SIP Paging and Tracking

Behcet Sarikaya, Xiao (Abbie) Zheng
Computer Science Department
University of Northern British Columbia
Prince George, BC V2N 4Z9 Canada
Email: (sarikaya, zhengx@unbc.ca)

Abstract—The paper introduces a new paging technique to track and wake up a mobile node attached to an access point in wireless LAN network after a SIP INVITE message is initiated by a caller. Tracking Agent keeps track of the mobiles’ handoffs between the access points. Paging Agent triggers tracking agent to page the mobile when a SIP INVITE is received for one of its users. Context transfer feature of our paging protocol allows the paging messages to deliver the station context in order to enable faster session reestablishment. Access point then does onlink paging in wireless link. SIP extensions are needed to trigger the paging agent to start paging and mobile nodes to notify their dormant status using an extended SIP REGISTER method. Onlink paging can be implemented using TIM or DTIMs of 802.11. Tracking protocol is analyzed to compare soft-state and hard-state approaches for state inconsistency ratio, message rate and the overall cost. Simulation model we developed enables us to evaluate the traffic introduced by the tracking protocol and the cache (state) size. Paging protocol is analyzed for CPU processing times and the transmission delays in SIP session setup with paging. Simulation of the paging with context transfer is used to show the gains in reauthentication.

Index Terms—Fluid flow and random walk mobility model, Paging Agent, Tracking Agent, Context Transfer, Onlink paging, Session initiation protocol.

I. INTRODUCTION

Location management which keeps track of the location of mobile nodes (MN) for delivery of information constitutes an important component of mobility management in wireless networks. Various strategies of location management have been developed to date to locate a dormant MN inside a wireless network when incoming calls arrive for the MN. Location management has two tasks: location tracking, which is initiated by MN, and paging, which is initiated by network entities such as base stations, access points (AP), and access routers (AR) [1].

Public cellular networks such as the third/fourth generation (3G/4G) Universal Mobile Telecommunications System (UMTS) networks handle dormant mode MNs effectively at the link layer, i.e. Layer 2 (L2). MN when dormant constantly monitors the paging channel and wakes up when it is paged. Location tracking is hierarchical, with cells, the base station’s coverage area, constituting the lowest level. Several cells inside a radio network controller is a Universal Terrestrial Radio Access Network registration area (URA) and several URAs constitute a registration area (RA). UMTS does location tracking of active MNs at URA and cell levels. Dormant mode MNs save battery by registering their location at RA level only when they move to another RA [2].

The Internet Protocol (IP) Paging (Layer 3 Paging) strategies exist in order to handle dormant MNs roaming in wireless LAN environments such as in Wi-Fi hotspots. A L3 paging area is defined as the coverage area of one access router. MN makes location registrations when it crosses L3 paging areas. Any traffic destined to MN triggers a paging
request to wake up MN within a paging area. Inside a L3 paging area there are several WLAN cells (L2 paging areas), coverage areas of APs. Location tracking of active MNs at the access routers can be done at cell level and this can be combined with IP Paging in order to efficiently locate MN in its most recent L2 paging area. Waking up the mobile node is done at the link layer using specialized beacons [3], [4].

The Session Initiation Protocol (SIP) is an application layer protocol that deals with the creation, modification and termination of multimedia sessions [5]. SIP is the protocol for Voice over IP (VoIP) call setup. SIP mobility has extensively been investigated [6], SIP and Mobile IP both can be used to deal with mobility issues [7], [8], [9]. However when they are used together, there could be a severe redundant registration overhead especially when mobile node frequently roams to new subnets. Besides, the indirect communications and tunnelling caused by MIP may also increase the end to end delay which is intolerable for VoIP applications. SIP protocol can be extended to support terminal mobility without MIP [10]. Even though a great deal of research has been done on SIP mobility, more remains to be done on how to deal with the issues arising from the mobile nodes in dormant mode roaming in Wi-Fi hotspots which we address in this paper.

We consider the following scenario. A mobile user connects his/her SIP-enabled phone to IEEE 802.11 wireless LAN network and receives a call from another mobile user connected to 3G/4G UMTS network. The callee is in dormant mode and therefore the call is missed. The solution is to do the location tracking using SIP Location Tracking and then alert/ wake-up the callee using SIP Paging.

The paper first introduces SIP Paging and Location Tracking methodology in Section 2. Sections 3 and 4 discuss the tracking and paging protocols, respectively. Section 5 and 6 presents an analysis and simulation of SIP tracking and paging protocols, respectively. Finally Section 7 concludes the paper.

II. SIP PAGING AND LOCATION TRACKING

Session initiation in VoIP starts with the caller sending the INVITE request to the SIP server (Proxy) in the domain for the callee. The caller and callee are identified at the application layer using Uniform Resource Locator/ Identifier (URL/URI) which has a form similar to an email address such as sip:bob@biloxi.com. The establishment succeeds with 200 OK reply transaction issued by the callee reaching the caller providing IP addresses which are used to transmit multimedia streams. The call is terminated by either side sending the BYE request followed by 200 OK reply.

As an example, Bob’s computer with an IPv6 address identified with SIP URI of sip:bob@2001:410:1000:1:3656:78FF:FE9A:BCDE is used to represent location. SIP registration establishes a binding between an address-of-record (AOR), e.g. sip:bob@unbc.ca and a contact address in IPv4/v6. SIP User Agent (UA) sends its registration message to the home registrar, e.g. sip:unbc.ca.

SIP Paging is triggered after a SIP INVITE message is received for a dormant MN connected to a WLAN access point (AP) by the proxy server. If the proxy server forwards this datagram to the access router of WLAN network, AR needs to perform a neighbor discovery that involves all APs under the AR. APs notify all MNs using the delivery traffic indication map (DTIM) beacons. However, MNs in power save mode usually do not receive such beacons and thus do not participate in the neighbor discovery and as a result do not receive the INVITE message [4]. Because of this, proxy server has to start a paging procedure which will wake up the MN and deliver the INVITE message. We call such a protocol SIP Paging protocol in this paper.

SIP Paging and Location Tracking architecture contains several entities: a SIP proxy server, a Location Server (colocated with Registrar) (LS), a Paging Agent (PA) and a Tracking Agent (TA) (Fig. 1). MN is identified by LS through SIP registration with its MAC address and its IP address which might have changed. LS asks PA to page the MN. PA starts the paging procedure by sending the Paging Request message to the tracking agent (TA). TA finds the corresponding entry in the paging cache by the matching MAC address and replaces it with MN’s IP address. SIP Paging and Location Tracking system is composed of two protocols, the tracking component called SIP Tracking (discussed in Sec. 3) and the Paging component called SIP Paging (discussed in Sec. 4).

SIP Paging is hierarchical. At the top level is the SIP Proxy with its PA/TA. The next level is ARs
with each AR in charge of its Layer 3 paging area. The lowest level is APs with each AP in charge of its WLAN cell. TA sends the Paging Request message to the AR of the most recent L3 paging area. AR starts the paging with the AP to which MN was most recently associated with. Paging request/reply messages are UDP messages. Hierarchical paging requires the MNs in dormant mode to be tracked at the routing area level, to know MN's latest Layer 3 and Layer 2 attachment points, i.e. the most recent AR that serves MN and the most recent access point (AP) MN has associated with. This requires MN to register its dormant mode status which is possible by SIP dormant mode registration.

TAs can be replicated at each AR. Such an approach would reduce tracking message delays as TA is closer to MNs but the cost increases. In Section V, we present an analysis of the centralized TA versus replicated TAs.

A. SIP Dormant Mode Registration

MN's going into the dormant mode need to signal this to SIP Paging and Location Tracking system. For this purpose, we extend the SIP protocol by adding dormant mode registration messages to support paging. SIP REGISTER Request message is extended to indicate dormant mode registration with five new contact header parameters - mode, AR, MAC, ReceiveDTIMs and idletime.

When MN detects prolonged inactivity, it sends a dormant mode REGISTER request message to the SIP Proxy. SIP Proxy responds with a dormant mode response message (200 OK) and updates its location service database. When the idle time expires, the entry in the location service database will not be deleted, but the value of mode will be changed to active. The idle time of MN can be extended by another dormant mode REGISTER request message.

The other three new contact header parameters are AR, MAC, ReceiveDTIMs. AR field contains the IP address of AR that MN is attached, MAC field has MN's MAC address and ReceiveDTIMs field indicates whether its receiveDTIMs bit is set.

The extension header fields are used as follows: When mode field is set to idle, SIP Proxy then knows that the MN is in dormant mode and the expected duration is taken from idletime header field. The other fields are sent in PAGE request message to PA and are used by Paging and Tracking protocols. SIP Proxy keeps track of the last two values of AR in new AR and old AR and sends both values to PA. PA uses AR information of MN to perform paging context transfer to reduce the paging delay. We discuss the paging context transfer in Section IV-A. MAC header field is used for mapping MN's MAC address to IP address by TA. TA passes on receiveDTIMs to the access point which uses this information to determine the method to perform on-link paging. AP uses DTIM beacons if receiveDTIMs is set otherwise AP uses TIM beacons.

When MN moves to another subnet in dormant mode, it must send a normal registration message and a dormant mode registration to SIP Proxy. An example dormant mode REGISTER request/response exchange is shown below:

```
Register sip:registrar.unbc.ca SIP/2.0
From:Alice <Alice@unbc.ca>;tag=456248
To:Alice <Alice@unbc.ca>
Call-ID:123456@998sdasdh09
Cseq: 1 Register
Contact: <sip:Alice@192.0.2.4>;mode=idle;
AR=192.0.2.1;ReceiveDTIMs=true;idletime=300;
MAC=00-11-11-80-38-36
Expires: 3600
Content-Length:0
```

```
SIP/2.0 200 OK
From:Alice <Alice@unbc.ca>;tag=456248
To:Alice <Alice@unbc.ca>;tag=2493k59kd
Call-ID:123456@998sdasdh09
Cseq: 1 Register
Contact: <sip:Alice@192.0.2.4>;mode=idle;
AR=192.0.2.1;ReceiveDTIMs=true;idletime=300;
MAC=00-11-11-80-38-36
Expires: 3600
Content-Length:0
```

B. SIP PAGE Request Message

We extend SIP protocol with a new PAGE method. PAGE method is used to initiate paging. SIP Proxy server sends this message to the Paging Agent (PA) in the administrative domain. PA is implemented in a SIP User Agent Client (UAC) module in order to process SIP methods like PAGE. When a PAGE request is received, PA starts paging
of the mobile nodes whose MAC addresses are given as parameters to PAGE. PA’s task is to map this request down to the IP layer by communicating with the tracking agent.

Fig. 2 shows the implementation of PA. A centralized TA can be collocated with PA as shown in the figure. PA operates at SIP level. It receives a PAGE request from a SIP server and then it uses the collocated TA to consult its paging cache and to initiate a paging on the wireless link at the Access Point (AP) to which the paged dormant mobile is presently associated. At the end of paging, PA replies with a 200 OK message.

The header fields in a PAGE method are MAC address of the mobile node, the old and new AR IP addresses and ReceiveDTIMs for one or more mobile nodes to be paged. PA will compare the old and new AR IP addresses. If they are different, context transfer will be triggered (we will discuss the context transfer with paging in Sec. IV-A in detail).

III. SIP TRACKING

IEEE has defined the inter access point protocol (IAPP) for station (mobile node) handover signaling and station security context exchange [11]. SIP Tracking uses IAPP handover messages but extends the operation from APs to ARs and TA. The messages are transported using UDP. IAPP requires APs to send Add-mobile message to all other APs when MN associates with the AP and to send Add-notify to the old AP when MN reassociates with the AP.

MNs roam in WLAN cells associating/reassociating with APs. SIP Tracking protocol is used to establish L2 location information called paging cache at the Tracking Agent (TA) which is collocated with a SIP server. ARs discover the TA in the domain using a discovery protocol. SIP Tracking operation is shown in Fig. 3.

A new entry to the paging cache is added when MN associates/reassociates with an AP. The paging cache contains MAC and IP addresses of MN and AP, a total of 20 bytes per MN in IPv4. ADD-notify interaction of SIP Tracking is sent by multicast by an AP when the mobile associates/reassociates with it. ADD-notify packet contains MAC address of the mobile in its data field. AR after receiving the ADD-notify extracts AP’s IP address and mobile node’s MAC address and sends an ADD-mobile UDP packet to TA.

TA gets MAC address and AP IP address fields from ADD-mobile messages received from AR. Note that at the time of an IEEE 802.11 association, the IP address of the mobile is not known. The mobile creates an IP address after it receives router advertisement from AR and then goes through duplicate address detection in IPv6 or makes a stateful address configuration with DHCP server. AR knows IP address of the mobile and sends ADD-mobile message to TA to update PC with MAC and IP addresses. After this message, TA can have the complete location information for the mobile.

In soft-state (SS) tracking protocol, the paging cache is soft-state, i.e. the entries expire and are removed if not refreshed after timeout. The entries may be updated due to mobility of MN reassociating with a new AP. The new AP then sends a MOVE-notify UDP packet to the old AP through AR which
in turn updates TA with ADD-mobile (Fig. 3). AR may refresh entries by periodically sending ADD-mobile messages for all MNs for which AR has a neighbor cache entry. When MN leaves the WLAN network, its paging cache entry will remain until the entry times out.

In hard-state tracking protocol (HS), a reliable signaling protocol such as TCP is used to establish state at TA. When MN leaves the WLAN domain, the state is removed explicitly using Remove-Mobile message of hard-state SIP Tracking protocol.

### IV. SIP Paging

PA starts paging procedure by sending the Paging Request message to TA. Paging Request message contains MAC address, AR address, ReceiveDTIMs and optionally a context data block fields for one or more MNs. TA finds the corresponding entry in the paging cache by the matching MAC address and replaces it with MN’s IP address then sends a Paging Request message to the AR which in turn sends it to the AP (Fig. 1).

After receiving Paging Request, AP starts on-link paging to wake up the node. On-link paging is an extension of IEEE 802.11 power save mode procedures in infrastructure networks. AP uses TIM or DTIM beacons depending on the settings of MN. For this purpose it uses ReceiveDTIMs field of Paging Request message (if set DTIM otherwise TIM beacons are used). MN wakes up and checks to see if its address is in the data frame and if it is, then it goes to the active mode in Layer 3.

AP sends a Paging Reply message back to TA and TA sends it to PA. After the reply is received, LS sends the URI and IP address to the SIP server and then SIP server sends the INVITE message to MN. If Paging Reply is negative, TA sends Paging Request message to the other APs in the same subnet and then to APs in the other subnets, i.e. TA is in charge of the whole domain.

If no positive reply comes from MN within a timeout period, AR will send Paging Request message to the other APs in the distribution system in a round robin fashion. If MN cannot still not be located under this AR, TA will page the other ARs in the domain. If no positive Paging Reply message is received, LS returns no URIs and therefore the SIP server is required to go to the other administrative domains [12].

A. Context Transfer with Paging

After waking up, mobile node may need to authenticate itself and perform layer 3 registrations before VoIP application could be resumed. This is a time consuming process. If AAA/Mobile IP authentication is used, a full authentication could take about 52.5 ms [13]. If MN has already authenticated itself in another subnet of the same administrative domain, we could transfer the MN’s context from the old AR to the new AR to reduce this delay. Context information is transferred in Paging Request messages extended with a context data block field.

When MN goes to dormant mode, a SIP dormant mode REGISTER request message is sent to the SIP Proxy. SIP Proxy gets the information of the AR that MN is attached from the AR extension header field of the dormant mode REGISTER request message as described in Section II-A. If MN moves to another AR in dormant mode, a new SIP dormant mode REGISTER request message will need to be sent to SIP Proxy. Context transfer will not happen at handoff time because MN is still in dormant mode and it could move to another subnet before waking up. SIP Proxy keeps track of the new AR and the previous (old) AR and sends both AR addresses to PA in PAGE request message.

When PA receives a PAGE Request, it compares the new and old AR values. If they are different, PA will start the context transfer by sending a context data request message to the old AR. Old AR responds with the context data message for this
MN as in IETF’s Context Transfer Protocol [14]. PA then adds the context data field to Paging Request message sent to TA (Fig. 4).

TA processes Paging Request message fields for each MN to be paged. It sends a Paging Request message for each different new AR under which there are MNs to be paged. TA includes the context data into the paging request and sends it to the new AR. New AR will extract and install the context and then send the context in a paging request to the AP that MN is attached.

V. SIP TRACKING PROTOCOL ANALYSIS

Tracking protocol is analyzed and some simulation results are presented.

A. Soft-State Model

A continuous time Markov chain model of various signaling protocols (soft-state, soft-state with explicit removal or reliable trigger messages, reliable trigger/removal messages and hard-state) is developed in [15] which we adopt in this paper.

AR in Fig. 3 is the signaling sender that installs/updates the state at TA, the signaling receiver and starts a refresh timer T of 2s, the same value as the neighbor cache refresh timer. TA, after receiving state installation/refresh message enters/updates the paging cache entry and starts a soft-state timeout timer X of 90s. The entry is removed after X times out because AR did not update it or because AR’s message has been lost which is called false removal.

Markov chain starts at state 0 which is also its absorbing state which models the handover occurring resulting in an ADD/MOVE-notify message sent to AR (Fig. 5). State 5 occurs when AR removed MN from its neighbor cache but MN is still in the paging cache at TA. State 2 is when both AR and TA states are consistent. The chain transitions to state 1 from state 0 if ADD-mobile message is lost. AR will refresh the state which is represented by a transition from 1 to 2. The chain transitions to state 3 from 2 when AR sends an ADD-mobile message to update (refresh) TA. The chain transitions to state 4 when the update/refresh message is lost. MN stays in the system for a mean session interval of 1 hour after which its state is removed at AR. This is modeled as a transition from state 1 to 0. The transitions from states 2 and 4 to 5 occur when TA has already installed state for the MN. Eventually this state will be removed which is shown by a transition from state 5 to state 0. Fig. 5 has transitions that model the false removal from state 2 and 4 to state 1.

The probability of loss for AR to TA communication is much lower than for MN to AP communication over 802.11 links. Since state in AR will be triggered by MN’s L3 activity, we will use 802.11 link loss rate as the probability of loss. Loss rates exceeding 20% have been reported in the literature [16], [17], so we will use 0.15 as mean value since loss rates are much lower in the links between AR and TA. Signaling state update interval depends on the mobility pattern of MN. We will use the time a pedestrian takes to cross 30m WLAN cell boundary to define the signaling state update interval \( \left( \frac{1}{\lambda_u} \right) \). This value is set to 60s for a slow move user and 5s for a fast move user (together with
The false state removal rate at receiver-soft-state $\lambda_f$ is the probability that the signaling messages are lost during the period of the soft-state timeout timer value and is calculated by:

$$P_{\lfloor X/T \rfloor}/X$$

which yields 0.0005625 for fast move and 0.000225 for slow move users.

For D, the signaling channel delay, the number of hops between AR and TA and association delay of an MN with its AP are the main contributing factors. Association delay involves the phases of probing, authentication and reassociation each involves the exchange of several frames between MN and AP and the total delay values between 100 and 400 ms are reported in [16]. Therefore the association delay will be the dominant factor for D and we use 200ms as a typical value.

B. Hard-State Model

As a hard-state signaling protocol, SIP Tracking sends ADD-mobile messages using TCP. Since TCP guarantees message delivery, there is no need to refresh the state established at the signaling receiver, the TA. Also there is no need to keep soft-state timeout timer at TA. However, a new message called Remove-mobile is needed for explicit state removal at TA which is sent when AR no longer has MN in its neighbor cache. Remove-mobile messages can not be triggered by MN disassociating from AP because IEEE 802.11 does not require MN to send a disassociate request message.

Markov model of the hard-state SIP tracking is shown in Fig. 6. A new state, state 6 is used to model the loss of Remove-mobile messages. We used 0.0001 for the false state removal rate, $\lambda_w$ at the hard-state receiver from [15].

C. Results Of Analysis

We obtained steady-state transition probabilities based on the configuration parameters given in Table I for both soft-state and hard-state Markov chains. We will compare soft- and hard-state approaches for inconsistency ratio, signaling rate and the cost of signaling overhead.

Inconsistency ratio is the fraction of time the system spends outside of state 2 where both TA and AR have consistent values. Fig. 7 shows the inconsistency ratio as a function of the session length.

Average signaling message rate is the total number of messages required during a signaling session normalized over the session length. Fig. 8 shows it versus sender’s mean signaling lifetime. Both of Figs. 7 and 8 show that with increasing session length, inconsistency ratio and average message rate decrease. This means that WLAN users staying active for short periods of time induce less tracking messages to be exchanged and also there is less likelihood of state inconsistency at AR and TA. The figures also show that fast moving users incur higher message rates in both soft- and hard-state protocols. Hard-state protocol has smaller message rates than soft-state protocol and inconsistency ratio is smaller for hard-state than soft-state protocol.
The cost of signaling is the sum of the signaling message cost and the cost arising from the state inconsistency. State inconsistency in tracking results in increased paging messages in order to correctly locate MN, i.e., it increases paging latency. We define paging latency as the number of extra signaling messages caused by state inconsistency, i.e., the paging request messages from AR to APs and the paging reply messages from APs to AR. In case of inconsistency, TA will page all the APs below an AR, thus assuming 5 APs per AR, the latency will increase to 10. We use this value to factor the inconsistency ratio in the overall cost formula.

Since we measure the cost versus T, soft-state refresh timer, the cost is fixed for hard-state signaling. The cost versus T is shown in Fig. 9. Soft-state overall cost values for slow move users decrease and approach to the fixed hard-state value up to some values of T after which the cost starts to increase. On the other hand, using a low value for T will increase the message signaling cost. The same observation can be made for fast move users. From Fig. 9, we can find optimal values to use for T, between 2 to 7 which are the values we used. For a loss rate $P_l$ of 0.2 the optimal value range for T is between 2 and 5. From this we conclude that lower loss rate enables longer optimal refresh timer values.

Fig. 10 shows the effects of loss rate on inconsistency ratio. Hard-state protocol has very low inconsistency ratio even at high loss rates due to the reduced signaling. For fast move users inconsistency ratio increases for both hard-state and soft-state protocols but hard-state ratios are much lower than the soft-state rate.
D. Effect of TA Placement on Tracking Performance

Tracking Agent could be replicated at each Access Router instead of a single centralized TA. Markov chain models will stay the same as described above whether TA is at SIP Proxy or at each AR but some of the parameters used in solving various performance measures will differ. If TA is placed at each AR, the signaling life time will be shorter. If there are four ARs in a subnet, the signaling life time for TA at each AR will be \(1/4\) of the signaling life time where TA is at SIP Proxy. We used \(\frac{1}{\lambda_d} = 900\) s for TA at each AR and obtained the results for the inconsistency ratio, signalling life time and the overall cost.

We can see from Fig. 7 and Fig. 8 that both inconsistency ratio and signaling message rate decrease as the signaling lifetime increases. So the centralized TA architecture will have lower inconsistency ratio and lower signaling message rate. We validated this with the analysis results, the resulting figures are omitted to save space.

After the analysis with the parameters described above, we also find that with the same soft state refresh time value, the overall cost is slightly lower when TA is at Proxy. With the same loss rate, the inconsistency ratio for TA at Proxy is slightly lower too. So we conclude that the centralized TA architecture will have a better performance than the replicated TA architecture.

E. Simulation Results

We developed a simulation model of the tracking protocol using OPNET in order to study the effects of MN mobility (speed) on the paging cache size and the traffic introduced by the protocol. Fig. 11 shows the simulation topology.

We placed 10 MNs under each of 4 APs initially (AP2-AP5 in Fig. 11). MNs start to move and change directions randomly every 30s. For the soft-state protocol, the refresh interval is set to 2s and cache expiry time is set to 6s. We measured the paging cache size at the TA as a function of the speed of MN and the results are shown in Table II. We also measured the tracking traffic rate in bits/s. Each update packet is 75 bytes long including MAC and IP headers. This would yield an average traffic rate of \(75*40/2\) or 12 kbps which was validated using simulation. Simulation results indicate that the tracking protocol is lightweight because its state size is \(O(n)\) where \(n\) is the number of MNs and the traffic introduced is almost negligible.

We also developed a simulation model to verify inconsistency ratio analysis results in Fig. 7. The simulation topology is the same as in Fig. 11, except that only one MN is in the domain at the right hand side. The MN moves between AP2 and AP5. The channel loss rate is set to 0.2 and the channel delay is set to 200ms as in Table I. For the fast move scenario, the time for MN to move from AP2 to AP5 is 15 seconds. The refresh interval is 2s and cache expiry time is 6s. For the slow move scenario, the time for MN to move from AP2 to AP5 is 3 minutes. The refresh interval is 5s and cache expiry time is 15s.

The results of the inconsistency ratio simulation is shown in Fig. 12. We can see that when the signaling time is short, the inconsistency ratio for the slow move is much lower than the theoretical analysis in Fig. 7. That is because in our slow move model, the first handoff happens after 60 seconds. As the signaling mean time increases, the simulation
results get closer to the analysis results.

VI. ANALYSIS OF PAGING

We analytically derive the update and refresh processing loads for SIP paging induced at SIP servers. We compare the processing loads with another paging system called Mobile IPv6 Hierarchical Paging (MIPv6HP). Hierarchical MIPv6 (HMIPv6) is an extension of Mobile IPv6 to support intra domain mobility [18]. MIPv6 Hierarchical Paging protocol (MIPv6HP) extends HMIPv6 with paging. Subnets in a network are divided into different paging areas. When a mobile node moves into a new subnet in the same network, if the old subnet and new subnet are in the same paging area, MN doesn’t perform a layer 3 registration; otherwise it registers with the server called Mobility Anchor Point (MAP) [4].

Deterministic fluid flow and random walk mobility models are used to model user (mobile node) mobility in WLAN cells [19]. For both models, we assume an hexagonal cellular network architecture. The coverage area of an AP is an hexagonal cell with a perimeter of $L_B$ and area of $\frac{3L_B^2}{\pi}$. There are $B_D$ APs and $R_D$ ARs in the network.

For the deterministic fluid model, the rate of mobile hosts crossing a boundary of perimeter $l$ at a speed $v$ is $R(l) = \rho vl/\pi$, when the user movement is uniformly distributed over $[0, 2\pi]$. The parameter $\rho$ represents the active user density.

The rate of layer 2 handoffs is $R(l)B_D$. So the processing load for Add-Mobile messages is $P_{TAA} \frac{\rho v L_B B_D}{\pi}$, where $P_{TAA}$ is CPU processing time for Add-Mobile messages.

The number of refresh messages at TA is $N$, where $N$ is the number of users and $T$ is the refresh interval for soft state protocol. So the processing load for refresh messages is $P_{TAR} \frac{\sqrt{3\rho L_B^2}}{24T}$, where $P_{TAR}$ is CPU processing time for refresh messages.

Similarly, the processing load for SIP REGISTER request messages is $P_{ProxyU} \frac{\rho v L_B \sqrt{B_D R_D}}{\pi}$, where $P_{ProxyU}$ is CPU processing time for SIP REGISTER messages. The processing load for refresh messages for SIP REGISTER requests is $P_{ProxyR} \frac{\sqrt{3\rho L_B^2}}{24TSIP}$, where $P_{ProxyR}$ is CPU processing time for the refresh messages and $T_{SIP}$ is the refresh interval.

$CPU_{TA}$ or $CPU_{Proxy}$ in fluid flow model can be expressed as: $CPU_{Proxy} = P_{TAA} \frac{\rho v L_B B_D}{\pi} + P_{TAR} \frac{\sqrt{3\rho L_B^2}}{24T} + P_{ProxyU} \frac{\rho v L_B \sqrt{B_D R_D}}{\pi} + P_{ProxyR} \frac{\sqrt{3\rho L_B^2}}{24TSIP}$

If TA is at AR as in MIPv6HP, the CPU processing load will be divided into TA part at AR and PA part at SIP Proxy, where

$CPU_{TA} = P_{TAA} \frac{\rho v L_B B_D}{\pi} + P_{TAR} \frac{\sqrt{3\rho L_B^2}}{24T}$

$CPU_{PA} = P_{ProxyU} \frac{\rho v L_B \sqrt{B_D R_D}}{\pi} + P_{ProxyR} \frac{\sqrt{3\rho L_B^2}}{24TSIP}$

As for MIPv6HP, we can derive CPU processing time at the MAP similarly as in $CPU_{TA}$:

$CPU_{MAP} = P_{TAA} \frac{\rho v L_B B_D}{\pi} + P_{TAR} \frac{\sqrt{3\rho L_B^2}}{24T} + P_{MAPU} \frac{(1-\gamma)\rho v L_B \sqrt{B_D R_D}}{\pi} + P_{MAPR} \frac{\sqrt{3\rho L_B^2}}{24TSIP}$

where $P_{MAPU}$ and $P_{MAPR}$ are MAP’s CPU processing time for registration messages and refresh messages respectively.

Table III shows the configuration parameters and their values we use in plotting $CPU_{MAP}$ and
TABLE III
CONFIGURATION PARAMETERS FOR PROCESSING LOAD ANALYSIS

<table>
<thead>
<tr>
<th>Item</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ρ</td>
<td>1000/km²</td>
</tr>
<tr>
<td>L_P</td>
<td>0.24km</td>
</tr>
<tr>
<td>R_P</td>
<td>20</td>
</tr>
<tr>
<td>P_TA</td>
<td>0.2msec</td>
</tr>
<tr>
<td>P_TA</td>
<td>0.2msec</td>
</tr>
<tr>
<td>P_ProxyU</td>
<td>0.15sec</td>
</tr>
<tr>
<td>P_ProxyR</td>
<td>0.015sec</td>
</tr>
<tr>
<td>T</td>
<td>2sec</td>
</tr>
<tr>
<td>T_SIP</td>
<td>300sec</td>
</tr>
<tr>
<td>T_MAP</td>
<td>300sec</td>
</tr>
<tr>
<td>P_MAPU</td>
<td>0.5sec</td>
</tr>
<tr>
<td>P_MAPR</td>
<td>0.05sec</td>
</tr>
<tr>
<td>γ</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Fig. 14. Random Walk Mobility Model Architecture

$CPU_{Proxy}$. SIP REGISTER request message processing time is set to 150ms [20]. Numerical CPU processing times in seconds for SIP paging and MIPv6HP are plotted in Fig. 13. In the figure, the number of APs and velocity are represented with a tuple (AP, v) as one of the coordinates. Fig. 13 shows that when v and the number of APs increase, the CPU processing time increases slower in SIP paging than in HMIPv6 hierarchical paging. This is due to the smaller processing time value used for SIP REGISTER request message.

If an MN is located in a cell of r ($r > 0$), the probability that a movement will result in an increase in the distance from the center cell will be $\frac{1}{3} + \frac{1}{6r}$, and the probability that a movement will result in a decrease in the distance from the center cell will be $\frac{1}{3} - \frac{1}{6r}$. The transition probabilities $\alpha_{r,r+1}$ and $\beta_{r,r-1}$ represent the probabilities of distance of MN from the center cell increasing or decreasing.

$$\alpha_{r,r+1} = \begin{cases} (1 - p) \frac{1}{3} & \text{if } r = 0 \\ (1 - p)(\frac{1}{3} + \frac{1}{6r}) & \text{if } 1 \leq r \leq R \\ \end{cases}$$

$$\beta_{r,r-1} = (1 - p)\left(\frac{1}{3} - \frac{1}{6r}\right) \text{ if } 1 \leq r \leq R$$

Let $\pi_{r,R}$ be the steady state probability of state r within a subnet consisting of R rings.

$$\pi_{0,R} = \frac{1}{1 + \sum_{r=1}^{R-1} \prod_{i=0}^{r-1} \frac{\alpha_{i,i+1}}{\beta_{i+1,i}}}$$

$$\pi_{r,R} = \pi_{0,R} \prod_{i=0}^{r-1} \frac{\alpha_{i,i+1}}{\beta_{i+1,i}} \text{ for } 1 \leq r \leq R$$

So the probability that an MN performs a SIP registration is $\pi_{R,R} \alpha_{R,R+1}$. $CPU_{TA}$ in the random walk mobility model can be expressed as:

$$CPU_{proxy} = \frac{\pi_{R,R+1} P_{proxyU} + (1 - \pi_{R,R+1}) P_{TA}}{T}$$

where $T$ is the average residence time of MN staying in a cell.

Fig. 15 shows the variation in the CPU processing time as the average cell residence time is changed in the random walk model. Two values for R, 1 and 4 are considered and p is varied from 0.2 to 0.6.

From the figure, we observe that CPU processing time decreases as the average residence time increases. This is due to the fact that MN has lower mobility as the average residence time increases. When R is larger, each AR will be in charge of more APs. When a MN moves from one AP cell to another, the probability of it still staying under the same AR is higher. So, as Fig. 15 shows, the frequency of SIP registration will be lower, and the CPU processing time will also be less.

A. Paging Delay Analysis

We will analyze transmission delays in SIP paging. Processing delays and queueing delays will not be considered. SIP signaling transmission delay analysis on wireless links was presented for UDP,
TCP and RTP transport in [22]. We apply the UDP transport analysis of [22] to SIP Paging.

When SIP Proxy receives a SIP INVITE message for a MN in dormant mode, a paging request message is sent downstream to AP. AP will do the link layer paging and MN will send a paging reply back to AP. AP forwards the paging reply packet to SIP Proxy. Then SIP proxy knows that MN woke up and sends the SIP INVITE message to it. MN sends a 200 OK message back and SIP proxy acknowledges it with an ACK message. All the SIP messages are sent in UDP packets. We assume that the wired propagation and link delay are 0.5ms and the wired link bandwidth is 100Mb/s.

DTIM is used for link layer paging and beacon period is 3 and beacon interval is 100ms. With these values, link layer paging takes 300 ms. Fig. 17 shows the delay in SIP session setup with paging, D in seconds evaluated at various loss rates between

\[ Tr(i) = 2^{i-1} Tr(1) \]

Let us consider the delivery of the messages such as Paging Request/Reply, INVITE, etc. in Fig. 16. the original retransmission timer, \( Tr(1) \) will be \( Tr(1) = D_{message} + D_{reply} \), where \( D_{message} \) is the message transmission delay (Paging Request), \( D_{reply} \) is the reply message transmission delay (Paging Reply).

Let Nm be the maximum number of transmissions, i.e. the total of the first transmission and any subsequent application level retransmissions. The normalized delay \( T_{message} \) for the successful transmission of the SIP message is as follows:

\[
T_{message} = \frac{1}{1-q^{Nm}} [(1-q)(D_{message}) + (1-q)q(Tr(1) + D_{message}) + ... + (1-q)q^{Nm}(2^{Nm-1}Tr(1) + D_{message})] \\
= D_{message} - Tr(1) + \frac{(1-q)(1-(2q)^{Nm})}{(1-q^{Nm})(1-2q)} Tr(1)
\]

where \( q \) is the probability of a transaction having failed, which means either the message or the reply is lost. \( q \) can be expressed as a function of the loss rate as \( 1 - (1-P_l)^2 \).

The total delay is

\[ D = \sum_{i=1}^{N} T_{message}(i) \]

where \( N \) is the number of messages necessary for SIP session setup with paging.

We consider IEEE 802.11b which provides a wireless channel of 11Mbps for numerical analysis. The parameters are given in Table IV. The values are derived from [22], [23]. We set Nm to 5 and N to 4.

We assume the wired propagation and link delay are 0.5ms and the wired link bandwidth is 100Mb/s. DTIM is used for link layer paging and beacon period is 3 and beacon interval is 100ms. With these values, link layer paging takes 300 ms. Fig. 17 shows the delay in SIP session setup with paging, D in seconds evaluated at various loss rates between

<table>
<thead>
<tr>
<th>Messages</th>
<th>Message size(bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>paging request</td>
<td>140</td>
</tr>
<tr>
<td>paging response</td>
<td>56</td>
</tr>
<tr>
<td>SIP INVITE</td>
<td>728</td>
</tr>
<tr>
<td>200OK</td>
<td>898</td>
</tr>
<tr>
<td>ACK</td>
<td>428</td>
</tr>
</tbody>
</table>

TCP and RTP transport in [22]. We apply the UDP transport analysis of [22] to SIP Paging.

When SIP Proxy receives a SIP INVITE message for a MN in dormant mode, a paging request message is sent downstream to AP. AP will do the link layer paging and MN will send a paging reply back to AP. AP forwards the paging reply packet to SIP Proxy. Then SIP proxy knows that MN woke up and sends the SIP INVITE message to it. MN sends a 200 OK message back and SIP proxy acknowledges it with an ACK message. All the SIP messages are sent in UDP packets. We assume that the wired link is lossless. Let \( p \) be the probability of loss in the wireless link. Fig. 16 shows the interactions involved in SIP session setup after SIP Paging.

For packets involved in SIP session setup after SIP Paging, the back off timer after the ith transmission, \( Tr(i) \) doubles after each retransmission. So
B. Analysis of Context Transfer with Paging

We developed a simulation model of the paging protocol using OPNET to study the effects of context transfer. The simulation topology is the same as Fig. 11 except that only one MN is in the domain on the right side.

In our simulation, the mobile node will move from AP3 under AR2 to AP4 under AR3. Two scenarios are simulated here. In scenario 1, handoff happens when mobile node is in dormant mode. Context transfer will not be used. Authentication of MN will take 52.5ms.

In scenario 2, handoff happens when MN is in dormant mode too. We will transfer the context with paging messages as described above. In both scenarios, a SIP INVITE message will be sent for the dormant MN to trigger paging after the handoff.

We use PANA context as the context transferred in scenario 2. The elements in the PANA context are listed in the Appendix [24]. The length of context is 76+sizeof (ISP-Name) bytes. In our simulation, the context size is set to 100 bytes. We assume DTIM beacons are used in layer 2 paging. The beacon interval is 3 and beacon period is set to 10ms. The resulting paging response times are shown in Table V. We can see that context transfer is efficient to reduce the paging response time.

VII. CONCLUSIONS

We introduced a new methodology called SIP Paging and Location Tracking to track L3/L2 location of the mobile VoIP nodes in a WLAN domain and then page and wake up the node when a call comes. SIP Paging is done with a Paging Agent component to alert a dormant mobile node after a SIP INVITE message is received in the domain. SIP Location Tracking is done with a Tracking Agent at the SIP layer which receives location updates from the access routers based on associations made to the access points. Paging is triggered by PA and TA sends the paging request to the access point to which the mobile was associated last. The access point uses on-link paging to wake up the mobile. Mobile nodes are required to register their dormant mode status with SIP Proxy server using the extensions we introduced. PA communicates with SIP Proxy using a new SIP message we introduced.

We analysed two versions of the tracking protocol, soft-state and hard-state for state inconsistencies, signaling rates and the overall cost. Results indicate how the parameters of the soft-state protocol need to be tailored so that the cost is lower than the hard-state protocol. The results show that if the loss rates are lower higher values can be selected for soft-state refresh timer. We simulated the soft-state tracking protocol to determine paging cache size and the tracking traffic rate. We determined that our centralized TA architecture gives better performance results than the replicated TA architecture.

SIP Paging is also analyzed in order to determine the processing load at SIP Proxy using fluid flow and random walk mobility models and then compare the results with Mobile IPv6 Hierarchical Paging which uses replicated TAs. Paging delay is analyzed and the delay is shown to increase with the loss rate.
and the number of hops between the tracking agent and the access points influence the delay strongly. Context transfer with paging analysis shows that the paging response times can be drastically reduced if the context is transferred during paging.

We designed a new SIP Tracking protocol which relies on the access router creating/refreshing entries in the paging cache triggered by layer 2 handover events of the mobiles. If IAPP is already implemented in a WLAN network, SIP Tracking should be able to interoperate with IAPP. Extensions required for this to the SIP Tracking are minimal and are left for future work.

Three more versions of SIP Tracking protocol can be defined as a soft-state with explicit removal, reliable trigger and reliable trigger/removal messages. These three versions can be analytically compared with the soft-state and hard-state versions we have in the paper. SIP Paging transmission delay analysis can be extended assuming TCP transport for the messages. Replicating the Paging and Tracking Agents has to be considered for fault tolerant operation of SIP Paging. Implementation of SIP Paging and Tracking protocols is also left as future work.

REFERENCES


APPENDIX

<table>
<thead>
<tr>
<th>Data</th>
<th>Type</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Session-Lifetime Elapsed</td>
<td>unsigned32</td>
<td>Fixed</td>
</tr>
<tr>
<td>AAA-Key-int</td>
<td>UTF8String</td>
<td>Fixed(64 octets)</td>
</tr>
<tr>
<td>ISP-Identifier</td>
<td>Unsigned32</td>
<td>Fixed</td>
</tr>
<tr>
<td>ISP-Name</td>
<td>UTF8String</td>
<td>Variable</td>
</tr>
<tr>
<td>NAP/ISP Separate Authentication</td>
<td>Unsigned32</td>
<td>Fixed</td>
</tr>
</tbody>
</table>