Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT•

ELSEVIER

Computer Communications 26 (2003) 1990–1997

computer communications

www.elsevier.com/locate/comcom

Mobility support in hybrid wireless/IP networking

Jon W. Mark*, Jianping Pan, Sherman X. Shen

Centre for Wireless Communications, Department of Electrical and Computer Engineering, University of Waterloo, 200 University Avenue West, Waterloo, Ont., Canada N2L 3G1

Received 21 January 2003; accepted 21 January 2003

Abstract

The ramifications of Transmission Control Protocol (TCP)/Internet Protocol (IP) networking with mobile hosts and wireless links in a hybrid wireless/IP environment are examined. Two issues that affect the performance of TCP/IP are mobility support in the network layer and coping with packet errors in TCP flow/congestion control in the transport layer. The operational features of agent-assisted, router-oriented and hybrid strategies for mobility support are identified and discussed. Packet errors due to transmission through the wireless channel and handover induced by user mobility can pose problem for the conventional TCP flow control in the transport layer. Strategies to cope with errors in packets that enter the TCP connection are described and discussed.

Keywords: Mobility; Transmission control protocol; Performance degradation

1. Introduction

Global information delivery for mobile users necessitates an information transport platform consisting of a wireless front-end and a wireline backbone network. The wireless front-end offers the flexibility for supporting user roaming while the backbone wireline network provides global coverage. As a wireline network, the Internet has universal appeal. Fig. 1 depicts two possible hybrid wireless/Internet networking scenarios. The operation of the Internet is essentially overseen by the Internet Protocol (IP) in the network layer and the Transmission Control Protocol (TCP) in the transport layer of the protocol stack.

TCP/IP-based protocols, services, and applications have gained remarkable success with the Internet growth in the last two decades. It is expected that these functions can still run properly and efficiently in a heterogeneous networking environment.

The operation challenges and performance degradations that TCP/IP meets in this environment have received a lot of attention in the last few years. The purpose of this paper is to present an in-depth exposition of recent efforts that introduce mobility support and mitigate the performance degradation due to residual transmission errors in the TCP/IP protocol stack, particularly in the wireless access scenario depicted in Fig. 1(a). The treatment of mobility management strategies and the emphasis on accounting for packet loss due to events other than network congestion in TCP flow control are distinguishing features of the current paper compared to other exposition-oriented papers in the literature, e.g. [1–4].

Mobility support and coping with packet errors due to transmission and handover are critical issues in end-to-end information delivery over a hybrid wireless/Internet network. Section 2 describes and discusses certain strategies for supporting host mobility. As a flow/congestion control mechanism, conventional TCP lacks the ability to cope with errors already present in the packets entering the TCP connection. Strategies for coping with packet errors that occurred before the packets enter the TCP connection are described in Section 3. Concluding remarks are given in Section 4.

2. Host mobility support

Each user in a wireless/Internet network is allocated a single Internet address. This address resides in the mobile user's home network. When the mobile user migrates to a foreign network, a suitable strategy to maintain connection

* Corresponding author. Tel.: +1-519-888-4567x2532; fax: +1-519-746-3077.
E-mail address: jwmark@bbcr.uwaterloo.ca (J.W. Mark).

0140-3664/03/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/S0140-3664(03)00163-4

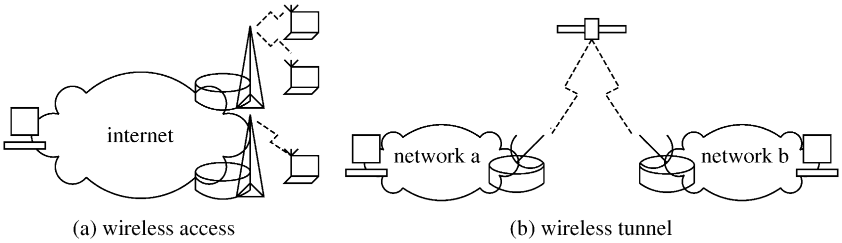


Fig. 1. Wireless/Internet interworking scenarios.

continuity is needed. *Agent-assisted*, *router-oriented* and a hybrid of both agent-assisted and router-oriented are three prominent strategies for maintaining connection continuity.

2.1. Agent-assisted strategy

The most easy-to-find and feasible strategy is to introduce some additional agents at the edge of the Internet, e.g., in the mobile's home network or visited networks, and to leave the current Internet routing fabric untouched. These agents normally assist packet exchanges between an mobile host (MH) and its correspondent host (CH) nodes, and some agents can further involve in the control interaction that keeps the MH tractable no matter when and to where it moves.

To keep the Internet routing fabric totally untouched, one possible approach within this strategy is to associate another IP address with an MH. Intuitively, one (*home address*) is assigned by the mobile's home network for identification purpose, and the other (*foreign address*) is assigned by the currently visited network for routing purpose. Both addresses are routing compatible, i.e. containing the location-dependent network prefix. The coherency between these two addresses is maintained by the mapping at a Home Agent (HA) residing in the mobile's home network. The Internet Engineering Task Force's (IETF) Mobile IP (MIP) and its follow-on schemes [5,6] utilize this approach. In the IETF MIP scheme, as shown in Fig. 2(a), mobility is absolutely transparent to a mobile's correspondents who still send packets to the mobile's home address. These packets are routed normally towards the mobile's home network, and are then intercepted by the HA. These packets are further encapsulated in an IP-in-IP format by the HA and sent to the mobile's foreign address. The foreign address usually identifies a Foreign Agent (FA) in the visited network which assists the packet exchange and control interaction, e.g., ICMP-based agent discoveries and UDP-based mobility registrations, between an MH and its HA. Finally the FA forwards the decapsulated packets to the MH. Outgoing packets from the MH are routed to its CH directly according to the normal Internet routing fabric.

The agent-assisted strategy has an obvious and critical advantage, i.e., nothing changed in or imposed on the routers. The schemes within this strategy can be deployed incrementally in the current Internet architecture, and only a few specially engineered agents are needed at the network edge and in the user premises. However, the overhead associated with control interactions among an MH and its FA and HA, e.g., agent discovery and mobile registration, is, unavoidable. Sometimes this overhead is comparatively high and not acceptable. Furthermore, when a mobile is engaged in handover signaling, incoming packets might get lost. Another obvious deficiency with this strategy is the sub-optimal triangle routing for packets sent by the CH. Even when a CH is very close to its MH, all packets in the MIP scheme or the first packet in the Loose Source Routing (LSR) scheme has to be unnecessarily forwarded to the HA first. This brings a potential operation fate-sharing and a possible performance bottleneck at the HA.

To mitigate this overhead, some follow-on schemes introduce a hierarchical or regional [6] agent layout (Fig. 3(a)), or a local anchor [7] (Fig. 3(b)) to decrease the interaction frequency between an MH and its HA. Therefore, when a mobile only hands over locally to its neighbors, the mobile registration, as well as the associated authentication and authorization, with its remote HA is not necessary and can be done locally. The incoming packet losses during a handover may be further decreased by defining a local and controlled multicast group for the adjunct FA nodes, or by letting the old FA forward the packets to the new FA which the mobile currently associates.

IETF also proposes an enhanced MIP scheme with Routing Optimization (MIP/RO) [6], largely avoids the triangle routing problem in the MIP scheme. In MIP/RO, as shown in Fig. 2(c), a HA still maintains the address mapping

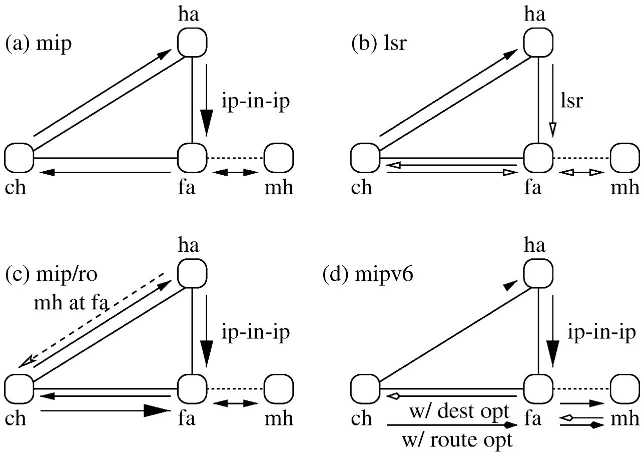
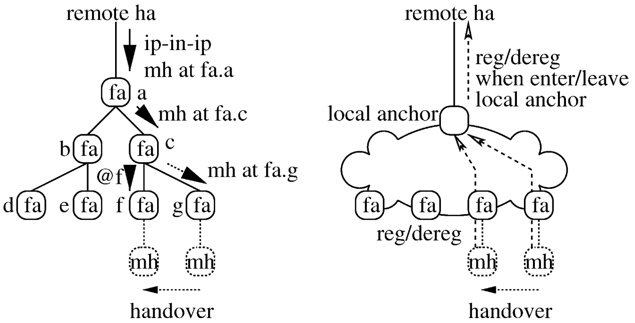


Fig. 2. Schemes with agent-assisted strategy.



(a) fa hierarchy (b) local archor

Fig. 3. Hierarchical agent and local anchor.

and further informs the CH of the mobile's foreign address. Then the MIP/RO-aware CH can tunnel the subsequent packets to the MH directly. IPv6 is the next generation protocol. Similar to IPv4, IPv6 also needs additional mobility support, and is denoted as MIPv6 [6]. MIPv6 eliminates the need for an FA, since IPv6 has several configuration schemes for a mobile to be assigned an address automatically. MIPv6 avoids triangle routing and packet tunneling, as IPv6 supports the Destination and Routing options. As shown in Fig. 2(d), an MH can inform its CH a Destination option, which is placed as a Routing option in outgoing packets from the CH. Therefore, subsequent packets, without any encapsulation from the CH, can reach the MH optimally. MIPv6 works quite similarly to the LSR scheme, but the LSR option in IPv4 is not widely supported and is rarely implemented correctly. There are other MIP-based schemes and extensions [6] that try to achieve low latency and fast handover, and to eliminate packet losses when a mobile moves in different networking scenarios.

2.2. Router-oriented strategy

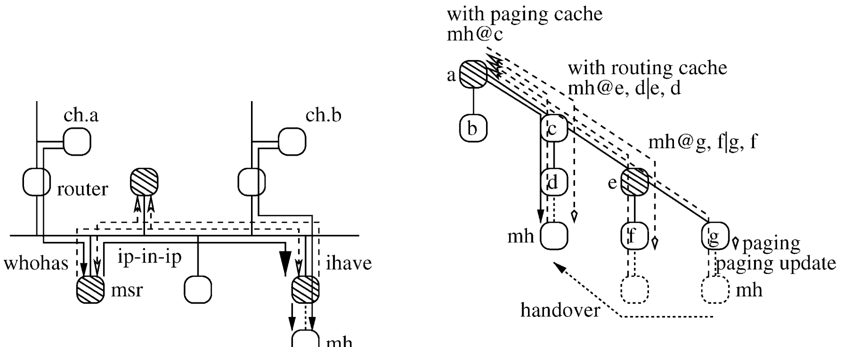
Contrary to putting additional agents at the network edge, another feasible strategy is to introduce mobility support directly at the routers for MHs, and to keep the original Internet addressing scheme untouched. The enhanced router can be scattered at the edge or in the core of the global Internet, or populated within a controlled network. These routers mainly forward the packets to the mobile by tunneling, hop-by-hop forwarding, or other means, and some enhanced routers may further involve in the mobility management (Fig. 4).

One possible approach within the router-oriented strategy is to let some routers keep track of MHs, i.e., knowing which router the MH currently associates, and partially modify the route for packets to MHs. This knowledge can be maintained and shared among a group of enhanced routers. The most attractive advantage for the router-oriented strategy is that, in most situations, incoming packets to MHs are always routed optimally.

As mentioned previously, cache maintenance is another unavoidable tradeoff. Also, it is obvious that this strategy has some difficulties on the incremental deployment over the global Internet. Indeed, most schemes within this strategy are designed for special systems and impose certain requirements on the underlying link topology and technology. Therefore, these schemes can be further optimized for their intended networking scenarios. To lower the cache maintenance overhead, the Cellular IP (CIP) scheme introduces routing and paging caches for active and idle users separately. Paging caches, usually in large amount but updated slowly, help CIP-aware routers to locate both active and idle MHs, while the routing caches, normally in small amount but updated frequently, are used only to forward incoming packets to active MHs. Given the fact that most of the time, MHs are idle and do not generate or accept any traffic, the total overhead to maintain the routing and paging caches can be hopefully minimized.

2.3. Hybrid strategy

Hybrid strategy, as its name implies, combines the features in the aforementioned two strategies. The schemes within this strategy not only introduce some agents at the network edge, but also rely on some enhanced routers in the core network. These agents function similarly as the ones used in the agent-assisted schemes to assist the packet forwarding and control interaction with MHs. These enhanced routers, similar to the ones used in



(a) mobile*ip (b) cellular ip

Fig. 4. Schemes with route-oriented strategy.

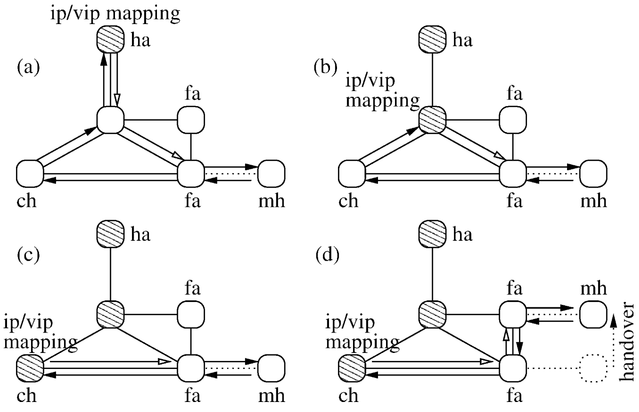


Fig. 5. Schemes with hybrid strategy.

the router-oriented schemes, also maintain the special routing caches to optimally forward packets to MHs.

Virtual IP (VIP) [8] utilizes this strategy by introducing a VIP address, which is compatible with the original IP address, and tracking the MHs current location. If a VIP-aware CH has the MHs VIP address, the packet is routed optimally as Fig. 5(c) shows. Otherwise, the packet goes to the MHs HA first with the MHs original IP address. If this packet meets a VIP-aware router that has a valid address mapping for this MH, it then goes to the MH directly, as Fig. 5(b) shows. Eventually, the unresolved packet reaches the MHs HA and it is updated with the MHs VIP address, as Fig. 5(a) shows. If the MH hands over to a new FA and the old FA is informed, this packet is further forwarded to the MHs current FA, as Fig. 5(d) shows. All forwarding paradigms are realized by updating the virtual destination in data packets, which can be implemented as an add-on IP option and is transparent to any VIP-unaware router. The routing cache in routers maintains an Addressing Mapping Table (AMT) by snooping data packets and by signaling the Create or Invalidate AMT (CAMT/InvAMT) messages.

The VIP scheme is applicable to any network with an arbitrary link topology, including the global Internet. The VIP scheme can achieve optimal or near optimal routing in many situations, as shown in Fig. 5(b) and (c). However, with the introduction of a new IP header or a new IP option, the employment of the VIP scheme and VIP-aware routers over the global Internet is quite complex. If no VIP-aware router is employed, the VIP scheme is almost the same as the two-addresses scheme with the agent-assisted strategy. If all routers are VIP-aware, it becomes very similar to the router-oriented schemes. Therefore, the gain that a scheme with the hybrid strategy can obtain is a balance between those with the agent-assisted and router-oriented strategies.

Table 1 summarizes and compares the strategies described in this section to support the host mobility in the global Internet and cellular systems. It is found that the MIP scheme is a strong candidate for macro mobility support in the global Internet, and the CIP and other similar schemes are currently competing to support micro mobility within cellular systems. There is increasing interest to integrate the MIP scheme into 3G cellular systems [9,10]. However, cellular systems have their own link layer mobility support through the established Home and Visitor Location Registers (HLR/VLR). Extending the MIP scheme to support intra and inter-domain handover is another possibility [11].

3. Coping with packet errors

Although wireless channel impairments are to be taken care of by the physical and link layer functions, the residual errors can nevertheless cause performance degradation to higher layers, especially TCP performance at the transport layer. As mentioned earlier, TCP assumes that every packet loss is due to network congestion. However, in a hybrid wireless/wireline network, residual transmission errors from the wireless segment and during mobile handovers can cause packet loss. In what follows, we examine appropriate strategies for coping with packet losses during transit from

Table 1
Mobility support strategies

Strategies	Schemes	Features	Problems	Improvements	Applications
Agent-assisted	MIP	Edge only, easy to deploy	Triangle routing, registration/ tunnel overhead, conflict with firewall, handover loss	MIP/RO, local anchor, reverse tunnel, FA forward/mcast	Global Internet, macro mobility
	LSR	No IP-in-IP tunnel, near optimal routing	LSR not widely supported, UDP: no system LSR reverse		
	MIPv6	Almost no IP-in-IP tunnel, near optimal routing	Registration overhead, conflict with foreign firewall	Similar to MIP	IPv6 Internet
Router-oriented	Mobile*IP	One address solution	MSR query/reply overhead		LAN/CAN
	CIP	Hop-by-hop routing, one address solution	Cache overhead, topology constrains	Page/route cache	Cellular systems, micro mobility
Hybrid	VIP	No IP-in-IP necessary, hopefully optimal routing	New IP header/option, updates in core routers		

Table 2
Symbols used in end-to-end flow/congestion control

Symbol	Description
ack	Acknowledgement
dupack	Duplicate acknowledgement
cwnd	Congestion window
rrt	Round-trip time
rto	Timeout
ssthresh	Slow Start threshold
seqno	Sequence number

the TCP sender to the TCP receiver that are caused by errors already in the packets entering a TCP connection.

To facilitate discussions on end-to-end flow control, a number of symbols are used in the sequel. For easy reference, the meanings of the symbols used are tabulated in Table 2.

3.1. Effect of residual transmission errors

Packets entering the TCP connection already containing errors will affect TCP performance. These errors can cause packet loss at the TCP receiver, which interprets all packet losses as caused by network congestion. Wireless transmission errors usually occur in bursts, and TCP might experience multiple and consecutive packet losses. TCP may use either a sender timeout or a multiple duplicate acknowledgement (*dupack*) strategy to detect packet losses. Usually a triple-*dupack* event is used to trigger Fast Retransmit after detecting the first lost packet (in TCP Tahoe and Reno), or after detecting the first few lost packets (in TCP NewReno and Selective Acknowledgement (SACK) [13]). For the subsequent lost packets, conservative timeouts are the only solution. Furthermore, the frequently invoked congestion control algorithms keep the Additive Increase and Multiplicative Decrease (AIMD) regulated *cwnd* always very small; most of the time, *cwnd* is too small for the receiver to return enough *dupacks* [14]. Therefore, the TCP sender may repeatedly experience exponential *rto* backoffs, which would keep itself and the receiver unnecessarily idle, resulting in lower application throughput, increased end-to-end delay, and poor wireless utilization.

3.2. TCP flow/congestion control

TCP is a window flow control mechanism. The behavior of TCP is depicted in Fig. 6. To provide reliable end-to-end service, the TCP sender advances and adjusts its sliding window to regulate the amount of in-flight data based on the control signals feed back from the receiver. The sender also tracks the round-trip time (*rrt*) of a packet and its acknowledgement (*ack*), and calculates the timeout (*rto*) accordingly. If no *ack* arrives within the *rto* interval for a particular packet, it is assumed lost and retransmitted using an exponential backoff approach.

As shown in Fig. 6(a), when the network load exceeds a certain threshold, the performance gain collapses in the operating region III. Here the gain, which is proportional to the ratio of throughput over delay, is maximized when TCP is in regions I and II. Slow-Start and Congestion Avoidance schemes were developed in the late 1980s to address these problems (see, e.g. [12]).

With congestion control, initially or after an idle period, TCP conservatively probes the network capacity from a minimum congestion window (*cwnd*) and exponentially increases the *cwnd* until it exceeds a Slow Start threshold (*ssthresh*), as Fig. 6(b) shows. TCP then continuously probes the network with linear increment. The maximum amount of in-flight data is regulated by the smallest *cwnd* and the flow control window. On the assumption that all packet losses are due to network congestion while retransmitting (see Fig. 6(c)), the TCP sender sets *ssthresh* to one-half of the current *cwnd* and throttles *cwnd* to its initial size when a sender timeout occurs.

TCP congestion control has been widely deployed in Internet applications. Fast Transmit and Fast Recovery [12]

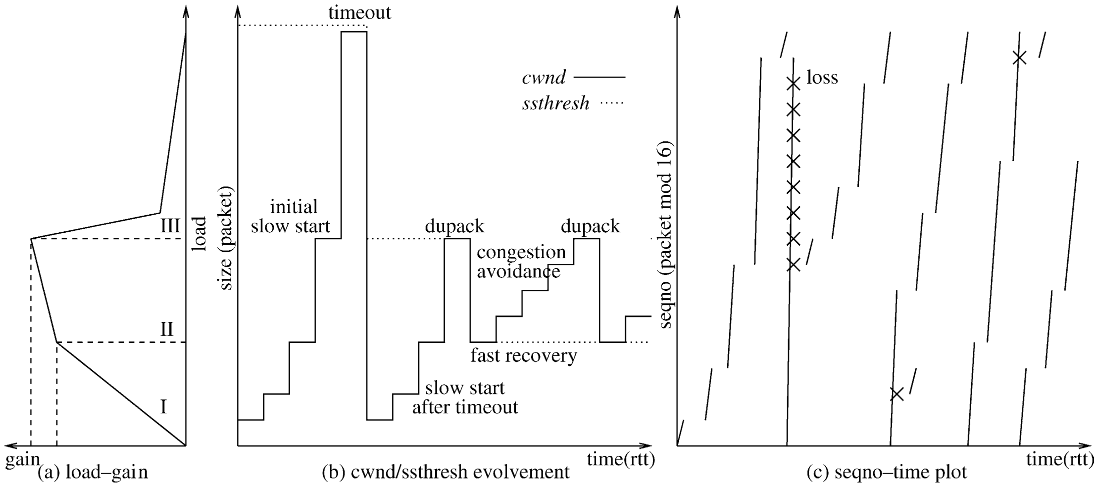


Fig. 6. TCP congestion control.

are two notable congestion control schemes. Since the receiver only acknowledges in-order packets cumulatively, out-of-order packets will trigger the sending of duplicate acknowledgments (*dupack*). With the assumption that a *dupack* implies at least one missed packet in the receiver buffer, when the number of *dupacks* exceeds a preset threshold (usually three in modern TCP variants), the most unacknowledged packet is assumed lost and retransmission takes place. The value of *ssthresh* is set to one-half of the *cwnd* value, and *cwnd* is either throttled to the initial size or decreased to the new *ssthresh* value, depending on whether Fast Retransmit or Fast Recovery is invoked in TCP.

Ordinary TCP may be enhanced by augmenting it with strategies that try to take errors inherent in the packets entering the TCP connection into consideration. The strategies considered in this section include *Endpoint-Oriented*, *Intermediate-Assisted* and *Edge-Router Screening*.

3.3. Endpoint-oriented strategy

With the *Endpoint-Oriented Strategy*, the engineering is usually performed at the TCP receiver. The usually approach is just reuse the ordinary TCP flow and congestion control mechanisms. To trade an efficient Fast Retransmit off a coarse-grained deficient timeout, Triggered Fast Transmit in [15] forces an MH to return several redundant *dupacks* after a handover. If there are packet losses during the handover, the *dupack* triggers the TCP sender to retransmit the lost packets earlier. To give the lower layer error recovery more opportunities, the Delay Duplicated Acknowledgment scheme [16] adds an additional delay for the receiver to return the third *dupack*. If the packet finally goes through and reaches the receiver before that delay expires, a new *ack* can be returned and the unnecessary retransmission can be avoided. Furthermore, when an MH is about to handover or expects severe packet losses, it can advertise a zero window [17]. The sender with a zero window goes into a persistent mode that suspends packet transmissions and retransmissions, and freezes the congestion controls and related timers. A frozen sender can be resumed through a non-zero window update after the MH finishes its handover gracefully.

The ordinary TCP congestion control mechanisms are not efficient in handling multiple and consecutive packet losses. Another approach within the endpoint-oriented strategy is to further extend and customize these control algorithms. When the TCP receiver acknowledges the received packets positively and cumulatively, it is difficult for the sender to determine how many and which packets are lost. SACK, either positive or negative, can be used as an add-on scheme. With SACK, the sender has the ability to recover multiple packet losses and avoid the unnecessary retransmission for the already received packets. Also, SACK can be further extended through Forward Acknowledgment (FACK) [19] or Duplicated SACK (D-SACK) [18]. In the case of FACK, the highest received *seqno* is identified; in the case of D-SACK, the duplicated received packets is identified. With these information and its own sending history, the sender can avoid and possibly undo the unnecessarily invoked retransmissions and congestion regulations.

Most TCP variants rely on a triple-*dupack* to triggered Fast Retransmit. However, when TCP flows over lossy links, with the frequently invoked congestion controls, *cwnd* will eventually become very small. With a small *cwnd* and possible consecutive multiple packet losses, most of the time it is impossible for the receiver to return enough *dupacks*, and the sender behavior is dominated by the deficient coarse-grained timeout. The Segment-in-Flight Estimation (*Sifest*) scheme proposed in [14] enables the sender to have more knowledge on the amount of in-flight packets and consequently reacts to *dupacks* more intelligently and adaptively. The *Sifest* scheme helps the loss-friendly sender to avoid unnecessary timeouts, which would keep itself unreasonably idle, and therefore brings higher application throughput and lower end-to-end delay. This scheme is also applicable when the TCP sender is session limited, e.g. conveying HTTP traffic, even when the *cwnd* is comparatively large. There are other schemes which adopt this approach, e.g., some guesswork to discriminate the source of congestion or corruption losses at the sender, according to the packet loss pattern, their relations with *cwnd* and *ssthresh*, or other heuristic hints [20–22].

3.4. Intermediate-assisted strategy

As mentioned previously that an FA could relay and assist the packet forwarding and control interactions for an MH in the network layer. An easy-to-develop strategy is one that introduces more functionalities at the FA, or at any wireless/wireline interworking node in the transport layer to mitigate the residual transmission errors from the physical layer. For notational convenience, the term ‘intermediate’ is used in this paper to represent *a wireless/wireline interface node*. Intermediate-assisted strategies include *Implicit snooping* and *Explicit notification*.

3.4.1. Implicit snooping

Implicit snooping is an approach that introduces some TCP helpers or agents at the intermediate, but are invisible to both the TCP sender and receiver. The control unit is a TCP packet or a TCP segment. A representative scheme that adopts this approach is Snoop [28]. The snoop agent buffers data packets and monitors their *acks*. If a packet is not acknowledged by the MH within snoop’s fine-grained local timer or the first *dupack* is returned and the packet is still in snoop’s buffer, it will be resent locally. The snoop agent also filters out the *dupack* if a local retransmission is feasible.

3.4.2. Explicit notification

Explicit notifications are performed using feedback, as shown in Fig. 7. The transmission link is characterized using

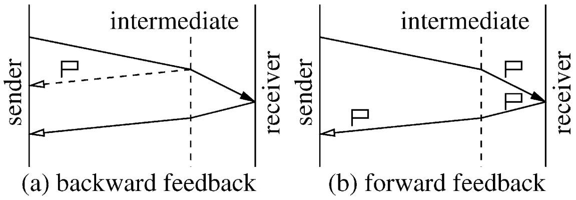
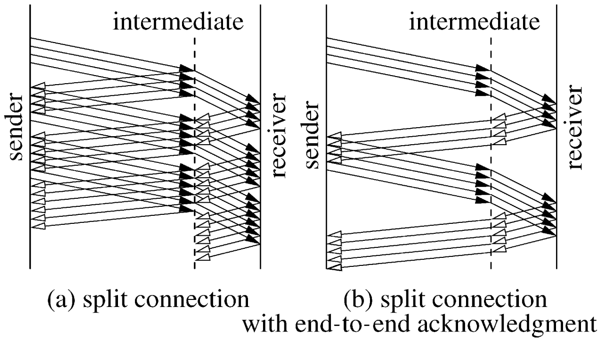


Fig. 7. Backward and forward feedback.

a two-state model: a good state and a bad state. Explicit Bad State Notification (EBSN) [30] is a backward feedback scheme, as shown in Fig. 7(a) shows. When the BS anticipates the channel is about to be in the bad state, an ICMP-like message is sent back to the TCP sender, which intentionally resets the sender timer. Without a timeout, congestion control algorithms are also frozen. FREEZE [23] is a similar scheme in which a zero windows is returned in an explicit message and the sender is in the persistent mode until being resumed explicitly. With Explicit Loss Notification (ELN) [24] similar to the forward feedback Explicit Congestion Notification (ECN) [29] shown in Fig. 7(b), the intermediate marks the data packets when burst losses are anticipated, and the receiver extracts the mark and sets a flag in *ack* packets.

In a *Split Connection* TCP approach the intermediate is the node joining the two halves. One ordinary TCP connection is established between the CH and the intermediate, and the other TCP or equivalent transport connection links the intermediate and the mobile. Indirect TCP (I-TCP) [26,27] is a representative scheme that adopts this approach. The two TCP connections are joined at the intermediate. Therefore, the transport layer flow, error, and congestion control can be performed separately over the wireless and wireline segments. The intermediate immediately acknowledges the packets it receives, as Fig. 8(a) shows, no matter when these packets are actually received by the receiver. Therefore, it is the responsibility of the intermediate to ensure reliable and in-order delivery to the final receiver.

The Split Connection approach should break the original end-to-end TCP reliability. The main criticism of split connection TCP is the lack of end-to-end semantics. However, end-to-end semantics can still be maintained to a certain degree through end-to-end acknowledgment, as Fig. 8(b) shows. WTCP [25] is a scheme that adopts this approach. The intermediate acknowledges the packet only when it receives the *ack* from the receiver. Therefore, end-to-end reliability is retained. WTCP assumes that both TCP endpoints use the Timestamp option for an accurate *rtt* measurement. To avoid the impact of local retransmission based on *rtt* estimation and *rto* calculation, the packet's Timestamp is updated according to its dwelling time in the intermediate.



(a) split connection

(b) split connection with end-to-end acknowledgment

Fig. 8. Split connection with/without end-to-end acknowledgment.

3.4.3. Edge router screening strategy

Although the physical layer functions may be quite sophisticated, residual transmission errors at the output of the cell-site receiver and packet loss due to handover signaling can have a tremendous impact on the flow control performance using TCP and TCP variants discussed earlier. The fundamental reason for this is that conventional TCP window flow control is strictly based on packet losses due to network congestion. Corrupted packets entering the core network can trigger the shrinking of the window size in the same way as packet losses due to network congestion.

With TCP flow control, the end-points of a TCP connection are the TCP sender and the TCP receiver. If packet errors inherent in the arrival traffic to the edge router (the intermediate) are screened and detected, they can be prevented from entering the network proper. A virtual TCP receiver may be constructed to co-locate with the TCP sender at the intermediate. On the assumption that error correction is performed at the link layer, the virtual TCP receiver only needs to emulate the error detection capability of the real TCP receiver.

The virtual TCP receiver interrogates the incoming traffic for errors. Packets with uncorrected errors are dropped without passing them to the TCP sender. In this way, traffic flows entering the network proper would be free of errors due to transmission and handover. The incorporation of virtual TCP receivers will undoubtedly increase the complexity at the edge router. Since the virtual TCP receiver does not have to buffer any data, the additional burden would not be overly complex.

4. Conclusions

The paper examines the design strategies, possible approaches, and proposed schemes for TCP/IP networking with MHs and wireless links in a hybrid wireless/wireline networking environment. IP-based mobility support is first addressed by an overview on IP functionalities and the challenges with MHs. Agent-assisted, router-oriented, and hybrid strategies to support the IP mobility are examined by discussing and comparing possible approaches and proposed schemes within each strategy. Attention is given to their design features, potential gains, underlying trade-offs, possible problems, and target applications. It is shown that MIP is a strong candidate for global mobility support, while a number of different schemes are competing to support micro mobility in cellular systems. Given the fact the Internet is the most pervasive global backbone and the cellular systems are the capacity-enhanced wireless infrastructure, the question now facing the research community is the following: How should MIP be extended to cellular systems, or how should MIP be interfaced with other cellular-oriented schemes, while still meet the requirements of security, reliability, and accountability?

Transport protocol performance, particularly for TCP, is then addressed by an overview on TCP congestion control algorithms and their impacts due to wireless impairments. Endpoint-oriented, intermediate-assisted, and edge-router screening strategies to mitigate the TCP performance degradation are examined. In each strategy, attention is given to the possible approaches and proposed schemes with their underlying rationales and protocol mechanisms, as well as the gains and trade-offs. It becomes clear that to improve end-to-end network performance, efforts in different protocol layers and at different entities among the connection path must be harmonized. Breaking the traditional layer transparency, e.g., by using inter-layer signaling, and breaking the end-to-end semantics, e.g., by employing intermediate proxy or agents, might be beneficial for some networking scenarios. More importantly, to what extent the layer transparency and end-to-end semantics can be broken requires careful consideration. Given the fact that TCP-transported applications dominate the current Internet traffic load, and a similar pattern is emerging in cellular systems, the question is: How should the TCP mechanisms and the lower layer cellular radio link control (RLC) protocols and radio resource management be harmonized, and to provide consistent and efficient transport services to upper layer interactive applications?

Acknowledgements

This work has been supported in part by a grant from the Canadian Institute for Telecommunications Research (CITR) under the NCE program of the Government of Canada, and in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant No. RGPIN7779.

References

[1] P. Bhagwat, C. Perkins, S. Tripathi, Network layer mobility: an architecture and survey, IEEE Personal Commun. 3 (3) (1996) 54–64.

[2] C. Barakat, E. Altman, W. Dabbous, On TCP performance in a heterogeneous network: a survey, IEEE Commun. Mag. 38 (1) (2000) 40–46.

[3] H. Balakrishnan, V.N. Padmanabhan, S. Seshan, R.H. Katz, A comparison of mechanisms for improving TCP performance over wireless links, IEEE/ACM Trans. Networking 5 (6) (1997) 756–769.

[4] K. Pentikousis, Small TCP in wired-cum-wireless environments, IEEE Commun. Surv. 3 (4) (2000) 2–12.

[5] C.E. Perkins, Mobile IP, IEEE Commun. Mag. 35 (5) (1997) 84–99.

[6] IETF Mobile IP Working Group, IP routing for wireless/mobile hosts (mobileip), http://www.ietf.org/html.charters/mobileip-charter.html, 2001

[7] J. Zhang, J.W. Mark, A local anchor scheme for mobile IP, Proc. P&Q Net'2000 (2000) 137–156.

[8] F. Teraoka, K. Uehara, H. Sunahara, J. Murai, VIP: a protocol providing host mobility, Commun. ACM 37 (8) (1994) 67–75.

[9] B. Sarikaya, Packet mode in wireless networks: overview of transition to third generation, IEEE Commun. Mag. 38 (9) (2000) 164–172.

[10] R. Ramjee, T.F. La Porta, L. Salgarelli, S. Thuel, K. Varadhan, L. Li, IP-based access network infrastructure for next-generation wireless data networks, IEEE Personal Commun. 7 (4) (2000) 34–41.

[11] S. Das, A. Misra, P. Agrawal, TeleMIP: telecommunications-enhanced mobile IP architecture for fast intradomain mobility, IEEE Personal Commun. 7 (4) (2000) 50–58.

[12] M. Allman, V. axson, W.R. Stevens, TCP congestion control, IETF RFC 2581 (1999)

[13] M. Mathis, J. Mahdavi, S. Floyd, A. Romanow, TCP selective acknowledgment option, IETF RFC 2018 (1996).

[14] J. Pan, J.W. Mark, X. Shen, TCP performance and improvement over wireless links, Proc. IEEE GlobeCom'2000 (2000) 62–66.

[15] R. Caceres, L. Iftod, The effect of mobility on reliable transport protocols, Proc. 14th IEEE ICDCS (1994) 12–20.

[16] N.H. Vaidya, M. Mehta, C. Perkins, G. Montenegro, Delayed duplicated acknowledgments: a TCP-unware approach to improve performance of TCP over wireless, Texas A&M University TR-99-003 (1999).

[17] T. Goff, J. Moronski, D.S. Phatak, V. Gupta, Freeze-TCP: a true end-to-end TCP enhancement mechanism for mobile environments, Proc. 19th IEEE InfoCom'2000 (2000) 1537–1545.

[18] S. Floyd, J. Mahdavi, M. Mathis, M. Podolsky, An extension to the selective acknowledgement (SACK) option for TCP, IETF RFC 2883 (2000).

[19] M. Mathis, J. Mahdavi, Forward acknowledgment: refining TCP congestion control, Proc. ACM SIGCOMM'96 (1996) 281–291.

[20] C. Parsa, J.J. Garcia-Luna-Aceves, Differentiating congestion vs. random loss: a method for improving TCP performance over wireless links, Proc. IEEE ICNP'2000 (2000) 90–93.

[21] N.K.G. Samaraweera, Non-congestion packet loss detection for TCP error recovery using wireless links, IEE Proc.: Commun. 146 (4) (1999) 222–230.

[22] D. Bansal, A. Chandra, R. Shorey, An extension of the TCP flow control algorithm for wireless networks, Proc. IEEE ICPWC'99 (1999) 207–210.

[23] A. Chan, D.H.K. Tsang, S. Gupta, Impact of handoff on TCP performance in mobile wireless computing, Proc. IEEE ICPWC'97 (1997) 184–188.

[24] H. Balakrishnan, R. Katz, Explicit loss notification and wireless web performance, Proc. IEEE Globe-Com Internet Miniconf. (1998).

[25] K. Ratnam, I. Matta, WTCP: an efficient mechanism for improving TCP performance over wireless links, Proc. IEEE ISCC'98 (1998) 74–78.

[26] A. Bakre, B.R. Badrinath, I-TCP: indirect TCP for mobile hosts, Proc. IEEE ICDCS'95 (1995) 136–143.

[27] Y. Zeng, J.W. Mark, X. Shen, Indirect RSVP for virtual cluster cellular mobile IP networks, Proc. IFIP Networking'2000 (2000) 338–349.

[28] H. Balakrishnan, S. Seshan, E. Amir, R.H. Katz, Improving TCP/IP performance over wireless networks, Proc. ACM MobiCom'95 (1995) 2–11.

[29] S. Floyd, TCP and explicit congestion notification, ACM Compos. Commun. Rev. 24 (5) (1994) 10–23.

[30] S. Bakshi, P. Krishna, N.H. Vaidya, D.K. Pradhan, Improving performance of TCP over wireless networks, Proc, IEEE 17th ICDCS'97, (1997) 365–373.