Towards Transport-layer Mobility: Evolution of SCTP Multihoming

Ł. Budzisz [a,*], R. Ferrús [a], A. Brunstrom [b], K.-J. Grinnemo [b,1], R. Fracchia [c], G. Galante [d,2], F. Casadevall [a]

[a] *Dept. of Signal Theory and Communications (TSC), Universitat Politecnica de Catalunya (UPC), C/. Jordi Girona 1–3, 08034 Barcelona, Spain*

[b] *Dept. of Computer Science, Karlstad University (KaU), Universitetsgatan 2, 651 88 Karlstad, Sweden*

[c] *Motorola Labs - Paris, Parc Les Algorithmes - Saint Aubin, 91193 Gif sur Yvette, France*

[d] *Networking Lab, Istituto Superiore Mario Boella (ISMB), via Pier Carlo Boggio 61, 10138 Torino, Italy*

Abstract

Recently, growing availability of emerging wireless technologies has pushed the demand to integrate different wireless-network technologies such as: wireless local-area networks, cellular networks, and personal and short-range networks. The interworking of heterogeneous radio access networks poses many technical challenges, with mobility management being one of the most important. In this paper we survey the existing proposals and show that transport-layer mobility is a viable candidate for implementing seamless handover in heterogeneous wireless access networks. Since the mobile Stream Control Transmission Protocol (mSCTP) is at the core of most relevant transport-layer mobility schemes being currently studied, we identify the key scenarios where the protocol can effectively leverage the multihoming feature to enhance handover support. Moreover, to provide the reader with a complete overview of the mSCTP's application area, we also survey the situations where the use of mSCTP-based schemes is not possible or has some limitations. Then, in one of the identified key scenarios, we investigate several challenging open issues related to path management and path-transition optimization by considering bandwidth-estimation schemes and link-layer support. Finally, we consider introducing concurrent multipath transfer (CMT) into mSCTP-based mobility schemes, as a future research direction.

Key words: transport-layer mobility, SCTP, handover management, performance evaluation, bandwidth estimation, concurrent multipath transfer

1 Introduction

The latest evolution and successful deployment of different wireless-network technologies (such as wireless local area networks (WLAN), cellular, personal and short-range) has spurred a strong demand to develop the framework for co-existence of heterogeneous wireless networks within, so called fourth-generation (4G) mobile data networks. According to [1], one of the most important technical challenges that the development of 4G networks poses is to provide seamless mobility that can guarantee service continuity for multi-mode mobile terminals like cellular phones, personal digital assistants (PDAs), and notebook computers. Seamless mobility requires the deployment of inter-system mobility management solutions, so that users and service providers are kept aside as much as possible from the complexity of inter-networking wireless access networks. In this sense, the development of mobility-management solutions over the Internet Protocol (IP) is a key aspect to achieve seamless mobility between heterogeneous wireless access networks.

Earlier work [2] on mobility management in heterogeneous networks discussed solutions affecting different layers of the IP stack. This paper, however, will mainly focus on transport-layer handover schemes, as this topic has still not been given enough attention in the research community. As the main contribution of the paper, we identify the key scenarios and challenging issues in handling seamless mobility at the transport layer in heterogeneous wireless access networks. Namely, we survey the scenarios where it is possible to apply transport-layer solutions like the mobile Stream Control Transmission Protocol (mSCTP) [3], having some benefits over other existing mobility solutions, and the scenarios where it is not recommended. Specifically, multihoming support is analyzed as the new protocol feature that lays the foundations of transport-layer mobility. In this context, the key topics are supported by the development of specific experiments on path management and path-transition optimization. In particular, after an initial analysis to assess the suitability of relying on the legacy SCTP failover mechanism to handle mobility, the use of link-layer information and end-to-end bandwidth estimation are considered in the protocol-optimization process.

* Corresponding author.

Email addresses: lukasz@tsc.upc.edu (Ł. Budzisz), ferrus@tsc.upc.edu (R. Ferrús), Anna.Brunstrom@kau.se (A. Brunstrom), karl-johan.grinnemo@kau.se (K.-J. Grinnemo), roberta.fracchia@motorola.com (R. Fracchia), giulio.galante.ext@nsn.com (G. Galante), ferranc@tsc.upc.edu (F. Casadevall).

[1] K.-J. Grinnemo is now a consultant at TietoEnator R&D Services AB, Kanikenasbanken 12, S-651 15 Karlstad, Sweden.

[2] G. Galante is now a consultant at Nokia Siemens Networks, Cinisello Balsamo (Milan), Italy

The rest of the paper is organized as follows: Section 2 introduces the details of mobility management and provides the reader with an overview of the existing mobility-management solutions, focusing on transport-layer schemes. Section 3 is devoted to the description of mSCTP, a transport-layer protocol that is at the core of the most relevant transport-layer mobility schemes, and to the identification of mSCTP's use for mobility management in heterogeneous wireless networks. Then, the main issues identified are illustrated by a set of experiments in Section 4. Finally, conclusions are drawn in Section 5.

2 Mobility management

2.1 Related work

In the near future, most Internet hosts will be mobile, so mobility should be supported throughout the Internet. In this context, an open challenge is the design of *mobility management* solutions that take full advantage of IP-based technologies to achieve the desired mobility between the various access technologies, and at the same time provide the necessary Quality of Service (QoS) guarantees.

Mobility can be classified into terminal, personal, session and service mobility. *Terminal mobility* is the ability of a mobile host (MH) to move between IP subnets, while continuing to be reachable for incoming requests and maintaining sessions across subnet changes. *Personal mobility* refers to the ability of addressing a user that can be located at several terminals. *Session mobility* refers to maintaining a session when moving between terminals. Finally, *service mobility* can be defined as the ability of users to maintain access to their services even when moving and changing terminals or service providers. Hereafter we concentrate on terminal mobility, since it is the foundation of the analysis addressed in this paper.

Management of terminal mobility includes two fundamental operations: location and handover management. According to [3], *handover management* deals with all the necessary operations to change a MH's point of attachment (PoA) to the IP network, while maintaining the communication with the correspondent node (CN). An IP-address change gives rise to challenges for maintaining an uninterrupted data flow, minimizing packet loss, and maintaining security. On the other hand, *location management* focuses on keeping track of a MH's current IP address, and providing this address to any entity needing to communicate with the MH, while being transparent to its peers.

Many proposals aimed at solving the problem of terminal mobility management in heterogeneous wireless networks providing IP connectivity can be found in the literature. A good survey on the current state of the art for mobility management in next-generation all-IP-based wireless systems can be found in [2]. Actually, the most representative mobility-management solution are Mobile IP (MIP) [4] and Mobile IPv6 (MIPv6) [5]. Both MIP schemes are usually classified as macro-mobility schemes, and are tailored to follow a MH's movement across different subnets within an Administrative Domain (AD), or across different subnets belonging to different ADs. Yet, if the MH's PoA is changed frequently, the MIP tunneling mechanism may lead to unacceptable network overhead in terms of increased delay, packet loss, and, especially, signaling. In this context, so-called micro-mobility schemes, such as Hierarchical Mobile IP (HMIP) [6], Cellular IP [7], Hawaii [8], and Fast Handoff [9], have been proposed to handle the movement of a MN within or across different PoAs in a subnet within an AD. A discussion of different micro-mobility protocols can be found in [10].

Another interesting approach to mobility management is introduced by Yabusaki et al. in [1]. The proposed solution advocates for the network itself to transparently handle mobility for mobile terminals. Thus, Yabusaki et al. suggest a network-centric solution to handle IP mobility in analogy with conventional 2G/3G networks, where mobility management has mainly been implemented as *network intelligence*, a concept just opposite to the *end-to-end intelligence* architectural principle of the Internet [11]. In this approach, IP addresses are used separately as host addresses and routing addresses. Thus, a *host address* is semi-permanently assigned to a MH and a *routing address* is temporarily assigned to the MH when datagrams are delivered to it. Datagrams are sent from a CN to a MH with the host address of the MH but then, within the IP mobile network, datagrams are transported using the routing address generated from the host address. All in all, user terminals are unaware of this rerouting management that is handled entirely in the network.

Handling mobility at the application layer has also received a lot of attention since a solution that is almost independent of the underlying wireless or wired access technologies and network-layer elements can be envisaged. In this context, the Session Initiation Protocol (SIP) [12] can be used for mobility management. Thus, when a MH moves during an active session into a different network, it first receives a new network address, and then sends a new *session invitation* to the CN. Subsequent data packets are forwarded to the MH using this new address. However, SIP by itself does not guarantee the maintenance of established Transmission Control Protocol (TCP) sessions or User Datagram Protocol (UDP) port bindings when moving, so further extensions such as S-SIP [13] are needed to provide seamless handover capabilities.

Besides network- and application-layer solutions, an important approach that is gaining attention in the last years is the support of mobility management at the transport layer. A complete survey and classification of mobility management schemes at the transport layer can be found in [14]. Transport-layer-based schemes enjoy several advantages such as inherent route optimization (triangular routes never occur), no dependence on the concept of home network or additional infrastructure beyond Dynamic Host Configuration Protocol (DHCP) [15] and Domain Name System (DNS) [16], the possibility of smooth handovers if the mobile node has multiple interfaces, and the ability to pause transmission during mobility-induced temporary disconnections [17]. Moreover, unlike network-layer schemes such as MIP, which make mobility transparent to upper layers by increasing the burden and responsibility of the Internet infrastructure, transport-layer schemes are based on an end-to-end approach to mobility that attempts to keep the Internet infrastructure unchanged by allowing the end-hosts to take care of mobility. It is also essential to point out the main inconvenience caused by the dominant role of well-established transport-layer protocols, like TCP. Therefore, most of the proposed transport-layer schemes require significant modifications of pre-existing protocol stacks.

Eddy, in [17], provides a comprehensive discussion on pros and cons of handling mobility management at different stack layers, concluding that transport-layer mobility schemes best fit the requirements of today's IP-based services, and that there should be more inter-layer communication to avoid conflicts and inefficiencies.

2.2 Transport-layer mobility

According to [3], *transport-layer mobility* is handled by the transport layers of the connection endpoints so that it is transparent to application-layer protocols not using IP addresses in their messages. A *mobility-enabled transport protocol* supports an IP-address change on the underlying network layer, while keeping the end-to-end connection alive. To achieve that, the MH first obtains a new IP address, then tells the CN — using the established transport-layer connection — that it is now reachable by the new IP address. Technically speaking, transport protocol adds the newly assigned IP address to the existing connection identifying the new connection to the server. To enable easy distinction of the different links at the MH, different IP addresses must be assigned to the server network interfaces. This allows representing different paths with different entries in the routing table of the MH.

So far, several proposals to handle mobility at the transport layer have been developed. In [14], a brief description of the most relevant solutions is provided jointly with a possible classification based on the authors' approach to mobility. In particular, the classification distinguishes between handoff protocols, connection-migration protocols, gateway-based mobility schemes and complete-mobility management schemes. *Connection-migration protocols*, such as Freeze-TCP [18], are specifically designed to reduce packet loss in the presence of long and frequent disconnections throughout the handover process. *Gateway-based mobility schemes*, as, e.g., the Mobile Socket Service (MSOCKS) scheme [19], introduce a gateway in the network for handling mobility. These solutions, however, bring in a single point of failure and may decrease fault tolerance. *Complete-mobility schemes*, also called *mobility managers*, like Migrate [20] and Seamless IP diversity-based Generalized Mobility Architecture (SIGMA) [21], include both handover and location management. Both solutions use DNS for location-management purposes, supporting either hard (Migrate) or soft handover (SIGMA). Such protocols only require modifications at the transport layer, thus leaving the existing network infrastructure unchanged. In contrast, *handoff protocols* only address handover-management issues, completely ignoring location management. Handoff protocols such as mSCTP or Mobile Multimedia Streaming Protocol (MMSP) [22] usually offer a soft-handover solution, and aim at reducing handover-induced packet loss, providing scalability and fault tolerance.

In this paper we will analyze the details of transport-layer-mobility solutions provided by handoff protocols. In particular we will focus on mSCTP, as an example of a mobility-enabled transport protocol, and also because of its new, interesting feature, multihoming.

3 SCTP for handover management support (mSCTP)

3.1 SCTP overview

The SCTP protocol, further referred to as *standard SCTP*, is defined in RFC 4960 [23]. Standard SCTP introduces a new feature called *multihoming*. Multihoming allows the use of multiple source-destination IP addresses for a single association between two SCTP endpoints. These IP addresses are exchanged and verified during the association initiation, and are considered as different paths towards the corresponding peer. Multiple paths are distinguished at each endpoint by their destination addresses. Among all available paths one is selected as the *primary path*, whereas all the rest are considered as *backup* or *alternate* paths. Multihoming, in case of IP networks, means multiple IP addresses, and typically multiple link-layer interfaces, as will be shown in the examples in the following sections.

Multihoming was designed for environments requiring high application availability, such as the delivery of Signaling System No. 7 (SS7) messages. Hence its scope of use, defined within RFC 4960 [23], is only for handling single retransmissions and performing primary path failover in case of permanent link failure. Other applications of multihoming, such as load balancing over multiple network paths, are not supported by the standard SCTP. Indeed, simultaneous data transfers over multiple paths may cause packet reordering leading to congestion control problems, since SCTP adheres strictly to the TCP congestion-control algorithm which is not designed to support multihoming.

Despite this limitation, SCTP multihoming seems an interesting protocol feature that may easily be leveraged to provide transport-layer handover to end-user applications. However, when considering standard SCTP multihoming support for transport-layer handover, it is very important to keep in mind that only one single path is used for data transmission (i.e., the primary path) while all other available paths can handle retransmissions only. Then, the decision of changing the primary path relies mainly on the failover mechanism. Another important consideration about multihoming support is that in standard SCTP no mechanisms are defined to dynamically change the set of IP addresses specified for an active association. Thus, in a mobile network scenario, if an association has already been established for a given IP address and a new PoA with a different IP address becomes available, there is no way to include it in the association and switch the primary path over to the new network connection.

3.2 The DAR extension: mSCTP

The *Dynamic Address Reconfiguration* (DAR) SCTP extension [24], although originally defined to help with IPv6 renumbering and hot-pluggable cards by the Internet Engineering Task Force (IETF) Signaling Transport (SIGTRAN) working group [25], can be easily leveraged to make SCTP a mobility-enabled transport protocol. It should be emphasized that this extension is seen as a mobility enabling feature, but not as a mobility solution by itself [26]. The DAR extension allows SCTP to dynamically add or delete IP addresses, and request the primary-path change during an active SCTP association, by means of two new chunk types[3]: *Address Configuration Change* (ASCONF, chunk type: 0xC1) and *Address Configuration Acknowledgment* (ASCONF-ACK, chunk type: 0x80); and six new parameters: *Add IP Address, Delete IP Address, Set Primary Address, Error Cause Indication, Success Indication, Adaptation Layer Indication.* Modifying the IP address(es) of the association increases the risk of association hijacking and therefore the ASCONF chunk must be sent in an authenticated way (an authentication chunk is bundled *before* the AS-

[3] A chunk is a unit of information within an SCTP packet.

CONF chunk), as described in [27]. Standard SCTP enhanced with the DAR extension is also referred to as *mobile SCTP* (*mSCTP*) [3,26].

With mSCTP, the primary path may be announced to the receiver's endpoint during association initialization and changed whenever it is needed during the association lifetime. When adding (deleting) an IP address to (from) an association, the new address *is not* considered fully valid (the existing address *must* be considered valid) until the ASCONF-ACK message is received. Changing the primary address may be combined with the addition or deletion of an IP address. However, only addresses already belonging to the association can be set to be the primary, otherwise the Set Primary Address request is discarded. mSCTP preserves the same congestion control rules as standard SCTP, and logically, a lot of research performed recently on SCTP could be useful for mSCTP development.

3.3 mSCTP applicability scenarios

As argued in Section 2.2, there are important advantages in handling handover at the transport layer. In particular, unlike a pure network-layer scheme, transport-layer handover has the ability to pause (hold) transmission during mobility-induced temporary disconnections, as well as the possibility of performing smooth handovers if the mobile node has multiple interfaces. Nevertheless, despite the importance of the aforementioned advantages, it is vital and highly relevant to identify under which situations/conditions such potential can really be exploited. In this sense, this section discusses the applicability of mSCTP in a set of practical scenarios in the context of heterogeneous radio access networks. In particular, the identification of mSCTP's applicability scenarios is based on the consideration of aspects such as the number of network interfaces on the MH, the number of IP addresses configured for the CN, and the IP address change during the handover process. For each scenario, expected benefits and open issues of the application of mSCTP are also stressed. It is worth to remark here that, even though the discussion focuses on mSCTP, most conclusions drawn here can be extended to any transport-layer handover solution.

However, before presenting the scenarios in more detail, an important comment on the naming convention must be made. The descriptions of the scenarios strictly follow the IETF naming convention defined in [28]. In particular, we focus on *access routers* (ARs), IP routers that reside on the edge of an Access Network offering IP connectivity to MHs, and acting as default routers for the MHs they are currently serving. Usually, each AR is connected to one or more Access Points (APs), sometimes called base stations or access point transceivers (in case of different technologies), which are layer-two devices of-

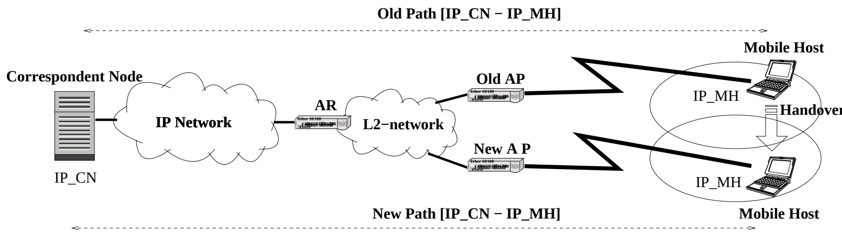


Fig. 1. Scenario A – The IP address is not changed in the handover process.

fering wireless link connections to MHs. Notice that, according to this naming convention, when addressing the use of mSCTP in 2G/3G cellular systems such as General Packet Radio Service (GPRS)/Universal Mobile Telecommunications System (UMTS), a base station would be referred to as AP and the role of AR would correspond basically to that of the 2G/3G network gateway (i.e., Gateway GPRS Support Node, GGSN in GPRS/UMTS) in charge of interconnecting the overall cellular network to an external IP packet data network and ultimately providing IP connectivity. In this way, the whole cellular network behaves as a layer-two network (L2-network). On the other hand, the naming convention used on WLANs is already aligned to the IETF naming convention with respect to the AP, but it is interesting to remark here that it is common to find WLAN devices with co-located AP and AR functionalities, referred to as *wireless routers*.

Scenario A. Although we have already pointed out that transport-layer solutions are targeted at scenarios where there is an IP-address change when moving from one PoA to another, we have retained this scenario to emphasize that nowadays the most common situation is that terminal mobility only results in an AP change, while staying in the same IP subnetwork, e.g., intra-system mobility in 2G/3G cellular networks (e.g., Global System for Mobile communications (GSM) or UMTS) or WLAN mobility within the same Extended Service Set (ESS), as presented in Fig. 1. Thus, mSCTP multihoming has no applicability here because the handover does not result in a change of the IP address used in the association. Efficient handover management can be achieved by means of link-layer solutions, so that no specific functionalities are strictly required within the transport layer to cope with the AP change. Nevertheless, despite this possible isolation of the transport layer from the cell-change process under such scenarios, cross-layer design constitutes an appealing research challenge to improve transport-layer performance by means of the information available from lower layers.

Scenario B. Future heterogeneous wireless networks, however, will bring a lot of diversity to the network structure, and terminal mobility among different radio access networks will most likely result in an IP-address change. Under such an assumption, the key feature of this scenario is that terminals can only use a single radio network interface at a time (see Fig. 2). We refer to this limitation as single-homed MH.

A typical situation in such a scenario is the handover of a WLAN terminal

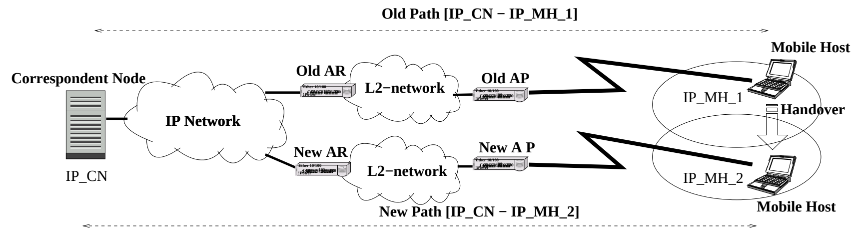


Fig. 2. Scenario B – Single-homed MH.

equipped with a single network card between two WLAN APs located in different IP subnetworks. Another situation is a dual-mode terminal (e.g. WLAN/GPRS) that cannot operate both interfaces simultaneously. This single-homed condition implies that the MH is not able to communicate with the new access point when using the old one and vice versa. This limitation can be tackled in two ways.

In the first approach, the MH disconnects from the old AP, obtains the new IP address from the new AP and sends the ASCONF chunk from that new IP address, even though the new IP address has not yet been included into the association. The packet containing the ASCONF chunk, to avoid being discarded by the receiver (the source address in the IP header is the new IP address), must contain the correct association verification tag in the SCTP-header Verification-Tag field, and the old IP address (the only valid address being part of the association) in the address parameter of the ASCONF chunk. Also the remaining parameters must be put in an appropriate order: Add new-IP address, Delete old-IP address, and Set primary address to new-IP address. Such a configuration will allow the receiver to recognize the association the chunk belongs to. During the lookup process, the receiver will first check the source address in the IP header. Upon failure of this attempt (the new address is still not a part of the association), the receiver will eventually get the appropriate IP address from the ASCONF-chunk address-parameter field. The lookup process is followed by the validation of the verification tag, and chunk processing. The receiver must then reply to the source IP address of the packet that, after processing all of the ASCONF chunk parameters, is the only IP address included in the association. The described mechanism has strong implications on security, and, as mentioned in Section 3.2, an authentication procedure is required before processing the ASCONF chunk to avoid the risk of association hijacking, as described in [24] and [29].

In the second approach, the radio access network has extensions facilitating the acquisition of a new IP address through the old AP. An example of such extensions is the Candidate Access Router Discovery (CARD) protocol that provides communication with the new AP through the old AP in order to obtain the new IP address, as shown in [30].

Assuming that the first approach is used in this scenario, as in [31], the main challenge is to keep system performance during the path-transition phase (i.e., from when the MH has sent a message with the Add IP Address and Set Primary Address option, until the transmission has started on the new address). In particular, the goal is to reduce the layer-two disconnection period, and consequently avoid timeout retransmissions being sent on the already inactive path.

Scenario C. In this scenario, we consider that MHs can actually be multi-homed, that is, more than one network interface can be operated simultaneously. However, regarding CN connectivity, a single-homed CN is assumed so that the CN is only reachable through a unique IP address. Fig. 3 illustrates this *asymmetric scenario* involving a dual-homed MH connected to a single-homed CN.

As most Internet servers nowadays are configured with only one IP address, this scenario is likely to become the most common in todays' heterogeneous landscape. Under such conditions, mSCTP can be applied to provide seamless handover between two APs connected to different subnets. The main phases in the handover process are explained below:

(1) Once a candidate AP has been selected, an IP address valid for the new location must be obtained. New addresses can be obtained either via DHCP or IPv6 auto-configuration in the new location.
(2) The new IP address is signaled to the mSCTP stack.
(3) mSCTP on the MH must notify the CN of the new address — the ADDIP address request is sent.
(4) The CN confirms the incorporation of the new address into the association. At that time, data from the CN towards the MH is still sent on the old path, as the primary-path change has not been requested yet. On the other hand, data coming from the MH can be sent to the CN through any network interface depending on its routing table configuration. Notice that the CN is single-homed and does not distinguish from which MH address it receives the data.
(5) Then, at an adequate moment (possibly the most important open issue is *how to determine when to switch the primary path*) the primary-path change is triggered by the MH, as it is commonly assumed that the handover decision is mainly related to the status of the radio link between the MH and the APs. The primary path change procedure is then started by sending an ASCONF chunk with the Set Primary Address parameter pointing to the new path.
(6) Once the change is confirmed by the CN, data should be sent only on the new path towards the MH.

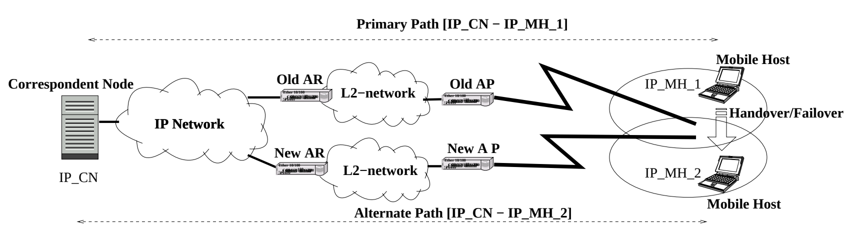


Fig. 3. Scenario C – Single-homed CN, Dual-homed MH.

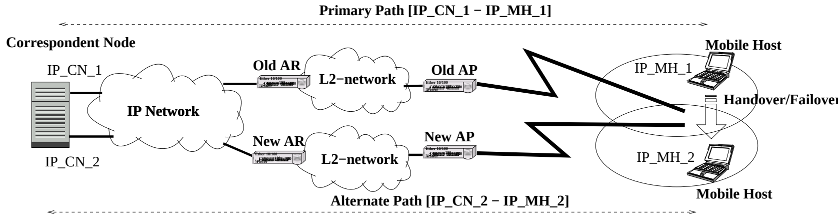


Fig. 4. Scenario D – Dual-homed CN, Dual-homed MH.

(7) Also, as soon as the connection with the old AP is lost, the unnecessary IP address should be removed by means of a Delete IP Address request.

Thus, the main challenges in this scenario are related to path management (i.e., criteria to trigger the primary path change) and path transition optimization (e.g., reducing the slow-start phase on the new path). Moreover, Concurrent Multipath Transfer (CMT) [32] can also be exploited in this scenario in the downlink direction. In this sense, the CN can wisely transmit new data towards more than one of the MH's IP addresses included in the association. In the uplink direction, however, the mSCTP association can only have one path leading to the unique CN IP address, and consequently CMT can not be employed within the mSCTP scheme.

Scenario D. In order to employ CMT in both the uplink and the downlink direction, we have to consider a multi-homed CN in a *symmetric scenario* as presented in Fig. 4. It must be stressed that such handover scheme will be similar to scenario C (asymmetric scenario), offering better flexibility as a consequence of having multiple paths between the endpoints. Indeed, the exploitation of multiple interfaces may be useful not only in case of failover due to mobility, but also in case of degraded performance of the active interface (due, e.g., to congestion).

In the presented use case we will encounter the same open points as in the single-homed fixed server case (triggering criteria), i.e., there will be room for performance improvements, such as transport-layer performance optimizations (path selection, slow-start-phase reduction) that will be discussed in Section 4.

Table 1 summarizes the discussion presented in this section.

4 Evaluation of handover strategies

This section provides the reader with some simulation results showing the performance of mSCTP. Our analysis focuses on the scenario identified in case D in Section 3.3, as it is the most comprehensive, and comprises the most relevant open points. First, we evaluate the basic failover mechanism that serves as a benchmark. Then, we analyze the existing protocol extensions that provide solutions to optimize handover, such as link-layer support and bandwidth

Table 1
Summary of mSCTP application scenarios.

	Scenario A	**Scenario B**	**Scenario C**	**Scenario D**
IP address change during handover	No	Yes	Yes	Yes
Number of interfaces on the MH	Multihoming not available at the transport layer	One (Single-homed)	Many (Multi-homed)	Many (Multi-homed)
Number of IP addresses on the CN	Not important	Not important	Single-homed	Multi-homed
Typical scenarios	Intra-system handover in 2G/3G Cellular or WLAN networks	Inter-system handover in heterogeneous networks	Inter-system handover in heterogeneous networks	Inter-system handover in heterogeneous networks
Open issues	Cross-layer design to enhance transport-layer performance	— Obtaining the new IP address — Transport-layer performance optimization	— CMT on downlink only — Handover triggering criteria — Transport-layer performance optimization	— CMT on up-/downlink — Handover triggering criteria — Transport-layer performance optimization

estimation techniques. Both subsections are illustrated with general results, providing the reader with the main concept of the introduced solution. Finally, we look at the optimization of the path transition process, illustrating with results the idea of the slow-start phase reduction and highlighting the main benefits of CMT.

4.1 Basic SCTP failover mechanism

As it was mentioned above, to introduce a good reference point for all considerations presented in this section, we first analyze the standard SCTP failover mechanism. In addition, we use it as a benchmark for performance evaluation of the other presented handover triggering solutions.

4.1.1 Preliminaries

As briefly mentioned in Section 3.1, SCTP was originally designed to transport telephony signaling over IP. To be able to comply with the availability and reliability requirements of telephony signaling, SCTP supports multihoming. In a multi-homed association, each endpoint chooses a primary destination address or path for the transmission of all new data chunks during normal transmission. The alternate paths only function as backup for the primary path, and, as long as the primary path is available, they are only used for retransmissions. A persistent failure to reach the primary destination eventually induces a failover, at which time the source endpoint selects one of the alternate paths as temporary primary path. The temporary primary path is then used until the original primary path becomes available again.

To detect path failure, SCTP provides two kinds of probing mechanisms: one for the primary path, and another for the alternate paths. To monitor the primary path, SCTP keeps an error counter that counts the number of consecutive timeouts. If the error counter reaches a certain tunable threshold, `Path.Max.Retrans` (PMR), the primary path is considered unavailable. However, if a Selective Acknowledgment (SACK) chunk is received before the error counter reaches `PMR`, the error counter is reset to zero. Since no data chunks are normally sent on the alternate paths, SCTP uses a heartbeat mechanism to monitor the availability of these paths. Special HEARTBEAT chunks are periodically sent on the alternate paths at a rate governed by a tunable heartbeat timer. If a HEARTBEAT-ACK chunk is not received before the timer expires, an error counter is incremented. Again, if the error counter reaches `PMR`, the corresponding alternate path is considered unavailable.

Fig. 5 further details the SCTP failover mechanism. Particularly, it shows the flow of events that take place when the primary path of a dual-homed association becomes unavailable. At (a), a link failure occurs on the primary path, and at (b) a timeout occurs for the data chunks sent at (c). The timeout triggers a window worth of data chunks to be retransmitted on the alternate path. Furthermore, the error counter of the primary path is incremented by one and the retransmission timeout (RTO) value is backed off to

$$RTO_{\text{new}} \leftarrow \min\{\max\{2 \times RTO_{\text{old}}, RTO_{\text{min}}\}, RTO_{\text{max}}\}, \tag{1}$$

where RTO_{min} and RTO_{max} determine the lower and upper limits of RTO.

At (d), a new data packet is sent on the primary path and the SCTP retransmission timer (T3-rtx timer), is restarted with RTO_{new}. The T3-rtx timer expires a second time at (e), and the actions taken at the first timeout event are repeated. After $\texttt{PMR} + 1$ consecutive timeout events, the primary path is abandoned and the alternate path is promoted temporary primary (f).

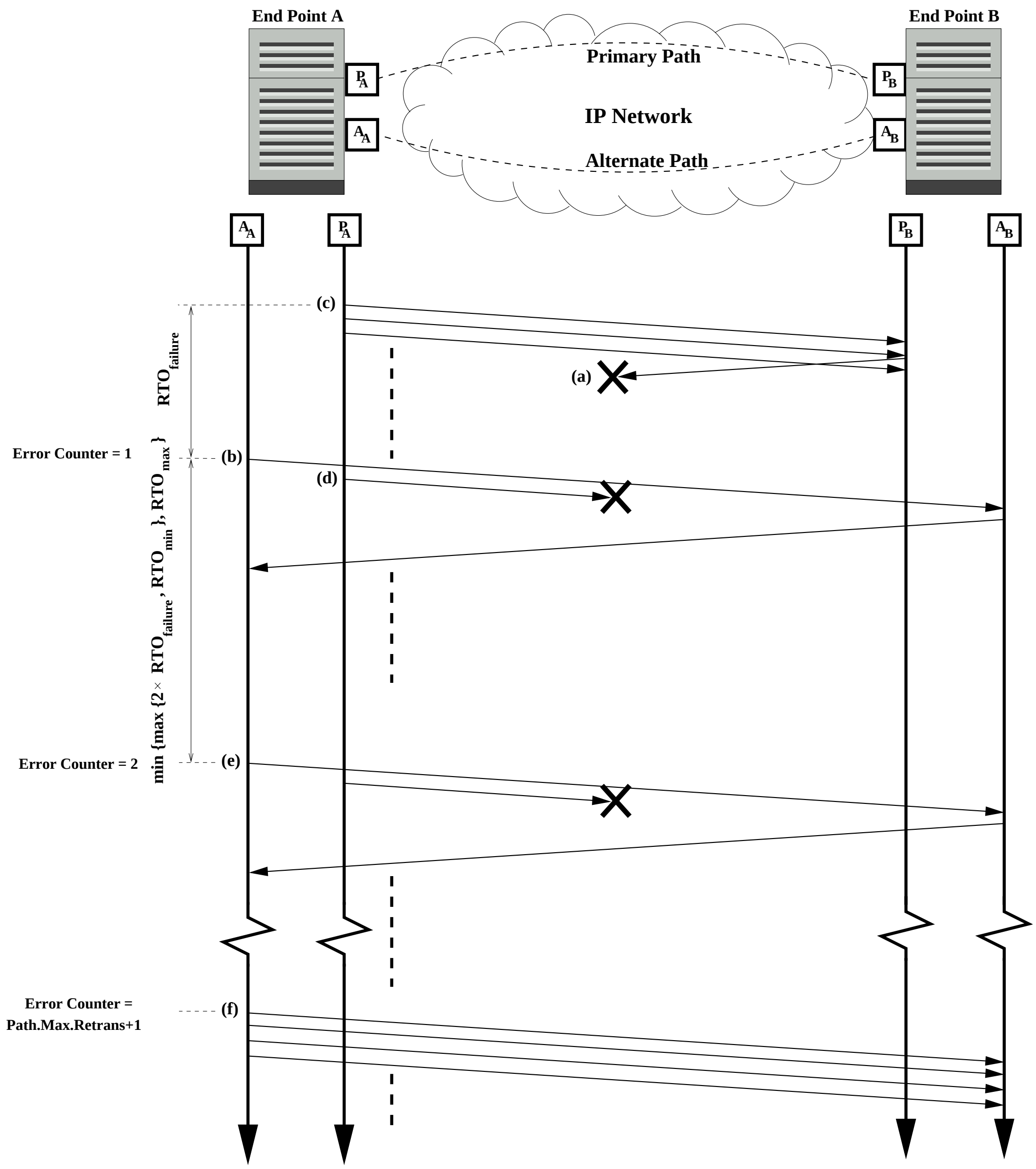


Fig. 5. The SCTP failover mechanism.

Since it takes $\mathtt{PMR}+1$ consecutive timeout events before the primary path is deemed unavailable, the minimum failover time is approximatively given by the equation:

$$T_{\text{fail}_{\min}} = \sum_{i=0}^{\mathtt{PMR}} \min\{2^i \times RTO_{\text{failure}}, RTO_{\max}\}, \tag{2}$$

where RTO_{failure} denotes the RTO at the time of the path failure [4]. Assuming

[4] A more accurate estimate of the SCTP failover time is found in [33].

that the RTO is always less than $RTO_{\max}$, equation (2) can be reduced to

$$T_{\mathrm{fail}_{\min}} = RTO_{\mathrm{failure}}(2^{\mathtt{PMR}+1} - 1). \tag{3}$$

4.1.2 Using SCTP failover for handover triggering

To study the feasibility of using the SCTP failover mechanism for handover between different types of wireless access networks, we conducted a simulation experiment in ns-2 [34]. The experiment considered the latency of handovers between three types of wireless networks, WLAN, UMTS, and GSM EDGE (Enhanced Data rates for GSM Evolution) Radio Access Network (GERAN), and is illustrated in Fig. 6. To make the handover latency independent of SCTP's RTO and SACK configuration, SCTP at both the CN and the MH was configured more aggressively than recommended in RFC 4960 [23]. Particularly, it used an $RTO_{\min}$ that was less than the minimum possible round-trip time, and no SACK delay. The parameter settings used to simulate the three types of wireless networks are listed in Table 2. The wireless network delay (wnd) models both the media access control (MAC)-contention delay and the link-propagation delay, but does not include the transmission delay. To account for the impact of the per-flow AP queue on the handover latency, each type of wireless network was simulated with two queue sizes (qs): one fairly small, 10 packets, and one larger than the bandwidth-delay product of any of the three considered network types, 50 packets.

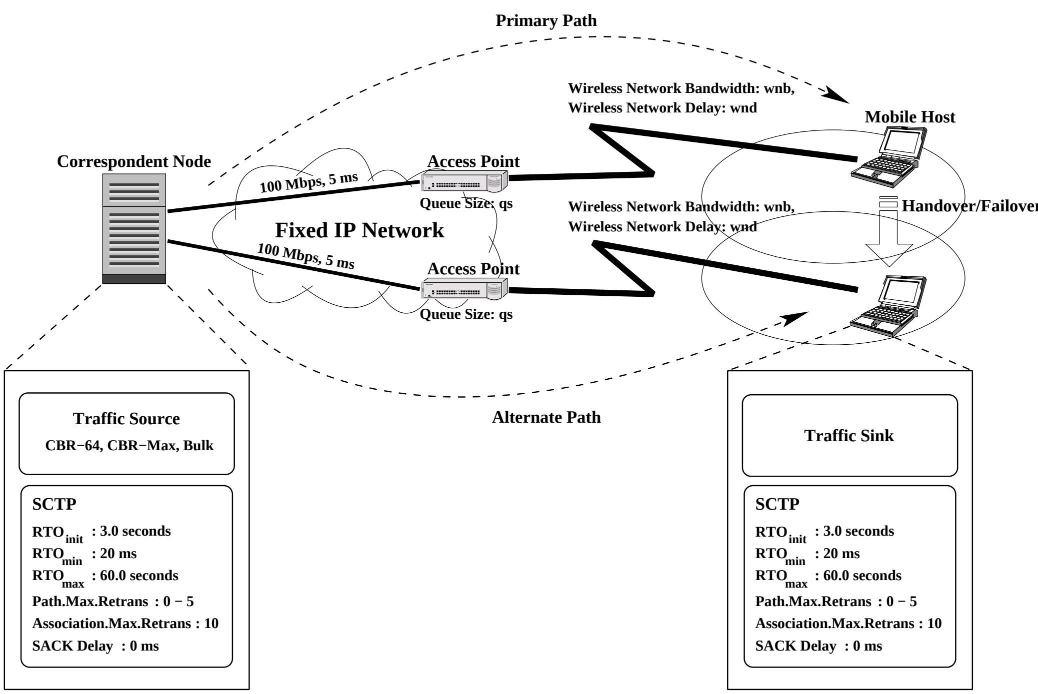


Fig. 6. The simulation experiment.

Table 2
Parameter settings for the three wireless network types studied in the simulation experiment.

Network	Parameter		
	wnb	**wnd**	**qs**
WLAN	11 Mbps	15 ms	10 packets, 50 packets
UMTS	384 kbps	80 ms	10 packets, 50 packets
GERAN	80 kbps	80 ms	10 packets, 50 packets

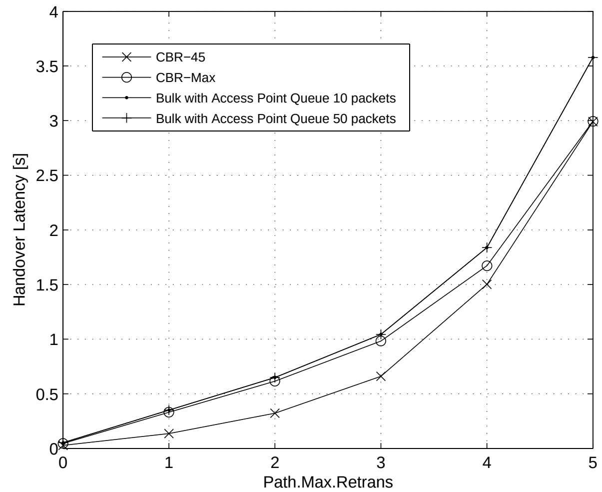
Since the SCTP failover time, and thus the handover latency, is strongly dependent on the traffic source (see Section 4.1.1), simulations were made with three different traffic sources: CBR-45, CBR-Max, and Bulk. CBR-45 and CBR-Max were both constant-bit-rate sources while Bulk was a greedy File Transfer Protocol (FTP) application. CBR-45 had a send rate of 45 kbps, and thus did only use a share of the available bandwidth in any of the three types of wireless networks. In contrast, CBR-Max sent with 70% of the wireless network bandwidth ($0.7 \times \texttt{wnb}$), and consequently used almost all available bandwidth. The properties of the three traffic sources are reported in Table 3.

To avoid systematic errors in the estimated handover latency, the start time of the traffic source was uniformly randomized within the first five seconds of a simulation. The movement of the mobile host to the new network was modeled as a direct break of the primary path. The break of the primary path took place when 10 seconds of a simulation run had elapsed and initiated a handover to the alternate path. Each simulation was repeated 30 times, and the mean and 95% confidence interval of the measured handover latencies were computed.

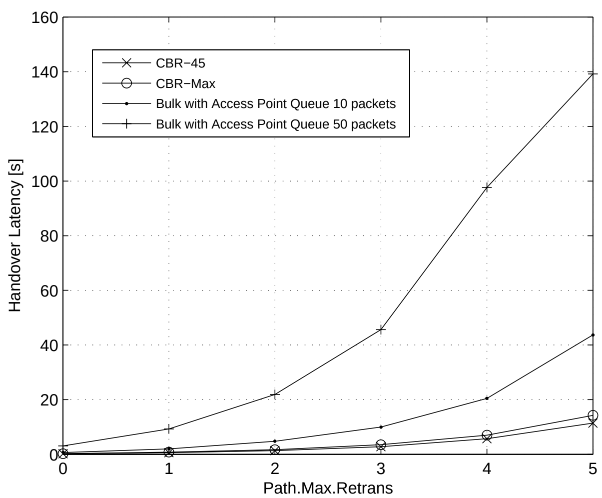
Fig. 7 shows how the average handover latency for the different traffic sources varied with `PMR` in the three handover scenarios: WLAN to UMTS, UMTS to WLAN, and GERAN to WLAN. Since the handover latency was primarily dependent on the primary path (see. Section 4.1.1), and only to a limited extent on the alternate path, the results from the other handover scenarios corresponded closely with the results from these three scenarios; for example, the average handover latencies in the WLAN-to-GERAN scenario were close

Table 3
Properties of the traffic sources.

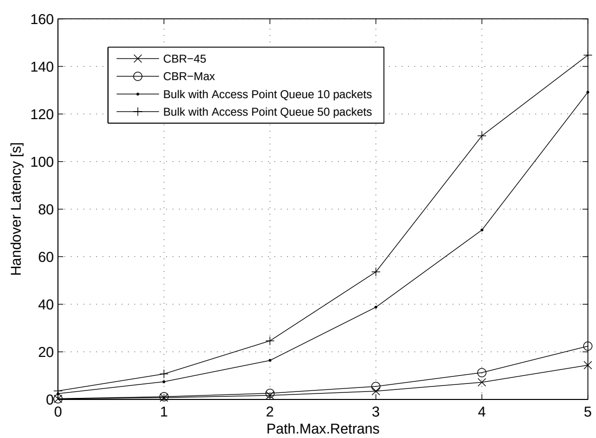
Traffic Source	Description	Packet Size	Send Rate
CBR-45	Constant bit rate	200 bytes	45 kbps
CBR-Max	Constant bit rate	1452 bytes	$0.7 \times \texttt{wnb}$
Bulk	FTP	1452 bytes	N/A



(a) Handover from WLAN to UMTS.



(b) Handover from UMTS to WLAN.



(c) Handover from GERAN to WLAN.

Fig. 7. Handover latencies between the three studied wireless network types as a function of `PMR`.

to the latencies in the WLAN-to-UMTS scenario. Also, since the size of the access point queue did not influence the handover latency of the constant bit rate traffic sources, only the results with an access point queue of 10 packets are shown in Fig. 7 for these traffic sources.

From Fig. 7 follows that the SCTP failover mechanism in most cases failed to provide handover latencies acceptable for real-time applications. Typically, such applications require latencies of less than 300 ms, however, SCTP only showed such short latencies when configured with a very small value of `PMR`. Particularly, only in the WLAN-to-UMTS scenario with the CBR-45 traffic source did SCTP provide real-time handover latencies for anything but `PMR` equal to zero, and then only for `PMR` equal to one. Considering that `PMR` should be set no lower than two to make SCTP reasonably resilient against spurious failovers, this essentially excludes the use of the SCTP failover mechanism for real-time applications. It also curtails its use for soft-real-time, interactive applications. This kind of applications often require handover latencies of no more than two seconds. However, as follows from Fig. 7, only in the WLAN-to-UMTS scenario did we have handover latencies of two seconds or less for all traffic sources when SCTP was configured with a `PMR` of at least two.

Although unsuitable for real-time applications, our experiment does not completely rule out the SCTP failover mechanism for non real-time applications. In fact, provided that handover latencies of several seconds are acceptable, our experiment suggests that the SCTP failover mechanism could indeed be used for these applications. Still, as will become evident in the following sections, there are better ways of using SCTP for handover than using the failover mechanism.

4.2 Path management optimization

The most crucial challenge for mSCTP is to provide optimal path management, aiming at improving the performance of the basic failover scheme presented in Section 4.1. The essential enhancements lie not only in providing a set of triggering rules that help to choose when to switch paths, but also in evaluating the possible gain in making such a decision at the transport layer. Consequently, in this section we consider both aspects: triggering-condition selection based on link-layer information, and feedback from bandwidth-estimation techniques, illustrated by the Autonomic Interface SeLEction (AISLE) SCTP extension [35].

4.2.1 Link-layer support

4.2.1.1 Preliminaries In homogeneous networks the handover decision made by mobile nodes is based exclusively on the information obtained from the radio-link layer such as the received-signal strength from the candidate APs. In contrast to this approach, in heterogeneous scenarios, information regarding different link features, such as available bandwidth, security, monetary cost, as well as end-user preferences may also be used in the decision process. This makes the entire handover process more complex and ambiguous as various aspects should simultaneously be taken into account to make a successful handover decision. To this end, there is some ongoing work in the IEEE 802.21 working group [36] on Media Independent Handover (MIH) specifying frameworks for inter-operation between various access technologies (vertical handover). There is also an IETF working group on Detecting Network Attachments (DNA) [37] devoted to improving the detection of IP-layer configuration and connectivity status.

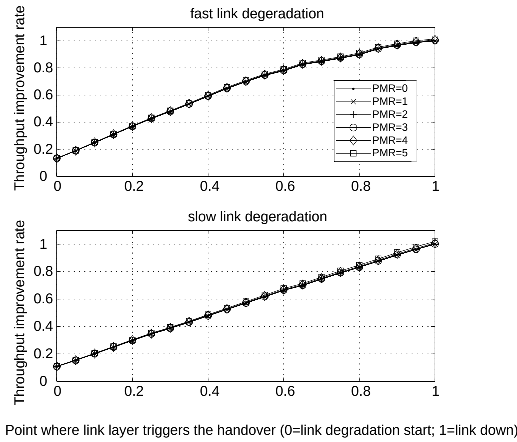
At first, it is necessary to determine what kind of support we can expect from the link layer and how this should be processed at the transport layer. According to [38], we can distinguish two categories of information that can be collected from the link layer: *events* and *parameters*. The former provide information about what happens at the link layer (AP detected, connected, disconnected), whereas the latter inform about the quality and features of a link (channel IP, signal strength, available bandwidth, price). Of course, the first problem that arises here is that different access technologies in heterogeneous wireless networks will most likely provide different parameters and events that are access-technology specific. In order to correlate that, parameters in different scales must be translated to the unified values for the upper layers. Such a common scale will comprise triggers and hints corresponding to the converted events and parameters, respectively. As an example, in [39], a part of the IETF DNA group has developed some patterns for triggers for GPRS, CDMA2000 (Code Division Multiple Access) and IEEE 802.11 link layers.

Having explained that, the next step is how to adjust the transport-layer protocol behavior to the user profile, or to the specific application. Thus, it is necessary to create decision metrics and design an appropriate handover policy. As shown in [40], the decision metrics that help to choose the appropriate network among those available should take into account: type of service, monetary cost, network and mobile-node conditions, system performance and last but not least, user preferences. This approach for 4G heterogeneous networks, results in multidimensional and highly complex cost functions. Meanwhile, the handover policy should not only include traditional techniques such as: *threshold value* (triggers the handover) and *hysteresis* (prevents the so called *ping-pong effect* — unnecessarily repeated switching), but also reflect criteria defined in the metrics.

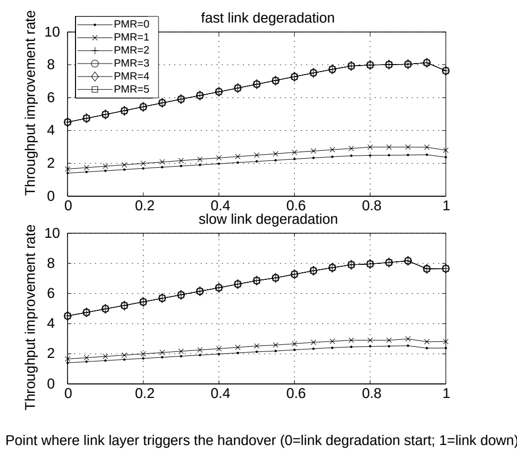
So far, in the literature devoted to mSCTP, only the traditional handover policies were analyzed. Examples of such research [41–43] include a traditional handover policy based on the relative signal strength criterion that creates a hysteresis between an aggressive and a conservative threshold value. The authors analyze the influence of different handover policies to determine when to add or delete the mobile nodes' IP addresses and how to change data delivery paths when handover happens.

4.2.1.2 An example handover policy Here we present an example extending the scope of the analyzed polices to mSCTP. We develop a scenario, where the user is oriented to stay in the fastest network as long as possible. If changing from a slower network to a faster network, the mobile user is interested in changing the point of attachment as soon as the faster network is detected. Therefore, right after the new IP address is obtained, the procedure for adding the new IP address to the existing connection is concatenated with the primary-path change that triggers the handover. Much more interesting in terms of the presented analysis is the case when the user switches from a faster to a slower network. In such case, the user is interested in staying as long as possible in the old access network. However, if this period is too long, it may result in firing the failover process and decreasing the overall throughput due to the failover latency.

We reutilized the simulation setup described in Section 4.1.2 analyzing the following two examples: change from WLAN to UMTS and change from UMTS to GERAN. Additionally, instead of the sudden link disconnection considered in Section 4.1.2, we introduced a link degradation pattern. This pattern captures the behavior of a link passing gradually from good conditions to link disconnection. The link degradation is modeled according to a linear increase of the frame error rate between 0 and 10% just before the link goes down. Two different durations are considered for this gradual degradation and they are referred to as *fast* and *slow* link degradation. During the entire link-degradation period the channel is stable enough to maintain the transmission on the degrading link. The link layer should notify the mobile user of when to switch to the slower network, taking into account the current handover policy. If the decision is made too late, and the degrading link is already down, the user will experience a failover that will cause a serious performance impairment. Switching to the slower link too early, will significantly decrease the overall performance. In this experiment, the main point was to show when the link layer should initiate the handover and compare it to the situation with no link-layer support, where the failover is the only event to trigger handover. We measured the improvement introduced by link-layer-triggered handover over failover-triggered handover for applications performing bulk data transfers. Fig. 8 presents the



(a) Handover from WLAN to UMTS



(b) Handover from UMTS to GERAN

Fig. 8. Normalized throughput improvement achieved by link-layer-triggered over failover-triggered handover as a function of normalized channel degradation time for a fast-to-slow-network handover for different `PMR` values.

ratio of the average throughput obtained by link-layer-triggered handover over the average throughput obtained by failover-triggered handover on a normalized time scale (from the moment the degradation started till the time the faster link became unavailable). The throughput ratio was calculated within the averaging window of 30 s.

As shown in the Fig. 8, in the case when the throughput difference between the available networks is huge, as in the case of WLAN-to-UMTS handover, the approach to stay as long as possible in the fastest network gives benefits even if no link-layer event is used to trigger a primary path change and finally it leads to a failover. Potential losses due to handover latency are fully compensated during the entire period of stay in the fastest network, independently of the link degradation pattern and the `PMR` setting. Also the adjustment made in

Section 4.1.2 plays an important role. In order to fully tailor mSCTP to the handover application, RTO.min was reduced from the default value of 1 s to 20 ms. That, in case of the fast network (WLAN), results in a relatively short failover time, and may be compensated, especially for bulk transfers.

The UMTS-to-GERAN handover should be performed before the UMTS link becomes unavailable because of the larger failover latency. If the link-change is performed as soon as the link degradation starts, the resulting overall throughput outperforms the failover degradation. As expected, the longer the user stays in the fastest network the better is the resulting throughput, as long as failover is avoided. In this scenario also the `PMR` parameter plays a significant role, resulting in a much higher gain for stable scenarios (`PMR` > 1).

4.2.2 Bandwidth estimation

Optimizing mSCTP path management, we will now look at the possible gain in making such a decision at the transport layer. An interesting SCTP extension that leverages multihoming and extends its use beyond the basic failover scheme is AISLE [35]. AISLE determines the wireless interface to be used for data transfer, by constantly monitoring the available bandwidth and the capacity between transport-layer endpoints, over both the primary and the secondary path. As a byproduct of such a strategy, the distribution of nodes across neighboring access networks is such that load is evenly balanced and per-node throughput is maximized. This approach works both under identical and different technologies, e.g., IEEE 802.11a/b or UMTS, although it is general enough to be extended to other technologies as well.

The estimation of the *available bandwidth* and of the *capacity* on each of the paths towards the destination addresses is essential for AISLE operation. This knowledge is used to enforce a dynamic redefinition of the identity of *primary* and *secondary* path. We use the term *capacity* of a link i to define the highest possible bit rate at which data can be transmitted on link i. This quantity will be identified as C_i. If link i carries a time-varying traffic load $R_i(t)$, then the *available bandwidth* on that link will be defined as $B_i(t) = C_i - R_i(t)$. Moreover, we refer to the capacity and available bandwidth of a path as the capacity and available bandwidth of its bottleneck link.

Capacity estimation and available-bandwidth estimation are traditionally carried out in different fashions. The AISLE approach to capacity estimation is based on the Sender-Based Packet-Pair (SBPP) technique described in Section 4.2.2.1. More specifically, AISLE replaces SCTP's HEARTBEAT chunk transmission on idle paths with the transmission of a 6-packet train on all active paths.

On the primary path, where there is a continuous stream of data packets and

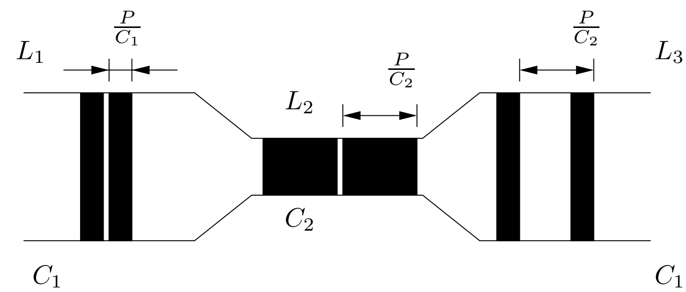


Fig. 9. Sender-Based Packet-Pair capacity estimation.

SACKs, AISLE uses the Westwood+ technique, described in Section 4.2.2.2 to estimate the available bandwidth.

As explained in Section 4.2.2.3, the AISLE path-selection algorithm requires capacity estimates on both the primary and the secondary path. Therefore, unlike SCTP, AISLE provides for probing heartbeats transmission on both the primary and the secondary path.

4.2.2.1 Sender-based packet-pair capacity estimation Fig. 9 illustrates the main idea behind the packet-pair capacity-estimation technique in a simple network scenario comprising links L_1, L_2, and L_3, having capacity C_1, C_2, and C_1 bits per second, separated by two nodes operating in store-and-forward mode. The pipe width graphically indicates the capacity of each link. In the example shown here, $C_2 < C_1$, so that L_2 is the *bottleneck link* of the path. Black rectangles represent P-bit packets sent on the links and their width is proportional to the packet transmission time P/C_i.

If two P-bit packets are sent back-to-back on L_1 and are inserted in the output buffer at the routers on either end of L_2 without any intervening packet sneaking in, they end up being sent on L_3 separated by $\delta = P/C_2$ — the packet transmission time on the bottleneck link L_2. It is therefore easy to obtain an estimate $\hat{C} = P/\delta$ of C_2.

Obviously, the best place for measuring δ is at the receiver, but it is also possible to have the receiver send acknowledgments back to the sender. This way, the transmitter can infer δ, under the hypothesis that the return path does not contain any bottleneck tighter than that on the forward path and no cross traffic alters the probe packets' spacing as explained in [44].

The effectiveness of the technique may be improved sending a 6-packet train (2 small — 2 large — 2 small), like the SProbe Tool [45]. Small packets are 40 bytes long, like standard SCTP HEARTBEATs; large packets are 1500 bytes long, a common maximum transfer unit (MTU) size. The bottleneck capacity may be estimated from the dispersion of the large packets that, taking more processing time at the nodes, have higher probability of being queued. The measurement is deemed valid only if there are no losses and packets are received in order. Moreover, the different packet sizes provide another heuristic test of the validity of the estimate: if no cross-traffic is spacing the probe packets and wireless errors do not cause too many retransmissions at the MAC layer, the inter-arrival time between large packets should be greater than that between small packets, otherwise the measurement is discarded.

4.2.2.2 Westwood+ available bandwidth estimation To estimate the available bandwidth on the primary path AISLE incorporates the idea proposed in TCP Westwood+ [46]. The available-bandwidth estimate $\hat{B}_i$ is derived on *bandwidth samples* $\hat{s}_i$, obtained on all data D_i sent on path i over contiguous, non overlapping time windows lasting either one round-trip time RTT_i, or 50 ms, whichever is larger. The bandwidth sample obtained on path i during the k-th time window can therefore be written as

$$\hat{s}_i^{(k)} = \frac{D_i^{(k)}}{\max\{RTT_i^{(k)}, 50\,\mathrm{ms}\}}$$

The actual available-bandwidth estimate on path i and sampling window k, $\hat{B}_i^{(k)}$, is then obtained as a smoothed exponential average of the available-bandwidth estimate on path i and sampling window $k-1$, $\hat{B}_i^{(k-1)}$, and the last bandwidth sample on window k, $\hat{s}_i^{(k)}$, according to:

$$\hat{B}_i^{(k)} = \frac{7}{8}\hat{B}_i^{(k-1)} + \frac{1}{8}\hat{s}_i^{(k)}$$

Further in the paper, we will drop the (k) superscript for notation simplicity.

4.2.2.3 Path selection The advantage of using multiple wireless interfaces would be wasted without an efficient management of the available paths, based on relaxing SCTP's rigid "primary-secondary" path definition. The idea at the core of AISLE path selection is quite simple: every time an AISLE sender has reason to believe that the primary path it is using has become congested, it tries to determine whether it would have better luck on the secondary path. A time hysteresis is introduced to avoid bouncing the data back and forth from one path to the other. Before detailing the procedure, we remark that such path management becomes especially appealing in the case of AISLE stations with dual attachment to different access networks (scenarios B, C and D in Section 3.3). Indeed, it is easy to see that, if too many wireless stations associated to the same access network, it would soon become congested. If another access network were within radio coverage of some of these stations, and if they used AISLE, they could decide to associate to the second access network, thereby relieving the congestion on the first one. In this way, an autonomic selection of the least-congested path would be achieved, without any user or system-operator intervention.

In describing the path-selection procedure the following assumptions are made. (i) Only two paths are available between two AISLE stations, and each goes through a different wireless network (e.g., WLAN or 3G network). (ii) For each path, an AISLE station estimates its capacity ($\hat{C}_1$ and $\hat{C}_2$, respectively) using the packet-pair technique; when either path becomes the primary, also an available bandwidth estimate ($\hat{B}_1$ and $\hat{B}_2$, respectively) is obtained.

AISLE infers congestion on the primary path when either of these situations occurs: (i) a packet loss is detected through retransmission-timeout expiration or triple duplicate-SACK reception (as in the TCP congestion control mechanism); (ii) the path's available bandwidth is smaller than 10% of its capacity. The latter condition lets stations that do not experience losses because their transmission rate is limited by a very-small congestion window (cwnd) trigger the path selection as well.

Then, assume that path 1 is the primary path. As soon as an AISLE station detects congestion on its primary path, it performs the following path-selection procedure.

Step 1: If no path swap has been performed in the past T_h seconds (hysteresis time), evaluate whether a swap should be performed with probability $p_{\mathrm{s}} = \hat{C}_2/(\hat{C}_1 + \hat{C}_2)$(goto 2); else, do nothing with probability $p_{\mathrm{ns}} = 1 - p_s$ (goto 3).

Step 2: If $\hat{B}_2$ is known and its value is fresh enough (i.e., it was last estimated not earlier than a bandwidth-decay time T_d ago), swap primary and secondary path if $\hat{B}_2 > \theta \hat{B}_1$, with $\theta > 1$. Else, (i.e., if $\hat{B}_2$ is unknown or stale): swap primary and secondary with probability p_{s}; do nothing with probability p_{ns}.

Step 3: End of path swap procedure.

Note that Step 1 avoids frequent swaps when $\hat{C}_2 < \hat{C}_1$. If swapping is considered (Step 2), the available-bandwidth estimate is used, which provides a more accurate indication than the path-capacity estimate.

4.2.2.4 Simulation results In this section we present the performance of AISLE running simulations with ns-2 on the mixed wired-cum-wireless topology of Fig. 4.

One CN is connected to the network through wired links running at 100 Mbps.

We consider FTP sources, generating long-lived flows at the CN, transferred through the Internet to the wireless stations. The packet size at the IP level is 1500 bytes.

We consider two different scenarios, detailed in the following. First, we assume that each station has two radio interfaces and can connect simultaneously to AP_1 and AP_2, which operate according to the 802.11a and the 802.11b standards, respectively. All stations using the path through AP_1 as primary transmit at 54 Mbps, while stations selecting the 802.11b interface transmit at 11 Mbps. In our reference scenario, WLANs employ the Distributed Coordination Function (DCF) at the MAC layer. The DCF parameters are set to the standard values; Request To Send / Clear To Send (RTS/CTS) are used for payloads in excess of 400 bytes and the Short Retry Limit and the Long Retry Limit are set to 7 and 4, respectively. The link-layer queues at the wireless stations are 50 data frames long, whereas the AP queues accommodate up to 400 data frames.

As far as the specific AISLE parameters are concerned, the threshold θ is set to 1.1, the hysteresis T_h to 60 s, and the bandwidth-decay time T_d to 120 s.

Independently of the adopted technology, we assume that the wireless channel is error-free (packets are lost due to buffer overflow at the AP or due to channel contention) and that the propagation delay is negligible on the wireless part of the network.

In particular, we investigate the capability of AISLE nodes to *create*, in a dynamic and autonomic manner, a network topology where users are optimally distributed among the available PoAs: we consider the case of users reaching a meeting zone and connecting one by one to the provided WLANs. The station inter-arrival time is a random variable exponentially distributed with mean 30 s. A maximum number of 50 wireless nodes is considered and every station, upon arrival, randomly selects one of the APs with equal probability.

Fig. 10 shows the time evolution of the number of stations using the 802.11a and the 802.11b APs. Even if, initially, every station randomly chooses the AP to associate with, nodes do not remain equally distributed between the APs. Indeed, an AISLE node spontaneously tends to use the interface that maximizes its throughput; thus, in the considered network scenario, a larger number of stations will associate to the 802.11a AP, whose capacity is larger. At any given time, the reference value represents the fair node distribution, derived as explained in [35]. These results clearly show that AISLE stations rapidly adapt to changes in the network topology, and optimally choose the radio interface.

Now, we consider the case of heterogeneous access networks, based on 802.11 and UMTS. The 3G cellular network, simulated with EURANE (Enhanced

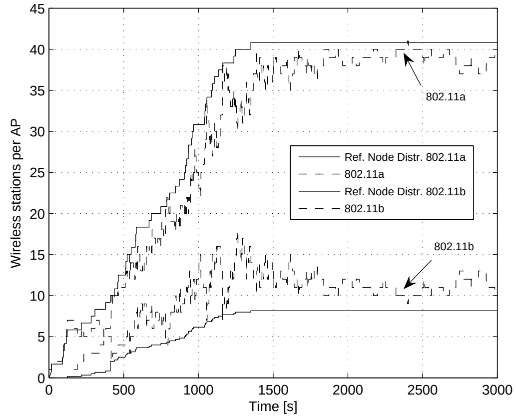


Fig. 10. Time evolution of the number of wireless stations using the 802.11a and the 802.11b APs.

UMTS Radio Access Network Extensions for ns-2) [47], includes both the UMTS Radio Access Network (UTRAN), that handles all the radio-related functionalities, and the Core Network, which is responsible for routing connections to external networks. Considering the Radio Link Control (RLC) protocol, we use the Acknowledged Mode, normally selected for web browsing and email downloading. At the physical layer, we use common transport channels: every channel is shared among all users within a cell [48]. We set the Transmission Time Interval (TTI) of the physical channels to 10 ms, and the RLC payload size to 40 bytes.

We evaluate how the wireless stations distribute among overlapping heterogeneous networks: PoA_1 is an 802.11b AP, while PoA_2 is a UMTS Node B. Stations using the 802.11b AP transmit at 11 Mbps, while stations using the UMTS Node B employ a channel at 384 kbps. All stations implement AISLE and they initially use the WLAN PoA.

Fig. 11 presents the number of stations using the 802.11 AP, as a function of

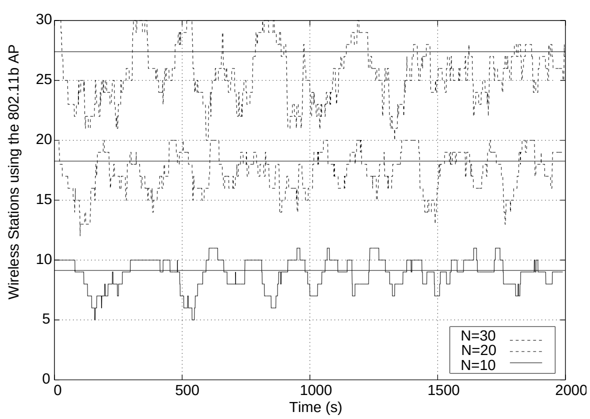


Fig. 11. Time evolution of the number of wireless stations using the 802.11b AP as a function of time, for different total numbers of users. Heterogeneous network scenario, with all stations being initially associated to the WLAN.

the simulation time, for N =10, 20 and 30 stations. Again, the user partition is very close to the optimal, fair value. Indeed, even if on the UMTS transport channels the transmission is scheduled in radio frames (TTI of 10 ms), affecting the accuracy of packet-pair capacity estimation, the bandwidth estimation technique accurately evaluates the available bandwidth, allowing a close to optimal node distribution.

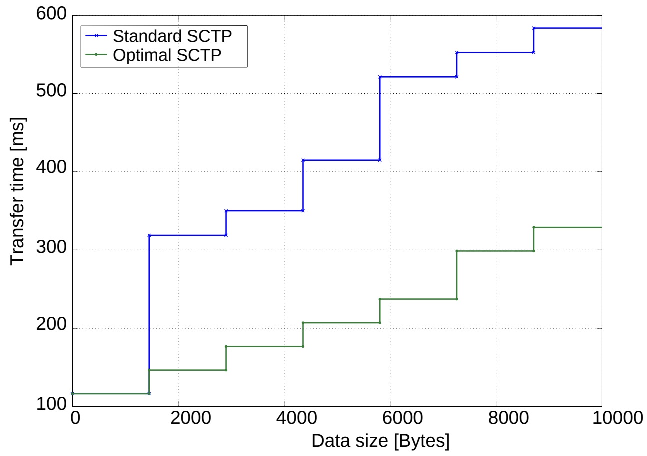
4.3 Path transition optimization

Obviously, when incorporating the mSCTP transport layer handover scheme, not only path management needs to be improved, but also the transition process must be optimized. In this section we demonstrate a couple of examples that smooth the transition process, including slow-start phase reduction and CMT.

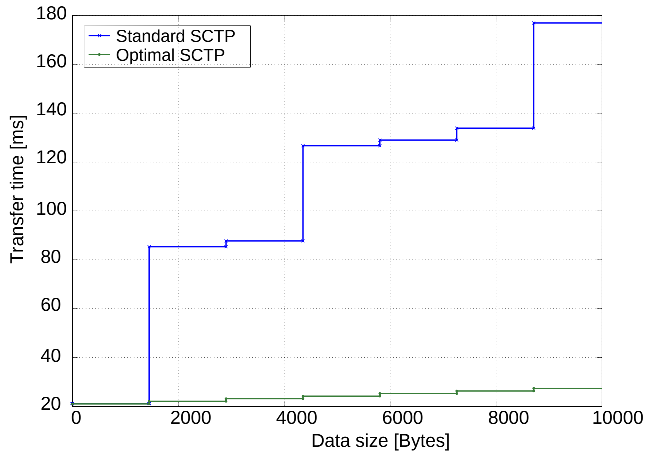
4.3.1 Using bandwidth estimation to reduce the slow-start phase

After handover, when SCTP has abandoned the original primary path and selected one of the alternate paths for the continuing traffic, it begins data transmission in slow start. Given that SCTP does not know the available bandwidth on the selected alternate path in advance, this startup behavior, of course, makes perfect sense. However, provided that an estimate of the link capacity can be obtained, e.g., through the use of the packet-pair technique discussed in Section 4.2.2.1, this behavior is indeed not optimal. To acquire some appreciation of the possible gains in using a bandwidth-aware startup scheme as compared to slow start, we reutilized the simulation setup in Section 4.1.2. We compared the transfer times of standard SCTP during its startup on the alternate path with those of a fictitious optimal SCTP. The optimal SCTP had perfect knowledge about the available bandwidth, and was able to send with the optimal rate directly from the start. In practice, issues such as imperfect bandwidth estimations and the effect of transmission bursts would of course have to be accounted for. However, these aspects were ignored here, since we focused on obtaining a rough estimate of the possible gains. We limited our study to the Bulk traffic source since the CBR sources did not adapt their send rates to the available bandwidth on the alternate path, and thus made the transfer times source dependent. Further, since the value of `PMR` had no effect on the startup behavior on the alternate path, simulations were only made with `PMR` set to three.

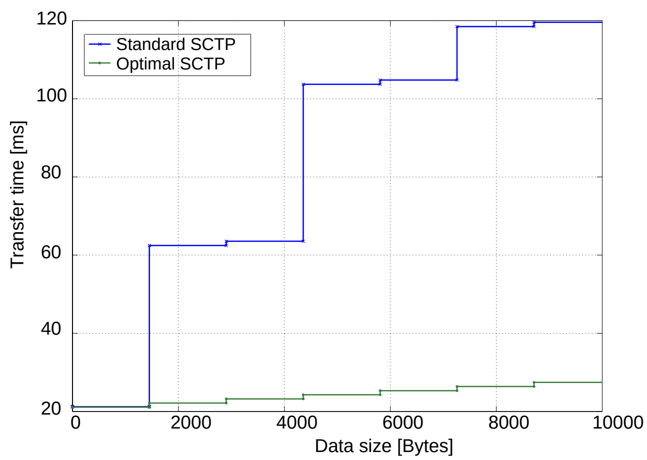
Fig. 12 shows the result of the simulation experiments. The transfer time denotes the time it takes to transfer various amounts of data during slow start. The horizontal lines seen in the figure each represent the data transfered in one



(a) Handover from WLAN to UMTS.



(b) Handover from UMTS to WLAN.



(c) Handover from GERAN to WLAN.

Fig. 12. Comparison of transfer times on the alternate path between standard and optimal SCTP.

SCTP packet. We observe that in all handover scenarios the improvements in transfer times with respect to the optimal SCTP was substantial; especially in the UMTS-to-WLAN and GERAN-to-WLAN scenarios, where the alternate path had a relatively large bandwidth-delay product, were the gains large. Particularly, the gain in transfer time was the same in these two scenarios, and was up to about 75% for data sizes of less than 6 kbytes. Since both these scenarios entailed handover to the same type of wireless network, a WLAN, the fact that the transfer times were the same in the two scenarios was not surprising. In the WLAN-to-UMTS scenario, the improvement in transfer time was not as large as in the other two scenarios. However, it was still significant, with up to 50% gain in transfer time for data sizes of less than 6 kbytes.

Although this experiment only provides a rough upper bound, with the optimal SCTP being an unattainable target, it still remains that even a conservative interpretation of the results suggests that large improvements in transfer times on the alternate path are possible with a bandwidth-aware startup scheme. Currently, we are studying ways of using the SCTP heartbeat mechanism together with the aforementioned packet-pair technique to implement a bandwidth-aware startup scheme.

4.3.2 Incorporating CMT schemes

Finally, we explore the benefits achievable by incorporating CMT into mSCTP mobility management. The most important scheme aiming at improving SCTP's multihoming performance is described in [32]. Although the schemes proposed in [49–51] were envisioned for load balancing traffic over persistent links, they may also be attractive for soft-handover scenarios, where the temporary availability of more than one link to the destination can be exploited to make the handover smoother. The key ingredients needed in this case are basically two. First, the SCTP send-buffer management and congestion control must be updated as described in Section 4.3.2.1, so as to account for the problems induced by concurrently sending data over multiple paths using only one sequence-number space. Second, a scheduling algorithm must be introduced to try and minimize packet reordering at the receiver as explained in Section 4.3.2.2.

4.3.2.1 Buffer management and congestion control To accommodate CMT, the single-buffer SCTP architecture must be replaced by a *multibuffer* structure, giving to each interface its own send buffer. The multibuffer structure guarantees path independence as far as transmission is concerned, but introduces the need for modifications to SACK handling at the source. First of all, out-of-order SACKs are not trusted as in the original SCTP and are discarded, because CMT may easily lead to SACK reordering.

Then, the standard SCTP per-association congestion control must be extended to a per-path congestion control. Indeed, the original SCTP only allows for cwnd adjustments when the association Cumulative Transport Sequence Number (TSN) ACK (Cum-ACK) Point is updated by an incoming SACK. This is correct when at most one path is used at any given time and consecutive packets arrive reasonably in-order at the receiver. However, when packets are concurrently transmitted to multiple destinations, the assumption of reasonably-ordered reception does not hold anymore, and SACKs, which do not update the association Cum-ACK Point, may acknowledge one or more chunks that were received in-order on different paths. In this case, the standard SCTP would behave incorrectly, not updating the cwnd of the paths the SACKs refer to, and missing all cwnd updates that do not move the association Cum-ACK Point. This problem can be easily solved by introducing a per-path Cum-ACK Point and updating it at every meaningful SACK reception.

Another point that requires attention is the Fast Retransmit algorithm: the original SCTP triggers it after three consecutive missing SACK reports, leading to congestion-window reduction and retransmissions on the connection over which the presumably-lost packets were last sent. However, when splitting a traffic flow over multiple connections, the path diversity induced by different bandwidth and delay on each path in the association, may lead to a high number of out-of-order packet arrivals at the receiver. This, in turn, causes the receiver to advertise large, enduring gaps in the received-chunk sequence, forcing several fast retransmissions and producing unneeded cwnd reductions. Also the solution to this problem is based on per-connection SACK processing. Chunk missing reports are trusted only if the SACK acknowledges for the first time at least one chunk that (i) was transmitted over the same path as the missing chunk, and (ii) has a TSN higher than that of the missing chunk. In other words, the number of missing reports for a chunk is incremented only if one or more chunks, sent on the same path as the missing one and after it, are acknowledged, thus implying the possibility of a real loss.

Notice that all changes necessary to support CMT only affect the SCTP sender, while the receiver architecture and behavior remain unmodified: the receiver buffer of a multipath-enabled SCTP association collects all chunks from each connection like in standard SCTP. SACKs are generated as in SCTP and transmitted over the link from which the last packet was received. Finally, no overhead is added to the original SCTP protocol, because the packet format is unchanged.

Finally, we want to stress the importance of correct receive-buffer dimensioning to prevent data-transfer starvation and guarantee good protocol performance. Here, for the sake of brevity, we completely skip the complex discussion of this issue and refer the interested reader to [52,53] for further details.

4.3.2.2 Packet-Scheduling Algorithm

As a second crucial component to accommodate CMT, we introduce a bandwidth-aware scheduler. The bandwidth-aware scheduler allocates each chunk to one of the available destinations exploiting a capacity estimate on each path, with the aim of maximizing the chance of ordered packet delivery at the receiver.

The scheduler works as follows. Whenever a new chunk having size P is available for transmission from the application layer, the scheduler computes its reception time R_i for each path i, as

$$R_i = \frac{O_i + P}{\hat{C}_i}$$

where O_i is the amount of outstanding data on path i. The resulting R_i accounts for two effects. $O_i/\hat{C}_i$ is the time needed to transmit at bit rate $\hat{C}_i$ an amount of data equal to the cwnd: a larger capacity $\hat{C}_i$ leads to a faster transmission speed. The term $P/\hat{C}_i$ is the time that it takes for the current chunk P to be transmitted over path i: a large chunk size P or a low estimated capacity $\hat{C}_i$ would lead to a long transmission time. The path with the lowest R_i is selected as the chunk destination, according to the Fastest Path First (FPF) principle stated in [49].

Note that reception times only relate to the last data window sent over the path rather than to those sent over the entire association life, and that path propagation delays are not taken into account in this version. The rationale behind this is twofold. First, it is better to evaluate the reception time R_i on the last data window rather than on the whole association duration because in the latter case capacity-estimation errors tend to accumulate and yield a reception-time estimate that gets worse and worse as time passes. Second, it is not possible to obtain an estimate of the paths' round-trip time before actually starting to send data over them.

5 Conclusions

In this paper we analyzed mobility management as the key issue in the coexistence of heterogeneous wireless networks. So far the literature devoted to mobility management in heterogeneous networks discussed mainly network- and application-layer solutions. In our opinion, transport-layer handover schemes are a worthwhile option and deserve receiving more attention from the research community despite their main drawback of requiring the modification of well-established transport-layer protocols. As an illustration, we identified key scenarios and challenging issues in handling seamless mobility at the transport layer in heterogeneous wireless access networks, using the mSCTP protocol as an example. Within this context, we showed the unsuitability of relying on the legacy SCTP failover mechanism to handle mobility, especially for real-time services. Consequently, in the protocol optimization process we envisaged the use of link-layer information and end-to-end bandwidth estimations. Yet, the main open issues remain in the adjustments of handover-triggering conditions when link-layer support is available and the application of bandwidth estimates to improve the path selection process. Last but not least, we sketched the future evolution track pointing towards CMT applications, identifying the most important issues that must be considered, such as buffer management, congestion control and appropriate scheduling algorithms.

Acknowledgments

This work has been supported by the NEWCOM Network of Excellence (http://newcom.ismb.it) partly funded by the European Community.

The work at UPC has also been supported by CYCIT (Spanish National Science Council) under grant TEC2006-09109, which is partially financed from the European Community through the FEDER program.

The work at KaU was also partially supported by grants from the Knowledge Foundation of Sweden with TietoEnator and Ericsson as industrial partners.

The work at ISMB was carried out in the framework of PRIMO, a project on reconfigurable platforms for wideband wireless communications, partly funded by the Italian Ministry of University and Research (MIUR).

References

[1] M. Yabusaki, T. Okagawa, K. Imai, Mobility management in All-IP mobile network: end-to-end intelligence or network intelligence?, IEEE Communications Magazine 43 (12) (2005) supl.16–supl.24.

[2] I. Akyildiz, J. Xie, S. Mohanty, A survey of mobility management in next generation All-IP based wireless systems, IEEE Wireless Communications 11 (4) (2004) 16–28.

[3] M. Riegel, M. Tuexen, Mobile SCTP, IETF draft, <draft-riegel-tuexen-mobile-sctp-09.txt>, work in progress (Nov. 2007).

[4] C. Perkins, RFC 2002, IP mobility support (Oct. 1996).
URL `http://www.ietf.org/rfc/rfc2002.txt`

[5] D. Johnson, C. Perkins, J. Arkko, RFC 3775, Mobility support in IPv6 (Jun. 2004).
URL http://www.ietf.org/rfc/rfc3775.txt

[6] E. Gustafsson, A. Jonsson, C. Perkins, Mobile IPv4 regional registration, IETF draft, <draft-ietf-mobileip-reg-tunnel-09.txt>, draft expired (Jun. 2004).

[7] A. Campbell, J. Gomez, C.-Y. Wan, S. Kim, Z. Turányi, A. Valkó, Cellular IP, IETF draft, <draft-valko-cellularip-01.txt>, draft expired (Oct. 1999).

[8] R. Ramjee, T. L. Porta, S. Thuel, K. Varadhan, S. Wang, HAWAII: a domain-based approach for supporting mobility in wide-area wireless networks, in: Proc. of the 7th International International Conference on Network Protocols, 1999 (ICNP 1999), 1999, pp. 283–292.

[9] K. E. Malki, H. Soliman, Fast handoffs in mobile IPv4, IETF draft, <draft-elmalki-mobileip-fast-handoffs-03.txt>, draft expired (Sep. 2000).

[10] A. Campbell, J. Gomez, S. Kim, C.-Y. Wan, Z. Turányi, A. Valkó, Comparison of IP micro-mobility protocols, IEEE Wireless Communications 9 (1) (2002) 72–82.

[11] B. Carpenter, RFC 1958, Architectural principles of the Internet (Jun. 1996).
URL http://www.ietf.org/rfc/rfc1958.txt

[12] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler, RFC 3261, SIP: Session Initiation Protocol (Jun. 2002).
URL http://www.ietf.org/rfc/rfc3261.txt

[13] J. Zhang, H. Chan, V. Leung, A SIP-based seamless-handoff (S-SIP) scheme for heterogeneous mobile networks, in: Proc. of the IEEE Wireless Communications and Networking Conference (WCNC 2007), Hong Kong, 2007, pp. 3946–3950.

[14] M. Atiquzzaman, A. Reaz, Survey and classification of transport layer mobility management schemes, in: Proc. of the 16th IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC 2005), Vol. 4, Berlin, Germany, 2005, pp. 2109–2115.

[15] R. Droms, RFC 2131, Dynamic Host Configuration Protocol (Mar. 1997).
URL http://www.ietf.org/rfc/rfc2131.txt

[16] P. Mockapetris, RFC 1034, Domain names - concepts and facilities (Nov. 1987).
URL http://www.ietf.org/rfc/rfc1034.txt

[17] M. Eddy, At what layer does mobility belong?, IEEE Communications Magazine 42 (10) (2004) 155–159.

[18] T. Goff, J. Moronski, D. Phatak, V. Gupta, Freeze-TCP: a true end-to-end TCP enhancement mechanism for mobile environments, in: Proc. of the 19th Annual Joint Conference of the IEEE Computer and Communications Society (INFOCOM 2000), Tel Aviv, Israel, 2000, pp. 1537–1545.

[19] D. Maltz, P. Bhagwat, MSOCKS: an architecture for transport layer mobility, in: Proc. of the 17th Annual Joint Conference of the IEEE Computer and Communications Society (INFOCOM 1998), Vol. 3, San Francisco, CA, USA, 1998, pp. 1037–1045.

[20] A. Snoeren, H. Balakrishnan, An end-to-end approach to host mobility, in: Proc. of the 6th International Conference on Mobile Computing and Networking (MobiCom 2000), Boston, MA, USA, 2000, pp. 155–166.

[21] S. Fu, L. Ma, M. Atiquzzaman, Y. Lee, Architecture and performance of SIGMA: A seamless handover scheme for data networks, in: Proc. of the IEEE International Conference on Communications (ICC 2005), Vol. 5, Seoul, South Korea, 2005, pp. 3249–3255.

[22] H. Matsuoka, T. Yoshimura, T. Ohya, End-to-end robust IP soft handover, in: Proc. of the IEEE International Conference on Communications (ICC 2003), Vol. 1, Anchorage, Alaska, USA, 2003, pp. 532–536.

[23] R. Stewart, RFC 4960, Stream Control Transmission Protocol (SCTP) (Sep. 2007).
URL http://www.ietf.org/rfc/rfc4960.txt

[24] R. Stewart, Q. Xie, M. Tuexen, S. Maruyama, M.Kozuka, RFC 5061, Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration (Sep. 2007).
URL http://www.ietf.org/rfc/rfc5061.txt

[25] IETF signaling transport (sigtran) working group webpage.
URL http://tools.ietf.org/wg/sigtran/

[26] S. J. Koh, Q. Xie, Mobile SCTP (mSCTP) for IP handover support, IETF draft, <draft-sjkoh-msctp-01.txt>, draft expired (Oct. 2005).

[27] M.Tuexen, R. Stewart, P. Lei, E. Rescorla, RFC 4895, Authenticated chunks for the Stream Control Transmission Protocol (SCTP) (Aug. 2007).
URL http://www.ietf.org/rfc/rfc4895.txt

[28] J. Manner, M. Kojo, RFC 3753, Mobility related terminology (Jun. 2004).
URL http://www.ietf.org/rfc/rfc3753.txt

[29] R. Stewart, M. Tuexen, G. Camarillo, RFC 5062, security attacks found against the Stream Control Transmission Protocol (SCTP) and current countermeasures (Sep. 2007).
URL http://www.ietf.org/rfc/rfc5062.txt

[30] M. Liebsch, A. Singh, H. Chaskar, D. Funato, M. Tuexen, RFC 4066, Candidate Access Router Discovery (CARD) (Jul. 2005).
URL http://www.ietf.org/rfc/rfc4066.txt

[31] M. Honda, Y. Nishida, J. Nakazawa, H. Tokuda, Performance enhancement of transport layer handover on single-homed mobile nodes, IEICE Transactions on Communications E90B (10) (2007) 2683–2692.

[32] J. Iyengar, P. Amer, R. Stewart, Concurrent Multipath Transfer using SCTP multihoming over independent end-to-end paths, IEEE/ACM Transactions on Networking 14 (10) (2006) 951–964.

[33] Ł. Budzisz, R. Ferrús, K.-J. Grinnemo, A. Brunstrom, F. Casadevall, An analytical estimation of the failover time in SCTP multihoming scenarios, in: Proc. of the IEEE Wireless Communications and Networking Conference (WCNC 2007), Hong Kong, 2007, pp. 3929–3934.

[34] The network simulator ns-2.
URL `http://www.isi.edu/nsnam/ns/`

[35] C. Casetti, C.-F. Chiasserini, R. Fracchia, M. Meo, AISLE: autonomic interface selection for wireless users, in: Proc. of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM06), Niagara Falls, NY, USA, 2006, pp. 42–48.

[36] IEEE 802.21 web page.
URL `http://www.ieee802.org/21/`

[37] IETF detecting network attachments working group webpage.
URL `http://tools.ietf.org/wg/dna/`

[38] J. Puttonen, G. Fekete, J. Makela, T. Hamalainen, J. Narikka, Using link layer information for improving vertical handovers, in: Proc. of the 16th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2005 (PIMRC 2005), Vol. 3, Berlin, Germany, 2005, pp. 1747–1752.

[39] S. Krishnan, N. Montavont, E. Njedjou, S. Veerepalli, A. Yegin, RFC 4957, link-layer event notifications for Detecting Network Attachments (Aug. 2007).
URL `http://www.ietf.org/rfc/rfc4957.txt`

[40] F. Zhu, J. Mcnair, Vertical handoffs in fourth-generation, multinetwork environments, IEEE Wireless Communications 11 (3) (2004) 8–15.

[41] S. J. Koh, M. J. Chang, M. Lee, mSCTP for soft handover in transport layer, IEEE Communications Letters 8 (3) (2004) 189–191.

[42] M. J. Chang, M. Lee, S. J. Koh, Transport layer mobility support utilizing link signal strength information, IEICE Transactions on Communications E87B (9) (2004) 2548–2556.

[43] Ł. Budzisz, R. Ferrús, F. Casadevall, Study on transport layer handover using SCTP, in: Proc. of the 8th International Symposium on Wireless Personal Multimedia Communications (WPMC 2005), Aalborg, Denmark, 2005, pp. 1551–1556.

[44] C. Dovrolis, P. Ramanathan, D. Moore, Packet-dispersion techniques and a capacity-estimation methodology, IEEE/ACM Transactions on Networking 12 (6) (2004) 963–977.

[45] Sprobe tool.
URL http://sprobe.cs.washington.edu/

[46] S. Mascolo, L. A. Grieco, R. Ferorelli, P. Camarda, G. Piscitelli, Performance evaluation of Westwood+ TCP congestion control, Performance Evaluation 55 (1–2) (2004) 93–111.

[47] EURANE, enhanced UMTS radio access network extensions for ns-2.
URL http://www.ti-wmc.nl/eurane/

[48] H. Holma, A. Toskala, WCDMA for UMTS: Radio Access for Third Generation Mobile Communications, 3rd Edition, J. Wiley and Sons, 2004.

[49] C. Casetti, G. Galante, R. Greco, Load balancing over multipaths using bandwidth-aware source scheduling, in: Proc. of the 7th International Symposium on Wireless Personal Multimedia Communications (WPMC 2004), Abano Terme, Italy, 2004, pp. 1551–1556.

[50] M. Fiore, C. Casetti, G. Galante, Concurrent multipath communication for real-time traffic, Computer Communications, Special Issue on Concurrent Multipath Transfer 30 (17) (2007) 3307–3320.

[51] F. Perotto, C. Casetti, G. Galante, SCTP-based transport protocols for concurrent multipath transfer, in: Proc. of the IEEE Wireless Communications and Networking Conference (WCNC 2007), Hong Kong, 2007, pp. 2969–2974.

[52] J. Iyengar, P. Amer, R. Stewart, Receive buffer blocking in Concurrent Multipath Transport, Tech. Report TR2005-10, CIS Dept., University of Delaware (Jan. 2005).

[53] J. Iyengar, P. Amer, R. Stewart, Receive buffer blocking in concurrent multipath transfer, in: Proc. of IEEE Global Telecommunications Conference (GLOBECOM 2005), Vol. 1, St. Louis, United States, 2005, pp. 121–126.

Łukasz Budzisz received a M.Sc. in Electronics and Telecommunication Engineering from Technical University of Łódź, Poland, in 2003. In 2004 he joined the Department of Signal Theory and Communications at Universitat Politecnica de Catalunya (UPC), Spain, where he is currently working toward the Ph.D. degree. In 2006 and 2007 he was a visiting researcher in the Department of Computer Science at Karlstad University, Sweden, and Protocol Engineering Lab at the Computer and Information Science Department of University of Delaware, respectively.

His research interests include performance evaluation of new transport layer protocols, mobility and resource management of heterogeneous IP-based wireless networks.

Ramon Ferrús received a M.Sc. and Ph.D. in Telecommunications Engineering from Universitat Politecnica de Catalunya (UPC), Spain, in 1996 and 2000, respectively. In 1996 he joined UPC, where he was an Assistant Professor from 1996 to 2002. He is currently an Associate Professor in the Signal Theory and Communications Department.

Currently his research interest is focused on architectures, QoS, mobility and resource management in the context of heterogeneous IP-based mobile communications systems. He is co-author of more than 40 papers published in international journals/magazines and international conference proceedings.

Anna Brunstrom received a B.Sc. in Computer Science and Mathematics from Pepperdine University, CA, in 1991, and a M.Sc. and Ph.D. in Computer Science from College of William & Mary, VA, in 1993 and 1996, respectively. She joined the Department of Computer Science at Karlstad University, Sweden, in 1996, where she is currently a Full Professor and Research Manager for the Distributed Systems and Communications Research Group.

She has a background in distributed systems, but her main area of work over the last years has been in computer networking with a focus on transport protocol design, QoS issues, cross-layer interactions and wireless communication. She has authored/coauthored five book chapters and over 60 international journal and conference papers

 **Karl-Johan Grinnemo** is a consultant at TietoEnator R&D Services in Karlstad, Sweden, where he is involved in projects concerning the design and implementation of signaling solutions for next generation UMTS networks (i.e., Evolved UMTS). He is also actively involved in the research on transport protocol solutions for IP-based signaling networks at Karlstad University. He received a M.Sc. in Computer Science and Engineering from Linköping Institute of Technology (LiTH), Linköping, Sweden, in 1994, and a Ph.D. in Computer Science from Karlstad University, Karlstad, Sweden, in 2006.

His research interests are in next-generation, convergent telecommunication networks.

 **Roberta Fracchia** graduated with summa cum laude from Politecnico di Torino, in Telecommunications Engineering in July 2002. In 2003–2004 she was a visiting researcher in the Information Systems Networking Lab at the Electrical Engineering Department of Stanford University. In 2006 she obtained the Ph.D. degree at Politecnico di Torino. In December 2006 she joined Motorola Labs in Paris where she is currently a senior research engineer.

Her research interests include performance evaluation of transport layer protocols, architecture and Radio Resource Management of next generation wireless networks.

 **Giulio Galante** obtained his Dr. Ing. degree in Electronics Engineering in 1998 and his PhD in Telecommunications Engineering in 2003 both from Politecnico di Torino.

During summer 2000, he was an intern with Lucent Technologies, Bell Labs, Holmdel, NJ. During 2001-2002, he visited the research group of Prof. Nick McKeown at Stanford University, CA. From 2003 to 2007 he was with Istituto Superiore Mario Boella, Turin, Italy. Since July 2007, he has been a consultant with Nokia Siemens Networks, Milan, Italy.

His research is mainly focused on the performance evaluation of wireless extensions to the TCP/IP protocol suite, and switching architectures based on off-the-shelf hardware and open-source software.

Ferran Casadevall received a M.Sc. and Ph.D. in Telecommunications Engineering from the Universitat Politecnica de Catalunya (UPC), Spain, in 1977 and 1983, respectively. In 1978 he joined UPC, where he was an Associate Professor from 1983 to 1991. He is currently a Full Professor in the Signal Theory and Communications Department.

In the last fifteen years, his research interest has included cellular and personal communication system, multipath transceiver design (including Software Radio techniques), Mobility and Radio Resources Management, End to end QoS issues. During the last ten years he participated in more than 30 research-projects , being the Project Manager of three of them: ARROWS, EVEREST and AROMA. (see http://www.gcr.tsc.upc.edu for details) Prof. Casadevall has published around 150 technical papers in both international conferences and magazines.