Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computer Networks 50 (2006) 128–144
Computer Networks
www.elsevier.com/locate/comnet

A performance analysis of context transfer protocols for QoS enabled internet services

N. Bartolini [a,*], E. Casalicchio [b]

[a] Department of Computer Science, University of Rome "La Sapienza", Via Salaria 113, Rome 00198, Italy
[b] Department of Computer Science, University of Rome "Tor Vergata", Rome 00133, Italy

Received 10 January 2005; received in revised form 10 January 2005; accepted 17 February 2005
Available online 5 July 2005

Responsible Editor: Dr. G. Morabito

Abstract

In nowadays wireless networks, mobile users frequently access Internet services that are often based on information concerning the application context and service status. In presence of mobility, the procedure of service handover, may require a restart of the ongoing service, if the necessary context information is not properly transferred to the new point of access. Context transfer procedures introduce additional overheads to handovers possibly affecting the quality of service perceived by mobile users and making handovers very critical. In this paper the need for efficient protocols for transferring service context and profile related information is pointed out with reference to many mobile internet services, and the possible scenarios are differentiated on the basis of the handover triggering mechanisms. A performance model to compare these mechanisms, when context transfer protocols run on top of IPv6 with fast handover, is proposed. Numerical results point out the necessity to adapt the triggering mechanism to the size of the context data.

Keywords: Context transfer protocol; IPv6; Fast handover; Performance analysis

1. Introduction

The introduction of several multimedia services in new generation of wireless networks, brought about the need to develop efficient methods to manage the mobility of users. Nowadays internet services are often session oriented, delay bounded (or real-time) and context sensitive. Just to

* Corresponding author. Tel.: +39 06 49918357; fax: +39 068541842.
E-mail addresses: novella@dsi.uniroma1.it (N. Bartolini), casalicchio@ing.uniroma2.it (E. Casalicchio).

1389-1286/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2005.02.009

mention some, VoIP, multimedia streaming, on-line games, on-line transactions and many Content Delivery Networks (CDN) related services are often session oriented, delay bounded and context sensitive. In wired networks, the use of broadband technologies has a significant impact on the user's perceived Quality of Service (QoS) making Service Level Agreements (SLA) achievable. On the contrary, in wireless networks the introduction of broadband wireless connectivity is not sufficient to guarantee the fulfillment of QoS requirements mostly due to users movement across the network coverage areas managed by different access routers (AR). Handover requests may be issued during critical service phases for which the avoidance of service disruption is mandatory, and the connection must be seamlessly handed off from a point of access to another. The fast handover mechanism, introduced to reduce the packet losses during handovers, needs to be enhanced with proper mechanisms to preserve the service continuity. In context and session based services, the realization of a handover is not only a matter of keeping a connection alive during users movements, but also of transferring the necessary information to avoid the re-establishment of a service session every time the user reaches a new point of access. The re-establishment of a service session causes the repetition of the service protocol message flow from scratch and is necessary if the information to keep the service alive is unavailable when a handover to a new point of access occurs. Thence service continuity and context transfer during handover procedures are very critical for delay sensitive and context dependent applications.

The IETF SeaMoby working group identifies general motivations for Context Transfer [13] and defines a Context Transfer Protocol (CTP) [14]. In Section 2, we consider critical scenarios like the one of Content Delivery Networks (CDN) supporting mobile users, in which context-aware handovers are of significant impact on quality of service. In Section 3, we show the interaction between CTP and Mobile IPv6 protocol, with fast handover mechanisms to reduce packet losses. Since understanding how and when the context transfer can be activated by a mobile node or access router is fundamental to give a performance model and evaluation of the CTP, in Section 4 we describe the CTP message flow in tree different cases: dummy (post-handoff) context transfer, mobile initiated context transfer and access router initiated context transfer.

A performance model of the CTP is given in Section 5, where performance is evaluated in terms of bandwidth occupation, packet loss, percentage of packets that violate the SLA, context transfer time and completion time of the protocol message flow. Section 6 concludes the paper.

2. Motivation for context transfer

All the information needed to negotiate, establish and manage network services may be considered part of the context to be transferred when a Mobile Node (MN) issues a handover request during an ongoing service.

The context data include:

- authentication, authorization, and accounting information [13] needed to permit the re-authentication of the mobile host and the mobile host's authorization to access the network service from a new subnet;
- header compression [13] information that is necessary to avoid the repetition of messages between the last hop router and the mobile host;
- network QoS information to avoid the re-negotiation and re-establishment of QoS agreements between the mobile node and routers;
- application level QoS parameters, e.g. maximum end-to-end perceived latency, level of image resolution (e.g. high-level resolution for laptop and low-level resolution for enlarged mobile phone/palmtop), maximum/minimum bit-rate for streaming sessions, security specification (e.g. which suite of encryption algorithms is allowed/used), service authentication (e.g. certificate, list of certification authorities, list of trusted servers);
- session state information, e.g. the list of items in the basket or the phase that most likely will be entered next, for an e-commerce session or the next chunk of data needed in a streaming

session, the next game phase for an on-line game session, the mailbox state or the file system information of an e-storage account.

We focus our attention on session oriented internet services/applications, that are geographically distributed over the devices of a CDN [4] or on a distributed proxy system. We consider a mobile user accessing an on-line streaming content or buying a book online. The mobile user accesses the services by means of an AR, a network layer device that implements functionality of wireless access point, router/gateway and replica selection (using any-cast mechanisms [6,11,24,1]). When the MN starts a web session, requesting the first HTML page of the target site, the AR selects the best suited replica server to fulfill the request of the MN. This selection process, is usually based on network and server centric performance metrics, such as network load, number of hops, network QoS parameters or replica servers load state.

When a mobile node changes location, a finer grain replica selection may be performed if the access router at the new location has context-aware information, such as: ongoing session phase, application layer QoS parameters, graphic representation capability. Hence the access router (a trusted AR) may enhance its selection capabilities keeping into account context-related information. If handover procedures were conducted without transferring any context related information, those parameters should be re-defined from scratch whenever the mobile host reaches a new access point. The re-negotiation of these parameters may require longer time than what is needed to perform the handover. The best solution is to transfer context from the access router of the region from which the mobile node is coming (pAR) to the access router of the area targeted by the mobile node (nAR).

3. Mobility management mechanisms

Though context transfer may reduce latency in handoff management by reducing the number of messages needed for service re-establishment, mobility management must also be supported by proper mechanisms at the network access level. In order to evaluate the impact of the context transfer protocol on performance, its interaction with the underlying mobility management protocol must be considered and investigated. In the following subsections we discuss about mobility management schemes and we focus our attention on the IPv6 protocol with fast handover mechanisms.

3.1. Mobility management schemes

To have an efficient and transparent mobility management, different problems must be solved at different layers of the TCP/IP protocol stack. Content based mobility management can be achieved at the application layer by transferring content related information among access points during handover procedures (this can be performed directly or via some intermediate protocol like SIP [9]). The transport layer could be properly tuned to deal with mobile scenarios in a wireless environment. In wireless networks, the error rate is considerably higher than in the wired case, therefore a missing ACK does not necessarily represent a situation of congestion. Network and data link layer should be involved in mobility management as well. The most common solutions to mobility management give the network layer a predominant role. The network layer is in fact the best suited layer to perform an efficient and application-transparent solution, like with the mobility extensions of IPv4 or in the proposed mobility support in IPv6 [10]. Data link level is important as well, as it faces key aspects of mobility such as radio propagation models, errors and delays, and must offer the network level a clear and hopefully medium-independent interface. Although every wireless technology could be very different, some general guidelines could be applied.

The higher the number of wireless access points, the higher the coverage and the service/bandwidth availability. Shrinking the coverage area of each wireless access point increases the frequency of handovers, therefore proper (ad upper-level oriented) protocols to manage mobility are needed.

The upper layers of the protocol stack should see a lower rate of handovers than the wireless

access layers, because there is no need to make all handovers visible to the upper layers. It is preferable to make data link handovers as transparent as possible to the upper layers of the protocol stack. This could be achieved by grouping several connected wireless access points into a single logical entity.

In this deployment scheme, one or more access points are grouped together as they are connected to the Internet via the same Access Router (AR). An example related to 802.11 networks can be found in [15].

In the subsequent discussion we consider only handovers that occur between different logical cells (each cell representing the logical coverage area of a single access router), i.e. handovers that need to be managed at the network level.

3.2. Mobile IPv6 with fast handover

Mobile IPv6 [10] defines the protocol operations and messages to achieve intra and inter-domain mobility within IPv6.

Auto-configuration [21,22] is an essential part of IPv6, and it is also used by a mobile node to obtain a new Care of Address (nCOA) when it handovers to a new AR: the mobile node sends a Router Solicitation message (RtSol), and the AR responds with a Router Advertisement message (RtAdv) which contains the information needed by the mobile node to construct its nCOA as a global unicast address [17,7], an address which could be routed over the Internet and used to communicate with a correspondent node (CN).

Every IPv6 node also has one or more link local address: in a wireless environment an address with this scope can be used only to communicate with a node in the same wireless cell, but does not require any advertisement from the router; there are also site local addresses (to be deprecated [5] and substituted by unique local IPv6 unicast addresses [8]) which are defined the same way as global addresses starting from a RtAdv message. As we are interested in inter-domain communication we refer only to global addresses, although our analysis could be applied to local addresses as well.

An IPv6 global address is composed by two parts, each 64 bits long: the first one basically defines the network the address belongs to, and the latter is obtained in a deterministic way from an unique interface identifier: an IEEE 802.X MAC address is a typical example [17]. This makes easy to track a user even when he/she changes network (with this approach, if privacy is a concern a privacy oriented extension should be used [16]). This approach increases latency, as the nCOA could be not unique and duplicate address detection (DAD) must be performed. The presence of duplicate addresses in the new wireless cell could cause the failure of the entire fast handover mechanism.

When the mobile node obtains its nCOA, it sends two messages to the CN: the Home Test Init, sent via the Home Agent, and the Care of Test Init, targeted directly.

The CN responds respectively with a Home Test message (routed through the Home Agent) and a Care of Test message (directly). By combining the information contained in all these messages, the mobile node determines a value used to cryptographically mark the binding update (BU) message the mobile node will send to the CN: this procedure is intended to avoid that a malicious node could move the mobile node out of the Internet, by sending fake BU messages. Detailed discussion and proposed optimizations can be found in [10].

So, when a handover occurs, the mobile node must first obtain a nCOA, and then start the return routability procedure. After this procedure, the mobile node sends the BU message to inform the CN of its nCOA. At the end of these steps, the CN and the mobile node can communicate directly, without using the mobile node Home Agent as an intermediate hop (a process called triangular routing, which could increase the round trip time significantly).

The time needed to complete this procedure, which is also the time to obtain and communicate the nCOA, is called *handoff latency*, and it is worth trying to reduce it as much as possible, in order to minimize service degradation due to handovers. The fast handover extension for IPv6 is intended to minimize the handover latency [18]. First, the current AR not only broadcasts its RtAdv messages (periodically or as an answer to a RtSol message) but also broadcasts advertisement from a

confining AR, by a Proxy Router Advertisement message (PrRtAdv). A PrRtAdv could also be the answer for a Proxy Router Solicitation message (PrRtSol), sent out by a mobile node when it detects, by some layer-two indicator, that a handoff is likely to occur.

When the mobile node gets a PrRtAdv, it has everything it needs to create the nCOA. It communicates the nCOA to the current AR, via a Fast Binding Update message (FBU), so a tunnel between the current AR (pAR, as the current AR is about to be the previous AR) and the new AR (nAR) could be established. This bidirectional tunnel is used to route packets from the nAR to the pAR. (In some cases the mobile node moves so fast that the mobile node gets connected to the nAR before a FBU could be sent to the pAR: in such a case the nAR must relay the nCOA to the pAR.)

As a final step, if the mobile node has sent the FBU to the pAR but is still connected to the current AR, it sends a Fast Neighbor advertisement message (FNA) to the nAR, to let it know the link layer address of the mobile node and to start buffering of packets that will arrive to the nAR before the mobile node actually performs the handover. If the FBU message is sent to the pAR by the nAR this step is not required, because the mobile node is already connected to the nAR and the nAR is already aware of mobile node's presence and link layer address.

It is worth pointing out that the whole fast handover mechanism can be applied only if the wireless interface is provided with an indicator of link layer events such as the discovery of a new AR or the degradation of signal quality to the current AR.

4. The context transfer protocol

In this section we shortly describe the Context Transfer Protocol (CTP) [14] that has been proposed by the IETF SeaMoby working group. We focus on the possible message flows generated to grant transparent mobility management to a mobile node accessing a CDN service [3], e.g. an e-commerce site or a streaming video, provided by means of a content delivery distributed infrastructure. If CDN services are ongoing while the user moves, context transfer is needed at different layers by different entities.

The exchange of context information, related to the user and to the ongoing service, may help the replica server selection process [24,19] when the mobile node enters a new network through a different access router. At the data link and network layer the transfer of AAA and compression header information helps to reduce the latency in connection re-establishment.

When the mobile node moves to a new AR all these data must be transferred from the previous AR and not obtained by an additional message exchange between the new AR and the mobile node. The transfer of context allows to avoid an unnecessary burst of data packets as the node gets connected to the new AR (this is particularly important if the mobile node moves fast, i.e. if it changes its AR frequently). Context transfer also minimizes the number of application data packets which cannot be processed properly due the lack of context-oriented information (in such a case the new AR could postpone the packet processing or apply a default treatment, both two sub-optimal approaches). The number of these packets could be used as a metric for evaluating proposed solutions.

The Context Transfer Protocol consists of few messages:

CTAR: the Context Transfer Activate Request message is sent by the mobile node to the nAR to initiate the actual context transfer from the pAR, whose IP address is contained in the CTAR message. This message is always sent by the mobile node after a handoff, because the node does not know if the context has already been transferred to the nAR.

CTR: the Context Transfer Request message is sent by a nAR to a pAR to ask for the context data to be transferred.

CTD: the Context Transfer Data message contains all the context a pAR will transfer to a nAR, and is sent after the reception of a CTR. Each context feature is identified by a Feature Profile Type, a 16 bits integer which defines the meaning of the variable-length context data following it.

The value of these fields have been defined by IANA [12].

CTAA: the Context Transfer Activate Acknowledge message is sent from a receiver of a CTAR (a nAR) to the mobile node to acknowledge the reception of the CTAR, if the mobile node has requested so.

CTDR: the Context Transfer Data Reply message is the optional acknowledge message sent from the receiver of a CTD (the nAR) to the sender (the pAR).

CTC: the Context Transfer Cancel message is sent by the nAR to the pAR to request the end of the context transfer process, when the nAR realizes that the process cannot be timely completed.

The Context Transfer Protocol could be initiated by one of the ARs or by the mobile node, as a trigger (a Context Transfer Trigger) arises.

The pAR may initiate the CTP if it someway detects that the mobile node is about to handoff to another AR: in such a case it predictively sends the CTD message to the nAR; when the mobile node actually handovers to the nAR it sends a CTAR message anyway, although the nAR already possesses the context data.

The same process could be initiated by the nAR when it detects that a mobile node is about to get connected to it: the nAR sends a CTR message to the pAR before the mobile node sends the CTAR message, so the CTD message (in reply to the CTR message) is received by the nAR before the time it would have been received if the nAR had waited for the CTAR message from the mobile node.

These first two scenarios are predictive, that is the context data transfer is initiated more or less before the actual handoff, and handoff latency is reduced.

The context transfer procedure can also be performed reactively. When the mobile node starts the handover at the data link layer, a CT Trigger arises so that the mobile node sends the CTAR message to the nAR, which in turn issues a CTR message to the pAR and receives from it the CTD message. This is a worst case scenario, showing the longest time to transfer the context.

In each scenario, there could be some context data that must be transferred after the handoff. As an example, if a context type contains the total number of bytes sent or received by the mobile node (an information that could be used for traffic accounting purposes), this value could vary from the moment the CTD message is sent to the nAR to the moment the actual handoff of the mobile node takes place, so the related context type must be (re)transferred after the actual handoff.

4.1. Interactions between fast handover and context transfer protocol

The context transfer is always triggered by means of a Context Transfer Trigger. The current version of the draft [14] does not define exactly what a CT Trigger is, although it seems to envision that the CT Trigger is a level two (data link) trigger.

We believe that the CT Trigger could be better defined as a network level trigger. By doing so, we have a trigger which could be managed by the mobile node operating system, without requiring a hook provided by the wireless interface firmware.

The main idea is to use the Fast Handover messages as CT Trigger. As an example, if a pAR sends a PrRtAdv message for a nAR, it should also send a CTD to the nAR it is proxying advertisements for. After the reception of the PrRtAdv message the mobile node has everything it needs to define a nCOA and eventually the pAR establishes a tunnel to the nAR where incoming packets must be treated according to parameters (e.g. SLAs) defined in the context.

We identify the following possibilities to detect a handover and initiate the context transfer procedure:

Dummy Context Transfer Protocol (D-CTP). This is the completely reactive case when the fast handover mechanism does not take place, so the context transfer is initiated after the handoff of the mobile node from the pAR to the nAR: the nAR sends a RtAdv message to the mobile node which constructs its nCOA and sends a CTAR message to the nAR, which in turn sends a CTR message to the pAR. Fig. 1(a) depicts this scenario. In this simple case we assume that the context information are used by the new access router only to get control over the ongoing session and future requests. In this case, context information

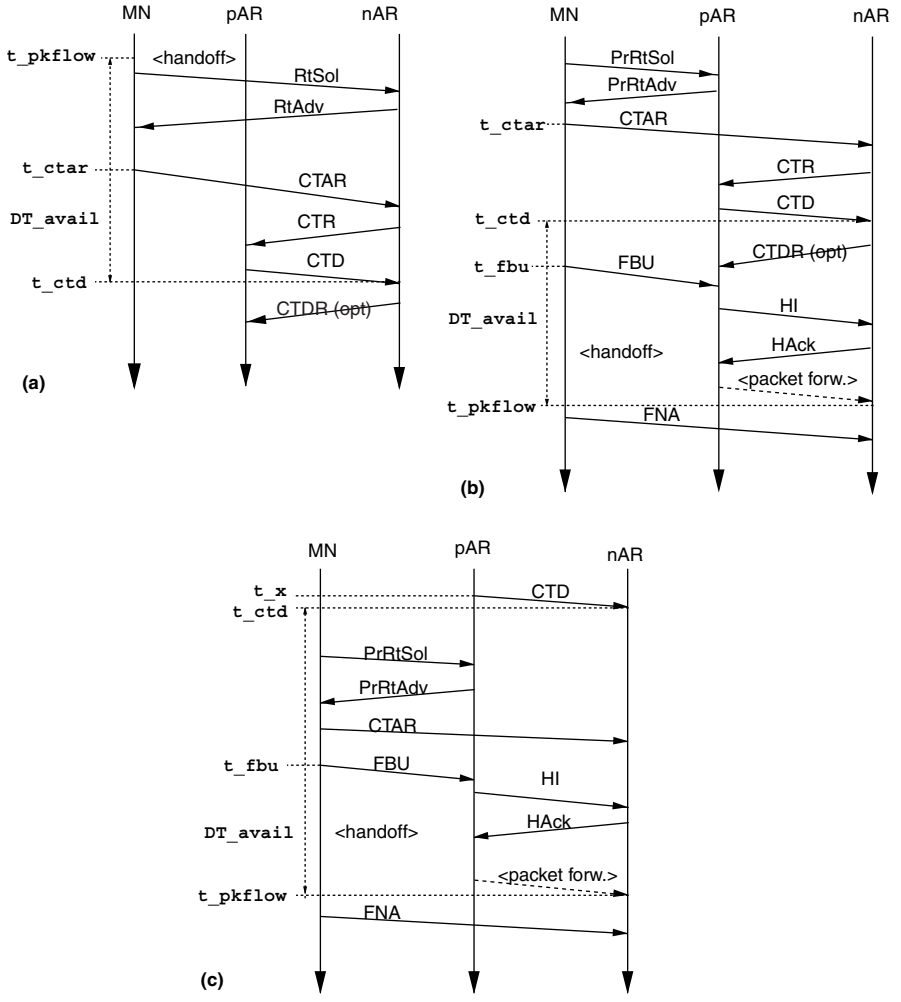

Fig. 1. Context Transfer Protocol scenarios: dummy (a), mobile initiated (b) and access router initiated (c).

cannot be used to perform request admission control when the service is handed over the nAR. The management of the ongoing active session phase is outside the control of the new access router that can get the control over the next session phase. No tunnelling is performed between the pAR and nAR so packets can be lost and service degradation could be experienced if the handoff happens in the middle of an active phase of a session.

Mobile Node Initiated Context Transfer Protocol (MN-CTP). The mobile node receives a PrRtAdv message from the pAR, and sends a CTAR to the nAR because it realizes that a handoff to the nAR is about to begin. It is worth noting that the mobile node could receive more than one PrRtAdv message on behalf of different nARs, because the pAR could advertise (and usually do advertise) all the confining nARs, and the mobile node could send the CTAR to one or more advertised nARs, without knowing in advance which one it will handoff to (or *if* an handoff will take place): as the mobile node is still connected to the pAR, the pAR will receive all the CTAR messages and route them to the different ARs. If a target AR honors the Context Transfer Protocol, it sends a CTR to the current AR after the reception of the CTAR. Fig. 1(b) shows the most favorable message flows for this scenario, when the actual

handoff takes place after the context data have been transferred. The mobile node initiated case is designed to allow the new access router to use the context information to decide whether to manage or deny service to the new mobile node. A service denial could be managed in different ways and a dissertation is outside the scope of this paper.

AR Initiated Context Transfer Protocol (nAR-CTP). The most predictive option is when the pAR (when it still is the current Access Router) sends a CTD describing a mobile node's context to one or more of its confining ARs. This can be done periodically or as a consequence of a CT Trigger. The receiving ARs cache this context, to be able to use it immediately after a handoff takes place. The context data are considered valid for a short period of time (possibly depending on the context type), after which they are deleted; this soft-state approach is envisioned both for scalability and because the context data could (although slowly) change. Frequency of the CTD messages and cache duration must be defined accordingly to handoff frequency, available bandwidth for inter-AR communication and context data semantics.

In this scenario the first step is always taken by the pAR, which predictively sends CTD to one or more nAR candidates. The converse usually cannot happen, because the CTR message must be authenticated by means of an authorization token supplied by the mobile node in the CTAR message. Therefore, as long as the CTAR message is not received, the nAR does not possess the token (and probably does not even suspect the mobile node's existence or proximity).

Fig. 1(c) shows the flow of messages when the pAR sends a CTD before the mobile node sends a PrRtSol message. Alternatively the pAR can trigger a PrRtSol message and send the CTD to the candidate nARs.

4.2. Context transfer protocol and wireless architecture

The described approaches assume very little about the wireless network architecture. As an example, we can consider a hierarchical architecture of ARs. When two confining ARs are descendants of the same parent AR, as they share some management information bases, they can share the mobile node's context. In this case, the context transfer protocol takes places only when the mobile nodes move from a pAR to a nAR which do not have a parent AR in common. In the most "flat" case, all ARs share a context database of management information, which holds all context transfer data for each mobile node in the wireless network. When the mobile node first enters the wireless network, a new entry in this management database is created, and information are updated as long as the node moves across the network. Context information are deleted when the mobile node abandons the network.

In this scenario a CTAR message is used by the nAR to obtain a permission to update the centralized context database (if the interaction with the mobile node—as it dwells in the nAR cell—requires a change in the context information); a CTDR message is a request to a pAR to synchronize its local context data with the centralized data, and a CTD message is not only a copy of these data, but also the grant for a nAR to update the centralized copy, to avoid that two ARs update simultaneously the centralized management database.

5. Performance analysis of CTP

We introduce a performance model to evaluate the cost of CTP in terms of: consumed bandwidth and number of packets that have been lost or erroneously processed according to the default method, without considering the necessary context information.

At least three entities are involved in CTP: the mobile node, the previous access router and one or more new access routers. Thus, as represented in Fig. 2, we distinguish among the amount of data exchanged on the side of the mobile node, B^{MN}, of the previous access router B^{pAR}, and of the new access router B^{nAR}.

When a mobile node handovers to a new mobile access router, $N_{\mathtt{lost}}$ packets could be lost, and $N_{\mathtt{default}}$ packets could be erroneously served by default, without considering context related information.

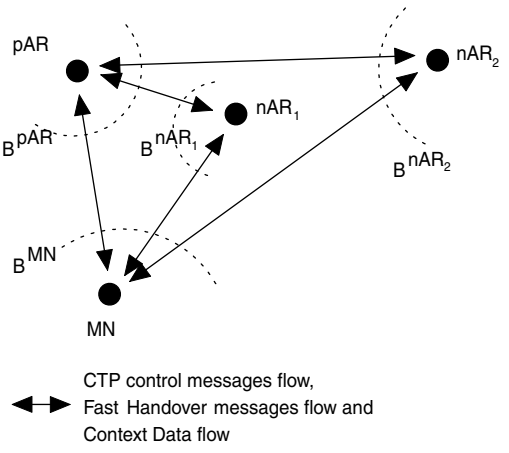

Fig. 2. Bandwidth consumption in CTP.

If an access router receives a packet before being able to consider the context related information, it processes the packet according to the default procedure, until the necessary information becomes available. When the AR receives context information and re-establishes the proper QoS level, packets will be properly prioritized.

5.1. Bandwidth consumption analysis

The Context Transfer Protocol works on an UDP-based transport layer. Our model is based on the assumption that CTP messages must fit the maximum segment size (MSS) of a data link frame (and obviously must be contained in one UDP/IP packet), to reduce the packet fragmentation and reassembly overhead. For synchronization messages it is easy to fit the MSS, nevertheless context data could need a proper encoding and/or compression. Which technologies to adopt to describe and encode the context, is an open issue and this choice will have a non-negligible impact on protocol performances.

Each CTP message travels over an UDP segment that introduces an 8 bytes long overhead $O_{\mathtt{udp}}$. The UDP segment is delivered using an IP packet, with a 20 bytes long overhead $O_{\mathtt{ip}}$. An additional overhead $O_{\mathtt{frame}}$ is also introduced to deliver the IP packets over the data link layer ($O_{\mathtt{frame}} = 18$ bytes for ethernet frames).

Therefore the total overhead that is needed to send a CTP message is $O = O_{\mathtt{udp}} + O_{\mathtt{ip}} + O_{\mathtt{frame}}$.

In our analysis we give a formulation of upper bounds on the total amount of data exchanged on the network links by each participant to perform the context transfer procedure. We use the following notation: $B^{\mathtt{participant}}_{\mathtt{scenario}}$ is the upper bound on the total amount of data sent/received by $\mathtt{participant}$, where $\mathtt{participant} \in \{\mathtt{MN}, \mathtt{pAR}, \mathtt{nAR}\}$ and the triggering mechanism is $\mathtt{scenario} \in \{\mathtt{dummy}, \mathtt{MN}_{\mathtt{init}}, \mathtt{AR}_{\mathtt{init}}\}$.

In the following expressions S is the maximum size of the messages that are exchanged to perform the context transfer in the different scenarios. $s_{\mathtt{ctd}}$ is the size of the message containing context data and k is the number of new candidate access routers.

In the worst case, the pAR will complete the context transfer with all k candidates nARs. In a well-designed architecture the nAR or pAR should abort the context transfer when it is sufficiently clear that the mobile node will not enter the service area of the nAR.

We now formulate $B^{\mathtt{participant}}_{\mathtt{scenario}}$ for the different entities and different scenarios.

$$B^{\mathtt{MN}}_{\mathtt{dummy}} = 3(S+O), \tag{1}$$
$$B^{\mathtt{pAR}}_{\mathtt{dummy}} = [2(S+O) + (s_{\mathtt{ctd}} + O)], \tag{2}$$
$$B^{\mathtt{nAR}}_{\mathtt{dummy}} = 3(S+O) + [2(S+O) + (s_{\mathtt{ctd}} + O)], \tag{3}$$

$$B^{\mathtt{MN}}_{\mathtt{MNinit}} = (4+k)(S+O), \tag{4}$$
$$B^{\mathtt{pAR}}_{\mathtt{MNinit}} = 3(S+O) + \{k[2(S+O) + (s_{\mathtt{ctd}} + O)] + 2(S+O)\}, \tag{5}$$
$$B^{\mathtt{nAR}}_{\mathtt{MNinit}} = 2(S+O) + [4(S+O) + (s_{\mathtt{ctd}} + O)], \tag{6}$$

$$B^{\mathtt{MN}}_{\mathtt{ARinit}} = (4+k)(S+O), \tag{7}$$
$$B^{\mathtt{pAR}}_{\mathtt{ARinit}} = 3(S+O) + [k(s_{\mathtt{ctd}} + O) + 2(S+O)], \tag{8}$$
$$B^{\mathtt{nAR}}_{\mathtt{ARinit}} = 2(S+O) + [2(S+O) + (s_{\mathtt{ctd}} + O)]. \tag{9}$$

The amount of data sent/received by the MN, given by Eqs. (1), (4) and (7), is directly proportional to the size of synchronization messages S in all scenarios, and also proportional to the number of k candidate nARs, in the mobile node initiated and access router initiated scenarios. In the mobile initiated and in the access router initiated

scenario it is important to operate a correct prediction of a small set of possible future access routers to reduce the bandwidth consumed at the MN that is typically a critical resource. We can also observe that $B^{\mathrm{nAR}}_{\mathrm{scenario}}$, i.e. the amount of data exchanged on the nAR side, given by Eqs. (3), (6) and (9) is directly proportional to the size of context data s_{ctd}. The first terms of Eqs. (3), (6) and (9) give a measure of the bandwidth consumed on the nAR–MN communication channel, while the second terms, give a measure of the bandwidth consumed on the pAR–nAR communication channels.

The bandwidth consumed by the pAR, in the last two scenarios, is a function of the number k of candidate nARs and of the size of context data s_{ctd}. The first terms of Eqs. (5) and (8) give a measure of the amount of data exchanged pAR–MN communication channel, while Eq. (2) and the second terms of Eqs. (5) and (8) measure the amount of data sent/received on the pAR–nARs communication channels, that is a function of s_{ctd} in the dummy scenario and a function of s_{ctd} and k in the mobile node initiated and access router initiated scenarios.

As a numerical example to give a quantitative idea of $B^{\mathrm{participant}}_{\mathrm{scenario}}$ we consider $S = 300$ bytes, $K = 4$ candidate access routers and $O_{\mathrm{frame}} = 18$ bytes. This numerical example is shown in Fig. 3, where the bandwidth consumed on the MN side is not included since it is intuitively independent of the size of the context data. On the mobile node side, the less expensive mechanism in terms of consumed bandwidth is the mobile node initiated, that does not require message exchanges with more than one candidate access router. Therefore, the dummy triggering mechanism consumes less bandwidth on the mobile node side, at the expense a degraded QoS.

In an analogous way, the pAR is the most stressed entity in terms of amount of data that is sent/received, because in the worst case the context will be broadcast to all the nARs that reply to the CTAR message or that are candidates.

Figs. 4–6 show the trend of $B^{\mathrm{participant}}_{\mathrm{scenario}}$ when the number of candidate nARs increases (from 1 to 10) and the context data size has a fixed value of 540 bytes, 1020 bytes and 1500 bytes respectively.

5.2. Packet loss and bad prioritization analysis

Let r be the cumulative rate at which the mobile node and its related correspondent node inject

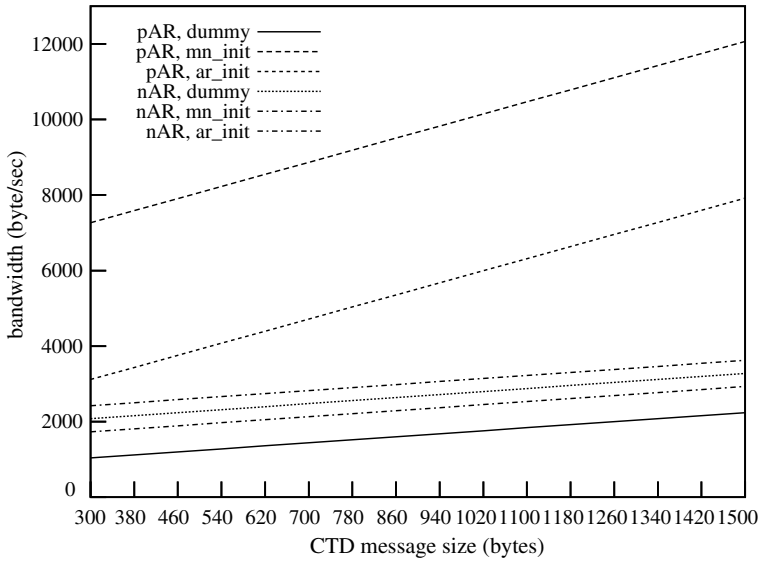

Fig. 3. Data exchanged on the ARs side as a function of the context data size according to different triggering mechanisms.

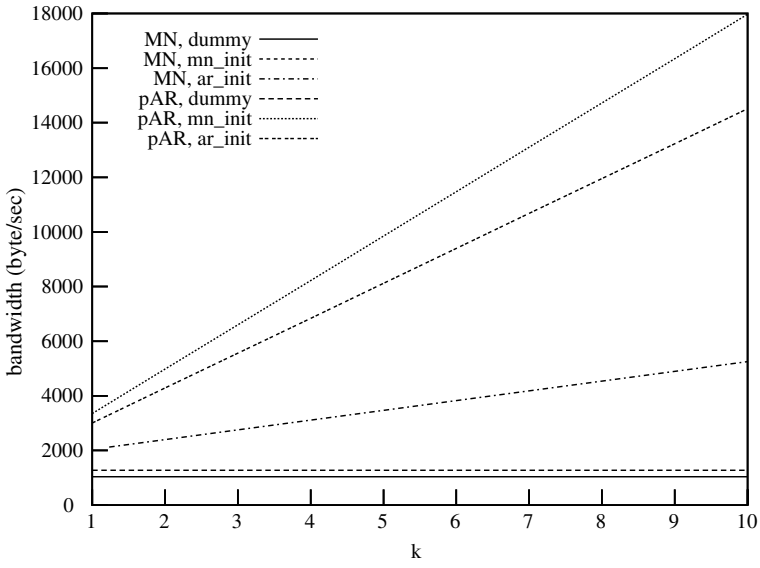

Fig. 4. Data exchanged according to CTP as a function of the number of candidate nARs ($s_{\mathtt{ctd}} = 540$ bytes).

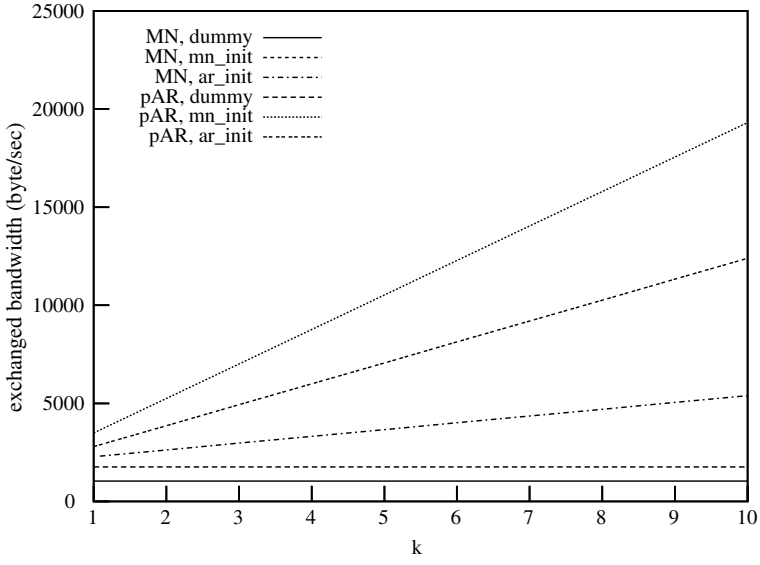

Fig. 5. Data exchanged according to CTP as a function of the number of candidate nARs ($s_{\mathtt{ctd}} = 1020$ bytes).

packets into the network, and D the latency in the communication path between the mobile node and the correspondent node, through the pAR. When a handoff occurs the MN registers itself in the new network and re-establishes the connection with the CN in $D_{\mathtt{conn}}$ time units. In absence of a buffering mechanism between the pAR and the nAR, $N_{\mathtt{lost}}$ packets are lost during handovers,

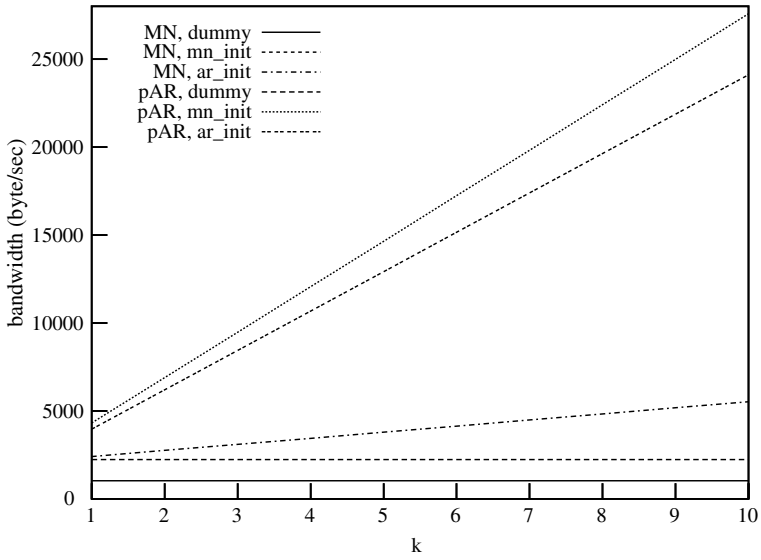

Fig. 6. Data exchanged according to CTP as a function of the number of candidate nARs ($s_{\mathtt{ctd}} = 1500$ bytes).

where $N_{\mathtt{lost}} = (t - t_{\mathtt{hoff}}) \cdot r = D_{\mathtt{conn}} \cdot r$, $t_{\mathtt{hoff}}$ is the handover start time and t, the instant of handover completion. On the contrary, if we use Fast Handover, packets are buffered by the pAR until a tunnel between the pAR and the nAR is established, therefore $N_{\mathtt{lost}} = 0$.

In QoS sensitive applications, even a short sequence of lost packets could result in a SLA violation. For example a random packet loss can be tolerated in a low quality audio/video streaming session but it is prohibited in a secure transaction data flow.

In this paper we only focus on QoS sensitive application, where the condition $N_{\mathtt{lost}} = 0$ is required, therefore the attention is restricted to the mobile initiated or access router initiated scenario. We refer to $t_{\mathtt{ctd}}$ as to the instant in which the context is available to the nAR, and we refer to $t_{\mathtt{pkflow}}$ as to the time the nAR starts processing packets directed from the CN to the MN. The elapsed time between the actual availability of the context data and the moment the first packets directed to the mobile node arrive to the nAR, can be expressed as $\Delta T_{\mathtt{avail}} \triangleq (t_{\mathtt{pkflow}} - t_{\mathtt{ctd}})$.

As shown in Fig. 1 the context transfer begins at the instant $t_{\mathtt{ctar}}$ in the mobile node initiated scenario and at time t_x in AR initiated scenario. The nAR receives the context at time $t_{\mathtt{ctd}}$, the handoff procedure starts at time $t_{\mathtt{fbu}}$ and the nAR starts receiving packets addressed to the MN at time $t_{\mathtt{pkflow}}$. When the context transfer procedure suffers from excessive delays and $\Delta T_{\mathtt{avail}} < 0$, there is a period of time, that is $t_{\mathtt{ctd}} - t_{\mathtt{pkflow}}$, during which a certain number of packets belonging to an ongoing service, is erroneously treated by a default procedure, without considering context related information, thus causing a violation of the agreements on quality. The average number of packets erroneously treated by default is $N_{\mathtt{default}} = -\Delta T_{\mathtt{avail}} \cdot r = -(t_{\mathtt{pkflow}} - t_{\mathtt{ctd}}) \cdot r$. On the other side, if the handover procedure is completed on time, that is, if $\Delta T_{\mathtt{avail}} \geqslant 0$, the SLA will be satisfied and $N_{\mathtt{default}} = 0$.

We can conclude that a sufficient condition for the fulfillment of the SLA is $\Delta T_{\mathtt{avail}} \geqslant 0$.

In the dummy scenario, the context transfer procedures are activated after the completion of the handover at the lower levels of the protocol stack, therefore, by definition, $N_{\mathtt{lost}} > 0$ and $\Delta T_{\mathtt{avail}} < 0$ in such a message flow scenario that cannot be used to improve QoS. In presence of the dummy triggering mechanism (Fig. 1(a)), $\Delta T_{\mathtt{avail}}^{\mathtt{dummy}}$ can be calculated as follows:

$$\begin{aligned}\Delta T_{\mathtt{avail}}^{\mathtt{dummy}} &= t_{\mathtt{pkflow}} - t_{\mathtt{ctd}} \\ &= t_{\mathtt{pkflow}} - t_{\mathtt{ctar}} - \frac{1}{2}\mathrm{RTT}_{\mathtt{MN}} - \frac{S+O}{b_{\mathtt{mn}}} \\ &\quad - \mathrm{RTT}_{\mathtt{AR}} - \frac{S+s_{\mathtt{ctd}}+2O}{b_{\mathtt{ar}}} \\ &= t_{\mathtt{pkflow}} - t_{\mathtt{pkflow}} - \frac{3}{2}\mathrm{RTT}_{\mathtt{MN}} - \mathrm{RTT}_{\mathtt{AR}} \\ &\quad - \frac{3(S+O)}{b_{\mathtt{mn}}} - \frac{S+s_{\mathtt{ctd}}+2O}{b_{\mathtt{ar}}} \\ &= -\frac{3}{2}\mathrm{RTT}_{\mathtt{MN}} - \mathrm{RTT}_{\mathtt{AR}} - \frac{3(S+O)}{b_{\mathtt{mn}}} \\ &\quad - \frac{S+s_{\mathtt{ctd}}+2O}{b_{\mathtt{ar}}}, \end{aligned} \tag{10}$$

where $\mathrm{RTT}_{\mathtt{MN}}$ is the Round Trip Time experienced by a mobile node exchanging packets respectively with the pAR and nAR,[1] $\mathrm{RTT}_{\mathtt{AR}}$ is the RTT experienced by two ARs that communicate each other ($\mathrm{RTT}_{\mathtt{AR}}$ is likely three orders of magnitude greater than $\mathrm{RTT}_{\mathtt{MN}}$, $\mathrm{RTT}_{\mathtt{AR}} \approx$ msec and $\mathrm{RTT}_{\mathtt{MN}} \approx$ μsec), $b_{\mathtt{mn}}$ and $b_{\mathtt{ar}}$ are the effective bandwidths of the link from MN and ARs and of the links between ARs respectively [2]. Eq. (10) shows the dependency of $\Delta T_{\mathtt{avail}}$ exclusively on the communication latency and on the available bandwidth to transfer the context.

In the Mobile Initiated scenario (Fig. 1(b)):

$$\begin{aligned}\Delta T_{\mathtt{avail}}^{\mathtt{mobile\_init}} &= t_{\mathtt{pkflow}} - t_{\mathtt{ctd}} \\ &= \left(t_{\mathtt{fbu}} + \frac{1}{2}\mathrm{RTT}_{\mathtt{MN}} + \frac{3}{2}\mathrm{RTT}_{\mathtt{AR}} \right. \\ &\quad \left. + \frac{S+O}{b_{\mathtt{MN}}} + \frac{2(S+O)}{b_{\mathtt{ar}}}\right) \\ &\quad - \left(t_{\mathtt{ctar}} + \frac{1}{2}\mathrm{RTT}_{\mathtt{MN}} + \mathrm{RTT}_{\mathtt{AR}} \right. \\ &\quad \left. + \frac{S+O}{b_{\mathtt{MN}}} + \frac{S+O}{b_{\mathtt{ar}}} + \frac{s_{\mathtt{ctd}}+O}{b_{\mathtt{ar}}}\right) \\ &= t_{\mathtt{fbu}} - t_{\mathtt{ctar}} - \frac{s_{\mathtt{ctd}}-S}{b_{\mathtt{ar}}} + \frac{1}{2}\mathrm{RTT}_{\mathtt{AR}}, \end{aligned} \tag{11}$$

therefore the inequality $\Delta T_{\mathtt{avail}} \geqslant 0$ only holds if $t_{\mathtt{ctar}} \leqslant t_{\mathtt{fbu}} + \frac{s_{\mathtt{ctd}}-S}{b_{\mathtt{ar}}} - \frac{1}{2}\mathrm{RTT}_{\mathtt{AR}}$. This implies that, in order for the context to be timely available at the nAR, the CTAR message must be sent as soon as possible, and the context transfer must be completed before the tunnel is established between the two access routers.

In case of high mobility, the Mobile Initiated scenario shows a high $N_{\mathtt{default}}$ value.

It is worth noting that the tunnel setup is faster than the context transfer procedure and that the necessary time to establish a tunnel between the ARs could be saved by means of persistent connections.

In the access router initiated scenario, (Fig. 1(c)) we have:

$$\begin{aligned}\Delta T_{\mathtt{avail}}^{\mathtt{AR\_init}} &= t_{\mathtt{pkflow}} - t_{\mathtt{ctd}} \\ &= t_{\mathtt{fbu}} + \frac{1}{2}\mathrm{RTT}_{\mathtt{MN}} + \frac{3}{2}\mathrm{RTT}_{\mathtt{AR}} \\ &\quad + \frac{S+O}{b_{\mathtt{MN}}} + \frac{2(S+O)}{b_{\mathtt{ar}}} - t_{\mathtt{ctd}} \\ &= t_{\mathtt{fbu}} - t_x + \frac{1}{2}\mathrm{RTT}_{\mathtt{MN}} + \mathrm{RTT}_{\mathtt{AR}} \\ &\quad + \frac{2S - s_{\mathtt{ctd}} + O}{b_{\mathtt{ar}}} + \frac{(S+O)}{b_{\mathtt{MN}}}. \end{aligned} \tag{12}$$

The condition $\Delta T_{\mathtt{avail}} \geqslant 0$ is true only if the following inequality holds

$$\begin{aligned} t_x \leqslant\ & t_{\mathtt{fbu}} - \frac{1}{2}\mathrm{RTT}_{\mathtt{MN}} - \mathrm{RTT}_{\mathtt{AR}} \\ & - \frac{2S - s_{\mathtt{ctd}} + O}{b_{\mathtt{ar}}} - \frac{(S+O)}{b_{\mathtt{MN}}}. \end{aligned} \tag{13}$$

This relationship shows that in the access router initiated scenario, the context transfer procedure can be delayed to reduce the waste of bandwidth due to the necessity to send the context related information to all the candidate nARs, thus giving the possibility to the pAR to base the procedure on a more refined choice of candidates. A high delay in the context transfer procedure brings to a scenario that is very similar to the mobile initiated one, showing that tradeoff solutions could be considered between a high bandwidth waste for many anticipated context transfers that guarantee high handover performances, and a low bandwidth waste of a delayed context transfer scenario that could lead to handover performance degradation.

We now show some numerical examples to give a quantitative idea of our performance study. We assume $S = 300$ bytes, $O = 46$ bytes, $b_{\mathtt{MN}} = 1$ Mbps,

[1] In the case of vertical handoff [20] the RTT experienced by the MN communicating with the nAR may be different from that experimented communicating with the pAR.

$b_{\mathtt{ar}} = 2$ Mbyte/s, $\mathrm{RTT}_{\mathtt{MN}} = 0.001$ ms, $RTT_{\mathrm{AR}} = 1$ ms, and $S_{\mathtt{ctd}}$ ranges from 300 to 1500 bytes.

Eqs. (11) and (12) show that in the mobile initiated and in the access router initiated scenario, the value of $\Delta T_{\mathtt{avail}}$, besides depending on the context message size, overhead and other network parameters, also depends on the values of $t_{\mathtt{fbu}} - t_{\mathtt{ctar}}$ and $t_{\mathtt{fbu}} - t_x$ respectively. Both the values of these time intervals depend on a series of factors.

The mobile node sends the FBU message to bind itself to a new network. Chances to predict

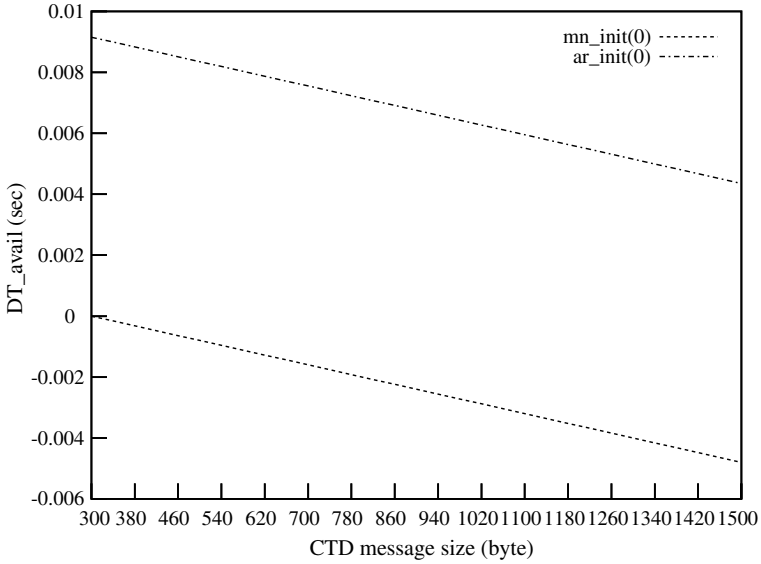

Fig. 7. $\Delta T_{\mathtt{avail}}$: case (A).

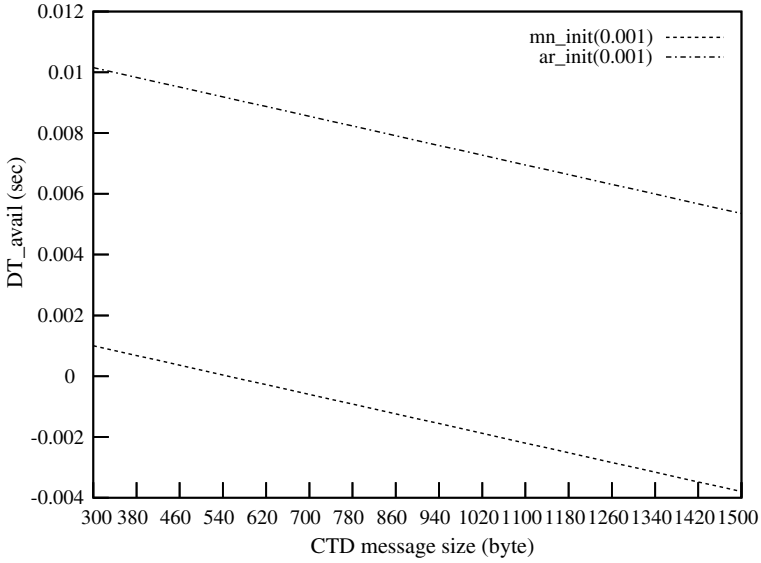

Fig. 8. $\Delta T_{\mathtt{avail}}$: case (B).

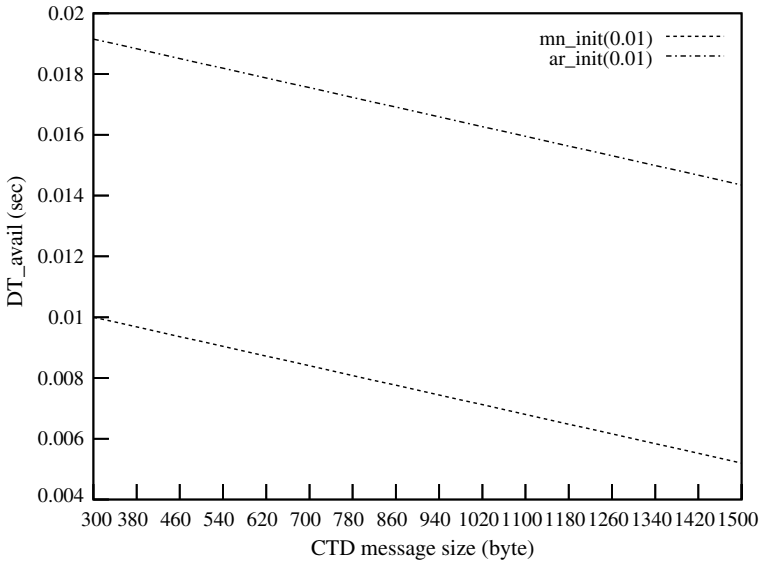

Fig. 9. $\Delta T_{\mathtt{avail}}$: case (C).

this action depend on the capability to process PrRtAdv messages and to predict the trajectory [23] and node speed. The CTAR message could be sent as the MN knows the addresses of the candidate nARs, choosing one or more of the addresses received in the PrRtAdv message. In the same way the pAR, in the AR initiate scenario, may broadcast a service context when it predicts that the user will leave the present AR coverage area. In our numerical example, we consider three cases: (A) $t_{\mathtt{fbu}} - t_{\mathtt{ctar}} = 0$ s and $t_{\mathtt{fbu}} - t_x = 0$ s, (B) $t_{\mathtt{fbu}} - t_{\mathtt{ctar}} = 0$ s and $t_{\mathtt{fbu}} - t_x = 0.001$ s, (C) $t_{\mathtt{fbu}} - t_{\mathtt{ctar}} = 0$ s and $t_{\mathtt{fbu}} - t_x = 0.01$ s, in the mobile initiated and in the access router initiated case respectively.

Figs. 7–9 show the numerical results.

In the considered cases, the AR initiated mechanism is the one that always guarantees no packets are erroneously treated by default, showing a positive value of $\Delta T_{\mathtt{avail}}$ for the considered values of $(t_{\mathtt{fbu}} - t_x)$. Fig. 7 shows that in case (A), if the mobile initiated triggering mechanism is in use, a certain amount of packets is erroneously processed by default, independently of the context data size.

In case (B), shown in Fig. 8, only if the context information can be encapsulated in small size packets ($s_{\mathtt{ctd}} < \sim 540$ bytes), the packets can be processed with the proper prioritization, otherwise, the packets are processed by default.

Fig. 9 shows that in case (C), the pAR has enough time to send the context, and the packet flow is processed with the proper QoS level even when the size the context data packets is reasonably high.

6. Conclusions and remarks

A considerable number of network services characterized by long lived sessions show a strong need for transparent procedures to transfer context information between network access points. The context transfer must be efficient to support low-latency and real-time applications.

In this paper we made a performance analysis of context transfer protocols, comparing three scenarios differentiated on the basis of the trigger mechanism in use to activate the context transfer procedures. Our analysis points out that, if the context data size is small, the mobile initiated procedure guarantees a good performance even when clients show high mobility. We also explain how predictive mechanisms reduce the cost of

handovers (in terms of number of lost packets and of packets processed by default), though requiring more bandwidth than dummy or mobile initiated solutions. Numerical results show the dependency of the amount of consumed bandwidth on the triggering mechanism. We show how the rate of packets that can be erroneously treated by default, i.e. without considering context related information during the time between the context transfer activation and its termination, is strictly dependent on the adopted trigger mechanism.

Acknowledgments

The work of Novella Bartolini has been funded by the WEB-MINDS project supported by the Italian MIUR under the FIRB program, and by the POLLENS project, supported by ITEA. The work of Emiliano Casalicchio has been funded by the PERF project supported by the Italian MIUR under the FIRB program.

References

[1] G. Agarwal, R. Shah, J. Walrand, Content distribution architecture using network layer anycast, in: Second IEEE Workshop on Internet Applications, San Jose, CA, July 2001.
[2] V.A. Almedia, D.A. Menasce, Capacity Planning for Web Performance: Models, Metrics, and Methods, Prentice Hall, 1998.
[3] N. Bartolini, E. Casalicchio, S. Tucci, Mobility-aware admission control in content delivery networks, in: Proceedings of IEEE/ACM MASCOTS, Orlando, Florida, October 2003.
[4] N. Bartolini, E. Casalicchio, S. Tucci, Performance Tools and Applications to Networked Systems, Lecture Notes in Computer Science, vol. 2965, Springer, 2004, Chapter: A Walk Through Content Delivery Networks.
[5] B. Carpenter, C. Huitema, Deprecating site local addresses, Internet Draft, November 2003.
[6] M. Gritter, D.R. Cheriton, An architecture for content routing support in the internet. Available from: <http://citeseer.ist.psu.edu/gritter01architecture.html>, pp. 37–48.
[7] R. Hinder, S. Deering, Ip version 6 addressing architecture, IETF Internet Draft, October 2003.
[8] R. Hinder, B. Haberman, Unique local ipv6 unicast addresses, Internet Draft, February 2004.
[9] J. Rosenberg et al., Sip: session initiation protocol, RFC 3261, June 2002.
[10] D. Johnson, C. Perkins, J. Arkko, Mobility support in ipv6, IETF Mobile IP Working Group Internet-Draft, May 2003.
[11] D. Katabi, J. Wroclawski. A framework for scalable global IP-anycast (GIA), in: SIGCOMM, 2000, pp. 3–15.
[12] J. Kempf, Instructions for seamoby and experimental mobility protocol iana allocations, IETF Internet Draft, June 2004.
[13] J.E. Kempf, Problem description reasons for performing context transfers between nodes in an ip access network, Network Working Group, RFC3374 September 2002.
[14] J. Loughney et al., Context transfer protocol, IETF Seamoby WG Internet-Draft, January 2004.
[15] P. McCann, Mobile ipv6 fast handovers for 802.11 networks, IETF Mobile IP Working Group Internet Draft, August 2004.
[16] T. Narten, R. Draves, Privacy extensions for stateless address autoconfiguration in ipv6, RFC 3041, January 2001.
[17] R. Hinden, S. Deering, Ipv6 addressing architecture, RFC2373, July 1998.
[18] R. Koodli, Fast handovers for mobile ipv6, IETF Mobile IP Working Group Internet-Draft, January 2004.
[19] M. Sayal, Y. Breithart, P. Scheuermann, R. Vingralek, Selection algorithms for replicated web servers, ACM SIGMETRICS Performance Evaluation Review 26 (3) (1998) 44–50.
[20] M. Stemm, R.H. Katz, Vertical handoffs in wireless overlay networks, Mobile Networks and Applications 3 (4) (1998) 335–350.
[21] T. Narten et al., Neighbor discovery for ip version 6 (ipv6), RFC 2461, December 1998.
[22] S. Thomson, T. Narten, Ipv6 stateless address autoconfiguration, RFC 2462, December 1998.
[23] T. Liu, P. Bahl, I. Chlamtac, Mobility modeling, location tracking, and trajectory prediction in wireless atm networks, IEEE Journal on Special Areas in Communications (Special Issue on Wireless Access Broadband Networks) 16 (6) (1998) 922–936.
[24] E. Zegura, M. Ammar, Z. Fei, S. Bhattacharjee, Application-layer anycasting: a server selection architecture and use in a replicated web service, ACM/IEEE Transaction on Networking 8 (4) (2000) 455–466.

Novella Bartolini graduated with honors in 1997 and received her Ph.D. in computer engineering in 2001 from the University of Rome, Italy. She is now assistant professor at the University of Rome. She was researcher at the Fondazione Ugo Bordoni in 1997, visiting scholar the University of Texas at Dallas in 1999–2000 and research assistant at the University of Rome 'Tor Vergata' in 2000–2002. Her research interests lie in the area of wireless mobile networks and content delivery systems.

Emiliano Casalicchio is researcher at the Computer Science Department, University of Roma "Tor Vergata". He graduated in Computer Engineering in 1998 and received a Ph.D. in Computer Science in 2002, from the University of Roma "Tor Vergata". His research interests are in the field of modeling and performance evaluation of Computer Networks, Distributed and Web systems. He is currently working also on Quality of Service topics in Grid Computing, and on modeling of complex systems and critical infrastructures.