffers, meaning that in practice the GSM RLP is *not* fully reliable. To reduce the number of resets, the maximum number of retransmissions (by default 6) can be increased during connection setup [8]. Throughput may also be increased by adapting RLP frame size. Although small frames simplify RLP operation and make it more robust in worst case channel conditions, choosing a frame size appropriate for prevailing conditions may lead to increased throughput.

	Without Wireless Errors	**With Wireless Errors**	**% Achieved**
LAN	1.5 Mbps	0.70 Mbps	46.66%
WAN	1.35 Mbps	0.31 Mbps	22.96%

Table 1: TCP throughput over LAN and WAN connections

	Physical Bit Rate	**TCP Throughput**	**% Achieved**
IEEE 802.11	2 Mbps	0.98 Mbps	49%
IEEE 802.11b	11 Mbps	4.3 Mbps	39.1%

Table 2: TCP throughput over IEEE 802.11 LAN connections

When end-to-end paths include multiple wireless links, the accumulated losses further reduce throughput, also causing underutilization of wireless links, an important issue for circuit switched CC links. Furthermore, when a TCP packet is lost after crossing some wireless links in the path, its retransmission has to cross them again, thus wasting bandwidth. Losses have more pronounced effects on paths with higher end-to-end delay which require TCP to maintain large transmission windows to keep data flowing. On such paths TCP also suffers from *spurious timeouts*, that is, timeouts which would be avoided if the sender waited longer for acknowledgments. CC systems explicitly allow prolonged disconnections during handoffs, thus causing spurious timeouts. Another problem, *spurious fast retransmits*, occurs when packets are reordered beyond the TCP duplicate acknowledgement threshold, an occasional event with the GSM RLP. Table I shows TCP throughput over a LAN path (a single WLAN) versus a WAN path (a WLAN plus 15 wired links) [10]. We first show throughput in the absence of any losses, and then when the WLAN suffers from independent frame losses at a FER of 2.3% for 1400 byte frames. Table II shows throughput over a single link path, using either an IEEE 802.11 or an IEEE 802.11b WLAN. Higher speed links are affected more by losses, since TCP takes longer to reach its peak throughput after each loss.

TCP Performance Enhancements

Transport Layer Solutions

The degraded performance of TCP over wireless links is mostly due to mistaking wireless losses for congestion. Thus, numerous proposals for appropriate TCP modifications exist. During handoffs in CC systems, packets may be delayed or even lost. Recovery from these losses should be initiated right after handoff completion, without waiting for a timeout. TCP can achieve this by receiving appropriate signals from lower layers [11]. Alternatively, TCP can exploit mobility *hints* from lower layers to heuristically distinguish losses due to handoffs. For these losses, TCP can avoid halving the slow start threshold during recovery, thus skipping the congestion avoidance phase. Another approach is for wireless link endpoints to *choke* TCP senders during handoffs, by transparently closing the receiver's advertised window [12]. The sender then freezes all pending timers and starts periodically probing the receiver's window. Shrinking the advertised window however violates TCP guidelines.

After handoffs, congestion avoidance helps probe the capacity of the new link. With other wireless losses though, retransmissions are sufficient for recovery. Since end-to-end retransmissions are slow, TCP connections may be *split* using as pivot points routers connected to both wireless and wired links [13]. End-to-end connections are thus decomposed into separate TCP sessions for the wired and wireless parts of the path. Another protocol optimized for error recovery may be substituted over the wireless links. Split schemes violate end-to-end TCP semantics, since acknowledgments may reach the sender before data packets reach their destination. To preserve TCP semantics, acknowledgments must be delayed, thus reducing throughput. Pivot points face significant overhead, since packets undergo TCP processing twice, and considerable per connection state is maintained there.

The *Eifel* scheme modifies TCP so as to avoid the spurious timeouts and fast retransmits due to handoffs or delayed link layer retransmissions [14]. Since these problems are due to TCP's inability to distinguish between acknowledgments for original packet transmissions and retransmissions, Eifel adds TCP timestamps to outgoing packets. Timestamps are echoed in acknowledgments, thus allowing spurious timeouts to be easily avoided, without changing TCP semantics. The end-to-end TCP recovery however is not accelerated. While TCP enhancement schemes would be attractive if only the endpoints needed modifications, in practice additional changes are needed. Some approaches require signaling from lower layers to detect handoffs. Others require software to be installed and state to be maintained at pivot points. In addition, split TCP schemes need alternative, TCP compatible, protocols to be deployed over wireless links for more efficient error recovery.

Link Layer Solutions

Instead of modifying TCP, we may hide wireless losses from it. In CC systems this is achieved by non-transparent mode RLPs. Another solution is to perform local error recovery (a link layer task) at the IP level, as in *Snoop TCP* [10]. Snoop tracks TCP data and acknowledgments by maintaining state for each TCP connection traversing a pivot point. Snoop caches unacknowledged TCP packets and uses the loss indications conveyed by duplicate acknowledgments, plus local timers, to transparently retransmit lost data. It hides duplicate acknowledgments indicating wireless losses from the TCP sender, thereby preventing redundant TCP recovery. Snoop exploits the information present in TCP packets to avoid link layer control overhead. It outperforms split TCP schemes [10], without violating TCP semantics. It also avoids conflicting local and TCP retransmissions [2] by suppressing duplicate TCP acknowledgments.

Snoop requires the TCP receiver to be located right after the pivot point. If a wireless host is sending data to a remote receiver, TCP acknowledgments are returned too late for efficient recovery, and they may even signify congestion losses. In this situation, *Explicit Loss Notification* (ELN) is needed for TCP to distinguish between congestion and wireless losses. If the Snoop agent detects a non congestion related loss, it sets an ELN bit in TCP headers and propagates it to the receiver, which echoes it back to the sender. Snoop can use queue length information to heuristically distinguish congestion from wireless errors. When receiving an ELN notification, the TCP sender retransmits the lost packet without invoking congestion control. Although ELN is applicable to most topologies, it requires changes to router algorithms. Also, a lost packet can only be retransmitted after a round trip time has elapsed, when an acknowledgment with the ELN bit set is returned.

CC system RLPs avoid the layering violations of Snoop, which examines TCP headers at the IP level, but they may retransmit data in parallel with TCP [2]. This however occurs rarely with fully reliable RLPs [8] and it is prevented by RLPs that abandon error recovery after some failed attempts [1]. Link layer schemes operate at the local level with low round trip delays that allow fast recovery, in contrast to TCP modifications. Their main limitation is that they offer a single level of recovery, which may not be appropriate for all higher layer protocols and applications.

Wireless System Evolution and TCP

The trend for CC systems is to provide increased speeds and better support for packet data services. The highest data rates will be offered in small areas, or *microcells*, where user densities are higher. The *High Speed Circuit Switched Data* (HSCSD) system is a GSM extension providing bit rates of up to 56 Kbps by reserving multiple TDMA slots for each data circuit. The *General Packet Radio Service* (GPRS) is a packet switched GSM extension. GPRS supports bit rates of up to 171 Kbps via dynamic TDMA slot reservation. Current implementations provide 20-40 Kbps of user throughput. Experiments show that Internet packet loss rates will be around 2%. The third generation European CC system, UMTS, is based on wideband CDMA, supporting both circuit and packet switched modes, at various bit rates. Phase one includes services similar to GPRS, providing bit rates of up to 384 Kbps, with forthcoming phases promising up to 2 Mbps in limited areas. In the USA, the GSM EDGE/IS-136 HS system will provide bit rates of 270-722 Kbps, or even over 2 Mbps in limited areas.

Many short range (in room) systems, or *Personal Area Networks* (PANs), have been designed for low bit rates, such as Bluetooth, a FH spread spectrum system providing bit rates of 400-700 Kbps. While Bluetooth should provide TCP performance similar to low end WLANs, there are serious problems concerning its radio link level interoperability with IEEE 802.11. The IEEE 802.15 project which specifies a PAN standard based on Bluetooth is working on this issue. For very high speeds, the *Local Multipoint Distribution System* (LMDS) will offer broadband *fixed* wireless Internet access using the 28 or 40 GHz frequency bands. LMDS is a *Wireless Local Loop* (WLL) system providing 1-2 GHz of bandwidth to fixed hosts. LMDS uses powerful link layer FEC schemes, and we have found that it can reliably carry TCP traffic [15].

The trend for WLAN systems is to provide higher speeds while also supporting mobility between adjacent networks, with each network essentially becoming a *microcell*. At the other end of the spectrum, sparsely populated areas can be covered by terrestrial or satellite systems using very large cells, or *macrocells*. Since increasing the number of cells for a given area means more expensive infrastructure, different systems will employ different cell sizes to achieve their goals. TCP/IP support will allow all these wireless systems to interoperate by becoming parts of the Internet. The next step is to provide direct interoperability between wireless systems by allowing users to transparently move not only between cells within the same system, but also from one system to another, depending on the services and coverage available. In these unified *hierarchical* cellular systems, large cells will be overlaid by multiple smaller cells in areas with increased user concentrations.

Since handoffs momentarily disrupt connectivity with adverse effects on TCP performance, hierarchical cellular systems must be carefully designed to avoid increasing the gravity of handoff induced problems. The small area and high data rates of microcells will lead to more frequent handoffs and potentially increased losses during each handoff. Handoffs between different systems may also dramatically change the performance of underlying wireless links. To reduce the magnitude of these problems, the key is to exploit co-operation between layers so as to enable protocols to adapt their behavior as needed. Intensive research is directed towards adaptive link layers that provide information to higher layers in an orderly fashion. The European Union WINE project is studying protocol adaptivity and link dependent configuration so as to optimize IP performance over wireless links, without exposing lower layer details to TCP. A protocol enhancing proxy approach has been developed, the *Wireless Adaptation Layer* (WAL), to handle automatic adaptivity. The emerging software radios, which allow the configuration of physical and link layer parameters in real time, will further enhance link adaptivity, hence protocol adaptivity will become even more important in the future.

Summary

We have discussed the performance problems that arise when using TCP over wireless links. We presented the characteristics of various wireless systems and then explained how these characteristics adversely interact with TCP mechanisms. We explained the causes of these problems and gave examples of their magnitude. We outlined and evaluated various TCP performance enhancements which focus on either the transport or the link layer. Finally, we discussed future directions for wireless system evolution and the challenges they will present with respect to TCP performance.

Acknowledgments

AUEB authors gratefully acknowledge the support of the General Secretariat of Research and Technology of Greece. VTT authors gratefully acknowledge the support of the Academy of Finland, TEKES and the VTT Wireless Internet Laboratory. We would also like to thank our colleagues on various projects for the measurements and joint results reported here, especially Giuseppe Coppola (Philips), Diego Melpigiano (Philips), Tommi Saarinen (VTT), Timo Sukuvaara (VTT) and Zach Shelby (VTT).

Author Biographies

George Xylomenos (xgeorge@aueb.gr) holds a Diploma in Informatics (1993) from the Athens University of Economics and Business, Greece, and M.S. (1996) and Ph.D. (1999) degrees in Computer Science from the Department of Computer Science and Engineering at the University of California, San Diego. He is currently a post-doctoral researcher at the Athens University of Economics and Business. His research focuses on improving the performance of Internet applications over wireless links and providing Quality of Service over wireless and mobile networks.

George C. Polyzos (polyzos@aueb.gr) received the Dipl. E.E. from the National Technical University, Athens, Greece, and the M.A.Sc. in E.E. and the Ph.D. in Computer Science from the University of Toronto, Canada. He is now a Professor at the Department of Informatics at the Athens University of Economics and Business. Before that he was a Professor at the Department of Computer Science and Engineering at the University of California, San Diego, where he was co-director of the Computer Systems Laboratory, member of the Center for Wireless Communications, and Senior Fellow of the San Diego Supercomputer Center. His research interests include wireless and mobile communications and computing, Internet technologies, and networks and multimedia systems design and performance evaluation.

Petri Mähönen, Ph.D. (Petri.Mahonen@oulu.fi), is director of research at the University of Oulu, Center of Wireless Communications, and research professor of telecommunications in VTT. In VTT he has been a founding scientific director of the Wireless Internet Laboratory initiative, and head of networking research. He is also currently a principal investigator for two large European Union research projects. His research interests include wireless networks, Internet protocols, super-large networks, applied physics, cognitive radios, and adaptive networking algorithms.

Mika J. Saaranen (Mika.Saaranen@ieee.org) received his M.Sc. degree in computer technology from the university of Oulu in 1993. He is currently working at the University of Oulu, Center for Wireless Communications. Before this he was working at VTT (Technical Research Center of Finland). His research interests include the areas of embedded and wireless Internet. The work includes aspects of how TCP/IP and existing neighboring lower protocol layers are affected by mobility and transmission errors of the wireless channel. On the other hand, embedded and minimized TCP/IP implementations are part of the research.

References

[1] P. Karn, "The Qualcomm CDMA digital cellular system," *Proceedings of the USENIX Mobile and Location-Independent Computing Symposium*, Aug. 1993, pp. 35-39.
[2] A. DeSimone, M.C. Chuah, and O.C. Yue, "Throughput performance of transport-layer protocols over wireless LANs," *Proceedings of the IEEE GLOBECOM '93*, Dec. 1993, pp. 542-549.
[3] G.T. Nguyen, R.H. Katz, B. Noble, and M. Satyanarayanan, "A trace-based approach for modeling wireless channel behavior," *Proceedings of the Winter Simulation Conference*, Dec. 1996, pp. 597-604.
[4] G. Xylomenos and G.C. Polyzos, "TCP and UDP performance over a wireless LAN," *Proceedings of the IEEE INFOCOM '99*, pp. 439-446, March 1999.
[5] T. Alanko, M. Kojo, H. Laamanen, M. Liljeberg, M. Moilanen, and K. Raatikainen, "Measured performance of data transmission over cellular telephone networks," *Computer Communications Review*, vol. 24, no. 5, Oct. 1994, pp. 24-44.
[6] S. Nanda, R. Ejzak, and B.T. Doshi, "A retransmission scheme for circuit-mode data on wireless links," *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 8, Oct. 1994, pp. 1338-1352.
[7] W. Stevens, "TCP slow start, congestion avoidance, fast retransmit, and fast recovery algorithms," Internet Request For Comments 2001, Jan. 1997.
[8] R. Ludwig, B. Rathonyi, A. Konrad, K. Oden, and A.D. Joseph, "Multi-layer tracing of TCP over a reliable wireless link," *Proceedings of the ACM SIGMETRICS '99*, June 1999, pp. 144-154.
[9] R. Ludwig and B. Rathonyi, "Link Layer Enhancements for TCP/IP over GSM," *Proceedings of the IEEE INFOCOM '99*, April 1999, pp. 415-422.
[10] H. Balakrishnan, V.N. Padmanabhan, S. Seshan, and R.H. Katz, "A comparison of mechanisms for improving TCP performance over wireless links," *Proceedings of the ACM SIGCOMM '96*, Aug. 1996, pp. 256-269.
[11] R. Cáceres and L. Iftode, "Improving the performance of reliable transport protocols in mobile computing environments," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 5, June 1995 pp. 850-857.
[12] K. Brown and S. Singh, "M-TCP: TCP for mobile celullar networks," *Computer Communications Review*, vol. 27, no. 5, Oct. 1997, pp. 19-43.
[13] A. Bakre and B.R. Badrinath, "Implementation and performance evaluation of Indirect-TCP," *IEEE Transactions on Computers*, vol. 46, no. 3, March 1997, pp.260-278.
[14] R. Ludwig and R.H. Katz, "The Eifel algorithm: making TCP robust against spurious

retransmissions," *Computer Communications Review*, vol. 30, no. 1, Jan. 2000, pp. 30-36.
[15] P. Mähönen, E. Russo, and M. Carlomagno, "40 GHZ LMDS-System architecture development," *Proceedings of the ICT '98*, June 1998, pp. 422-430.