On Analysing Cost for Optimizing the Watcher Subscription Time in the IMS Presence Service

Muhammad T. Alam
Room #5323
School of Information Technology
Bond University
Gold Coast, Australia
e-mail: malam@bond.edu.au

Abstract
IMS (IP Multimedia Subsystem) is the technology that will merge the Internet (packet switching) with the cellular world (circuit switching). Presence is one of the basic services which is likely to become omnipresent in IMS (IP Multimedia Subsystem). It is the service that allows a user to be informed about the reachability, availability, and willingness of communication of another user. The flow of messages will be massive for large amount of publishers and watchers joining an IMS system, because of the security architecture of the IMS. Although the IETF (Internet Engineering Task Force) engineers have proposed several solutions to reduce the signalling overhead to facilitate the presence service, the heavy traffic flows have been compromised with several factors like real time view and information segregation etc. The life time of a watcher subscription has not received any attention so far. The constant time set (both short and long) may create bottleneck because of excessive message flow in the network. In this paper, we propose a mathematical model to analyse the system-performance of the IMS presence service during heavy traffic. The model derives the cost functions that are based on the real parameters of the Presence Server. An algorithm is demonstrated based on the derived model to set up an optimal life time for the Presence Server to assign to its joining IMS watchers during busy traffic. Simulation results have been shown that provide useful insight into the system behaviour.

Keywords-**SIP, IP Multimedia Subsystem, Presence Service, Cost, Optimal life time** *(keywords)*

1. INTRODUCTION

IP Multimedia Subsystem (IMS) is a new framework, basically specified for mobile networks, for providing Internet Protocol (IP) telecommunication services [1]. It is the technology that will merge the Internet (packet switching) with the cellular world (circuit switching). It will make Internet technologies, such as the web, email, instant messaging, presence, and videoconferencing available nearly everywhere. Furthermore, the aim of IMS is not only to provide new services but also to provide all the services, current and future, that the Internet provides. Presence is one of the basic services that is likely to become omnipresent in IMS. It is the service that allows a user to be informed about the reachability, availability, and willingness of communication of another user. The presence service is able to indicate whether other users are online or not and if they are online, whether they are idle or busy. Additionally the presence service allows users to give details of their communication means and capabilities.

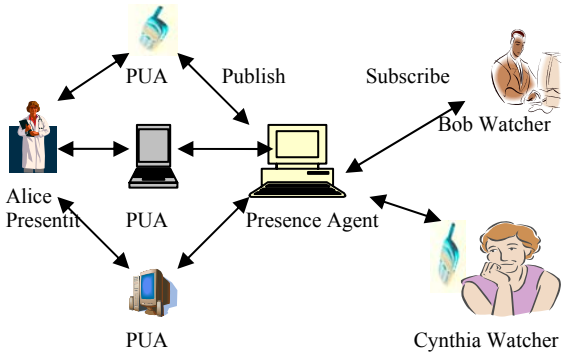


Figure 1: SIP Presence architecture

The presence framework defines various roles as shown in the above figure (figure 1). The person who is providing presence information to the presence service is called a presence entity, or for short a presentity. In the figure, Alice plays the role of a presentity. The presentity is supplying presence information such as status, capabilities, communication address etc. A given presentity has several devices known as Presence User Agents (PUA) which provide information about her presence. All PUAs send their pieces of information to a presence agent (PA). A presence Agent can be an integral part of a Presence Server (PS). A PS is a functional entity that acts as either a PA or as a proxy server for SUBSCRIBE requests. Figure 1 also shows two watchers: Bob and Cynthia. A watcher is an entity that requests (from the PA) presence information about a presentity or watcher information about his/her watchers. A subscribed watcher asks to be notified about future changes in the presentity's presence information, so that the subscribed watcher has an updated view of the presentity's presence information.

3GPP defined in 3GPP TS 23.141 [2] provides the architecture to support the presence service in the IMS. The used interfaces for this service are *Pen, Pw, Pi, Px Ut* and the protocols used are SIP (Session Initiation Protocol), Diameter and XCAP (XML Configuration Access Protocol) as described by the IMS technical specification. The watcher subscription flow is illustrated in the figure 2. The watcher application residing in the IMS terminal sends a SUBSCRIBE request (1) addressed to her list for example sip:alice-list@home1.net. The request (2) is received at the S-

CSCF (Serving Call/Session Control Function), which evaluates the initial filter criteria. One of those criteria indicates that the request (3) ought to be forwarded to an Application Server that happens to be an RLS (Resource List Server). A RLS can be implemented as an Application Server in IMS. The RLS, after verifying the identity of the subscriber and authorizing the subscription, sends a 200 (OK) response (4). The RLS also sends a notify request (7), although it does not contain any presence information at this stage. The RLS subscribes one by one to all the presentities listed in the resource list and, when enough information has been received, generates another NOTIFY request (13) that includes a presence document with the aggregated presence information received from the presentities' PUAs.

Figure 3 shows the RLS subscribing to one of the presentities contained in the resource list.

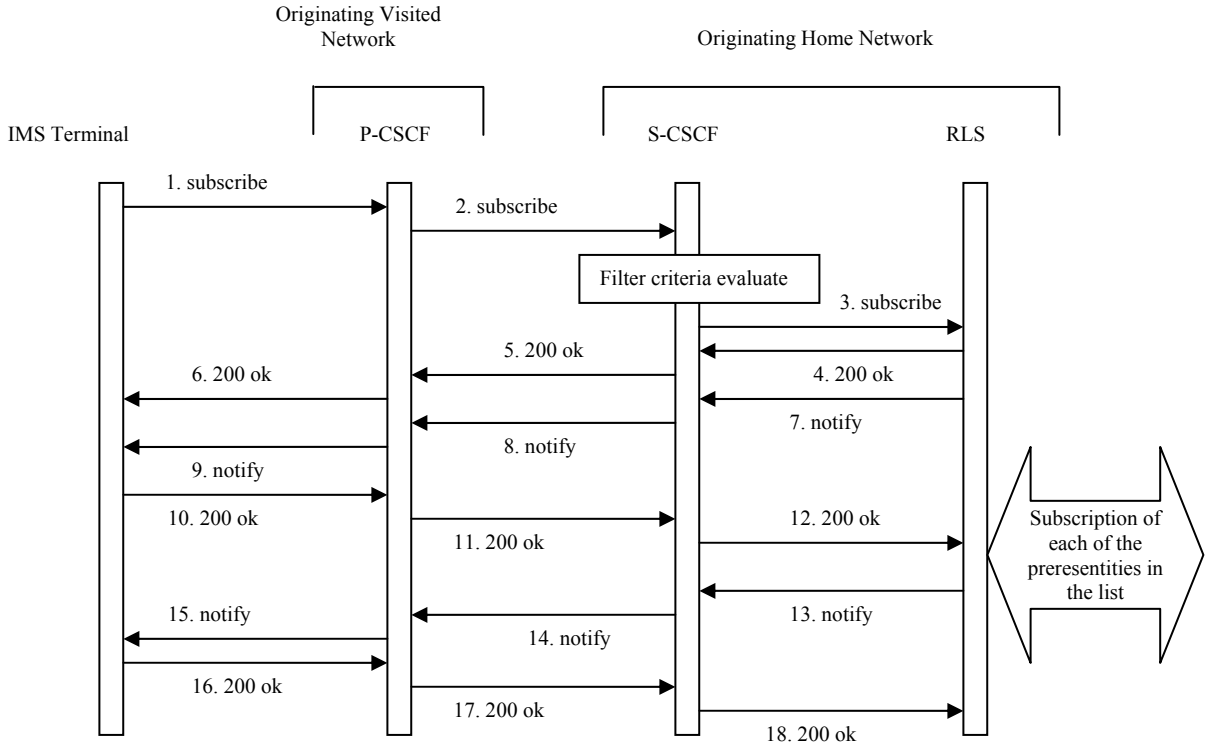


Figure 2: Watcher subscription to own list

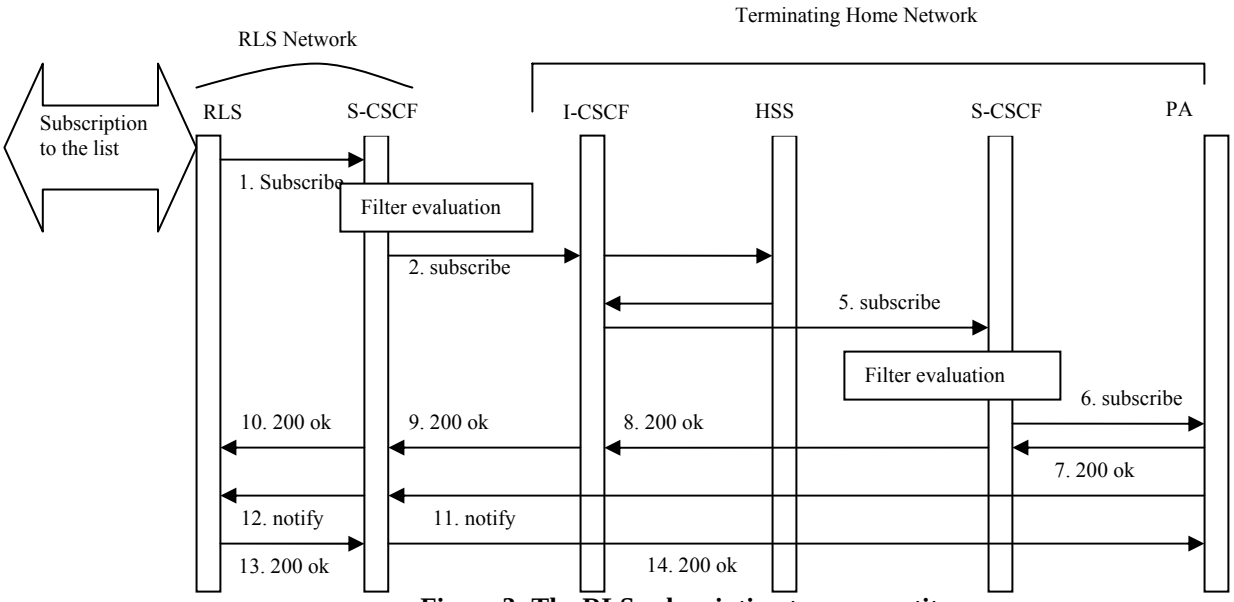


Figure 3: The RLS subscription to a presentity

When the IMS presence application starts, it publishes the current presentity's presence information. Figure 4 shows the flow. The IMS terminal sends PUBLISH request (1) that includes an Event header set of presence. The S-CSCF receives the request (2) that includes and evaluates the initial filter criteria for the presentity. One of the initial filter criteria indicates that PUBLISH requests containing an Event header set to presence ought to be forwarded to the PA where the presentity's presence information is stored. So, the S-CSCF forwards the PUBLISH request (3) to that Application Server. The PA authorizes the publication and sends a 200 (OK) response (4).

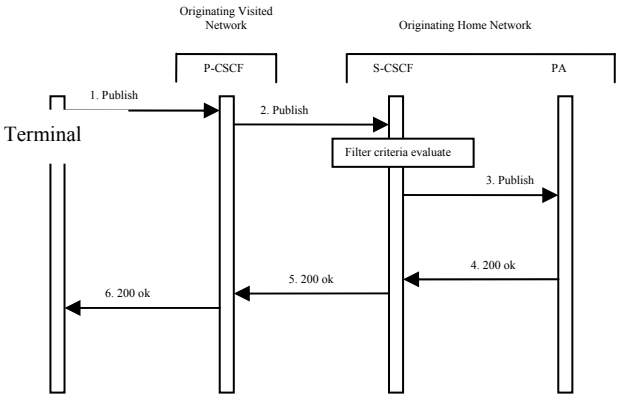


Figure 4: The IMS terminal publishing presence information

The above IMS presence architecture indicates that the flow of messages will be massive for large amount of publishers and watchers joining an IMS system. A watcher should not be able to watch infinite amount of time to its presentities when the PS encounters heavy traffic. Every time a presentity changes position, its watcher will have to be notified. Thus the watcher subscription time needs to be carefully designed. In this research, we propose an optimal time for a PS to provide to a joining IMS watcher based on the system load.

The paper is structured as follows. Section 2 shows the related work and section 3 proposes an analytical model to analyse the cost of the presence service in IMS framework. Section 4 proposes an algorithm to optimize the watcher subscription time with the overhead computation. Section 5 shows the simulation results for the proposed model and section 6 concludes the paper.

2. RELATED WORK

Session Initiated Protocol (SIP) is a prominent protocol today in the third generation network. It facilitates mainly multimedia data transfer. SIP has been chosen in IMS to play the key role for setting up the session while inter-working with other protocols. RFC 3265 [16] defines a framework for event notification in SIP. According to this document, the entity interested in the status information of a resource subscribes to that information. The entity that keeps track of the resource state will send a NOTIFY request with the current status information of the resource and a new NOTIFY request every time the status changes. A watcher receives NOTIFY message every time any of its presentity changes state. Although the event notification framework offers powerful tool in the IMS presence service that allows a watcher to be informed about changes in the state of a presentity, in some situations the amount of information that the Presence Server has to process might be large. Imagine, for instance, IMS presentities of a watcher are driving on highways. The corresponding IMS watcher will get very frequent updates, because the presentities' geographical position change rapidly.

The detail of Session Initiation Protocol and Mobile IP registration can be located in [3] and [4] respectively. Related work can be found in [5], [6]. Guerin (1987) showed that the sojourn time of a mobile node within a Location Area (LA) is exponentially distributed with mean i.e.,

$$t_{sojourn} = \frac{9}{3+2\sqrt{3}}\frac{R}{V}$$

where R is the "radius" of a LA and V is the average mobile node's velocity [17]. The calculated rate of LA boundary crossings is $1/t$sojourn.

The Presence Information Data Format (PIDF) is a protocol-agnostic document that is designed to carry the semantics of presence information across two presence entities. The PIDF is specified in the Internet-Draft "Presence Information Data Format (PIDF)" [7]. The PIDF encodes the presence information in an XML (Extensible Mark-up Language) document that can be transported, like any other MIME (Multipurpose Internet Mail Extension) document, in presence publication (PUBLISH transaction) and presence subscription/notification (SUBSCRIBE/NOTIFY transaction) operations. The Rich Presence Information Data Format (RPID) is an extension to the PIDF that allows a presentity to express detailed and rich presence information to his/her watchers. Like the PIDF, RPID is encoded in XML. The RPID extension is specified in [8]. The Timed Presence extension is specified in the Internet-Draft "Timed Presence Extension to the Presence Information Data Format (PIDF) to indicate Presence Information for Past and Future Time Intervals" [9] and allows a presentity to express what they are going to be doing in the immediate future or actions that took place in the near past.

A subscription can last for a period of time. If watchers want to keep the subscription active they need to renew it prior to its expiration. The PA will keep the PUA updated, using NOTIFY requests about changes in the list of watchers. That is, it will inform a presentity every time a new watcher subscribes or un-subscribes to the presentity's presence information. Every time a watcher wants to subscribe to the presence information of a presentity, the watcher needs to exchange a SUBSCRIBE transaction and a NOTIFY

transaction with the presentity's PUA, just to set up the subscription. Obviously, this mechanism does not scale well, particularly in wireless environment. In order to solve this problem the IETF has created a number of concepts as described below.

1. Partial notification one mechanism on which IETF engineers are working to reduce the amount of presence information transmitted to watchers. A weight or preference is indicated through a SUBSCRIBE request. The mechanism defines a new XML body that is able to transport partial or full state. Thus, the document size is reduced at the cost of information transmitted.

2. Event-throttling mechanism allows a subscriber to an event package to indicate the minimum period of time between two consecutive notifications. So, if the state changes rapidly, the notifier holds those notifications until the throttling timer has expired, at which point the notifier sends a single notifications to the subscriber. However, with this mechanism the watcher does not have a real-time view of the subscription state information.

3. Compression of SIP messages is another technique to minimize the amount of data sent on low-bandwidth access. RFC 3486 [11], RFC 3320 [13], RFC 3321 [12] defines signalling compression mechanisms. Usually these algorithms substitute words with letters. The compressor builds a dictionary that maps the long expressions to short pointers and sends this dictionary to the de-compressor. However, the frequency of data transmission is not reduced in such techniques.

Clearly each of the abovementioned works has limitations and tradeoffs. The life time of a watcher subscription time has not received any attention so far. An optimal watcher registration time procedure to allow Proxy-CSCF to reassign UE (User Equipment) in IMS needs to be considered. Every time UE/IMS watcher needs to re-subscribe while its timer (which is kept shorter than the subscription timer in the network) expires. If the UE does not re-register, any of its active sessions are deactivated in IMS. On the other hand, De-registration is accomplished by a registration with an expiration time of zero seconds. A forced de-registration from the network (Presence Server) may occur in case of data inconsistency at node failure. The constant time set may create bottleneck because of excessive message flow in the network. Specially, if an IMS watcher watches many presentities and if the watcher-subscription-time is not set carefully, it will be notified any changes made in its presentity list. Both long and short life time will introduce overhead in number of messages and cache respectively. Thus an optimal procedure to set the timer of the watcher subscription life time for the IMS node is desirable.

3. THE ANALYTICAL MODEL

This section proposes an analytic approach to model implicit timing algorithm for a PS to provide to a joining IMS watcher when the system load is heavy (i.e., there are massive amount of watchers and each of them watching massive number of IMS terminals).

Let the number of states for a presentity to change is arbitrary. The presentity can hop among any state from its initial state with arbitrary probability. However, the probabilities of coming back to its initial state are equivalent. The scenario is depicted in figure 5. We assume that state zero is the initial position of a presentity which may be thought of its actual anchoring position. The other states may represent the presentity's state change to busy, idle, not available etc. or even the location change for instance, availability in office etc. These state changes reflect the different values of the elements for instance; class, content-type, place-type, privacy, relationship, sphere etc. of the RPID (Rich Presence Information Data Format) extension. We assume that the presentity initial state is saturated so that upon completion of one state change, it will enter to another statically identical state instantaneously. The probability of staying at state zero is q_0 and $\sum_{i=0}^{m} q_i = 1$.

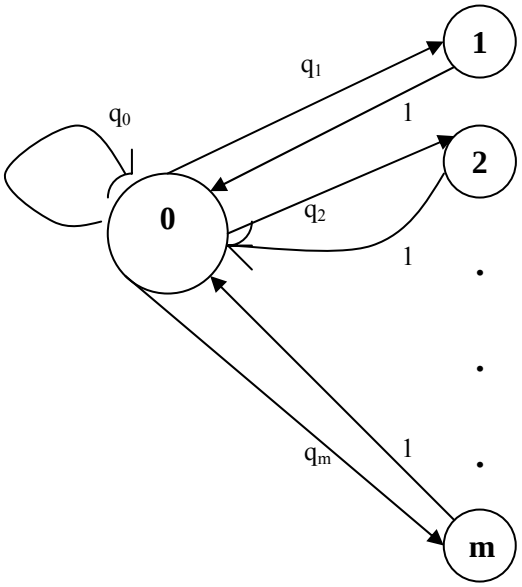


Figure 5-Markov chain for a Presenity's states

With these assumptions, the system can be modelled as a discrete-parameter Markov chain. The transition probability matrix P of the Markov chain is given by:

$$\begin{vmatrix} q_0 & q_1 & \cdots & q_m \\ 1 & 0 & \ldots & 0 \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ 1 & 0 & 0\ldots0 & 0 \end{vmatrix}$$

Since, $0<q_i<1 (i=0,1,\ldots,m)$, then this finite Markov chain is both irreducible and aperiodic. The unique steady-state

probability vector, $\boldsymbol{v}$, is obtained by solving the system of linear equations:

$\boldsymbol{v} = \boldsymbol{v}P$

or

$$v_0 = v_0 q_0 + \sum_{j=1}^{m} v_j, \qquad j = 0, \qquad (1)$$

$$v_j = v_0 q_j, \qquad j = 1,2,...,m. \qquad (2)$$

Using the normalization condition:

$$\sum_{j=0}^{m} v_j = 1,$$

We have (appendix A):

$$v_0 + v_0 \sum_{j=1}^{m} q_j = 1. \qquad (3)$$

Substituting $\sum_{j=1}^{m} q_j = 1 - q_0$, in equation (3), we get:

$$v_0(1 + 1 - q_0) = 1$$

$$\Rightarrow \qquad v_0 = \frac{1}{2 - q_0}, \quad j = 0,$$

From (2) we get, $v_j = \frac{q_j}{2 - q_0}, \quad j = 1,2,...,m. \qquad (4)$

v_j, as defined above is the visit to state j of a presentity. Let r be the total number of IMS presentities observed by the IMS watchers via a Presence Server in the system.

Therefore, the total average cost of presentities' movement for a Presence Server in a real-time interval t, in the long run is:

$$C_p = \frac{trq_j}{(2 - q_0)} \qquad (5)$$

We use the following delays as shown in [14] to establish the cost of a NOTIFY message, C_n via P-CSCF (Proxy-CSCF) at each hop. The parameters used are denoted as follows:

$\lambda_{i1,}$ $i=1,2,...n$: Notify message arrival rate at hop i,

$\lambda_{i2,}$ $i=1,2,...n$: arrival rate of messages other than Notify at hop i,

$\mu_{i1,}$ $i=1,2,...n$: serving rate for each Notify message in hop i,

$\mu_{i2,}$ $i=1,2,...n$: serving rate for messages other than Notify at hop i,

$\rho_{i1,}$ $i=1,2,...n$: load at hop i for Notify messages,

$\rho_{i2,}$ $i=1,2,...n$: load at hop i for messages other than Notify,

where $\rho_i = \lambda_i / \mu_i$, $\lambda_i < \mu_i$,

D_{1i}: the processing delay at hop i,

D_2: the propagation / internet transmission delay at each hop,

D_{3i}: the queuing delay at hop i,

We define $D_{1i} = \frac{1}{(\mu_{i1} - \lambda_{i1})}$

$$D_{3i} = \frac{\frac{1}{\mu_{i1}}(1 - \rho_{i1} - \rho_{i2}) + R}{(1 - \rho_{i1})(1 - \rho_{i1} - \rho_{i2})} \quad \text{from [15]}$$

Where $R = \frac{\lambda_{i1} X_1^2 + \lambda_{i2} X_2^2}{2}$; X_1^2, X_2^2 are the second moments of μ_{i1} and μ_{i2} respectively.

The propagation / internet transmission delay at each hop is considered to be a constant, $D_2 = \Delta$.

The cost of sending a NOTIFY message, C_n is measured as the sum of delays at each node that is involved to send the message between an IMS presentity and a watcher.

$$C_n = \sum_i [D_{1i} + D_{2i} + D_{3i}], \qquad i = 1,2,3...$$

Where, i is the number of hops.

Assuming M/M/1 system at the PS, the expected number of watchers in the server is given by:

$$E(X) = \frac{\rho_w}{1 - \rho_w},$$

Where,

$$\text{Traffic intensity, } \rho_w = \frac{\lambda_w}{\mu_w}$$

$\lambda_{w:}$ Average watcher arrival rate (Poisson), equivalently the watcher inter-arrival times are exponentially distributed with mean $\frac{1}{\lambda_w}$.

$\mu_{w:}$ Watcher service times are independent identically distributed random variables, equivalently the distribution being exponential with mean $\frac{1}{\mu_w}$.

It is noted that a Presentity may subscribe as a watcher and a presentity may be watched by several watchers.

Thus, the total average cost of sending NOTIFY messages, C_T for each watcher in the system (PS) is:

$$C_T = \frac{\sum_{x=1}^{r} N_x (v_{jx} C_{nx})}{E(X)} \qquad (6)$$

Where, N_x is the number of NOTIFY messages generated for a state change of the x^{th} presentity to notify its watchers.

Let, R is the presentity subscription rate in the system and C_{S_K} is the subscription cost for the k^{th} presentity in the system. C_{S_K} can be derived similar to $C_{n.}$

Therefore, the total average cost (in terms of delays) in a real-time interval T is:

$$C(t) = \int_0^T \{\sum_{x=1}^{r} N_x v_{jx} C_{nx} + C_{S_K} R\} dt$$

$$\approx T \sum_{x=1}^{r} N_x v_{jx} C_{nx} + T C_{S_k} R \qquad (7)$$

Note that both r and N_x are function of R i.e., the number of presentities and NOTIFY messages generated in the system at

any period of time dynamically depend on the number of presentities being subscribed in the system. The presentities may be overlapped by watchers; in that case, the *r* will not vary but N_x will. Since, we are only interested in the increasing traffic in this model, we do not address the issue of presentity un-subscriptions/deletions by the watchers.

If the presentities subscriptions are a Poisson counting process, then the probability of *k* presentities' subscription occur in a known period *T* is

$$\Pr[K = k \mid T = t] = \frac{(Rt)^k}{k!} e^{-Rt} \qquad (8)$$

Therefore, the probability of *k* subscriptions for a length of time can be determined by the *Laplace* transformation as follows:

$$p_r(k) = \int_{t=0}^{\infty} \{\frac{(Rt)^k}{k!} e^{-Rt}\} e^{-st} dt$$

$$p_r(k) = \frac{R^k}{k!} \left[(-1)^k \frac{d^k}{ds^k} (\frac{1}{s+R}) \right] \Big|_{s=R} \qquad (9)$$

From (9), a PS can compute the probability of specific number of presentities to be subscribed by the watchers in the system. Thus, the average total cost for presentities subscription per unit time is

$$C_S = \frac{1}{T} \sum_{k=1}^{\infty} p_r(k) C_{S_k} \qquad (10)$$

Rewriting (7) we have

$$C(t) = T \sum_{x=1}^{r} N_x v_{jx} C_{nx} + \sum_{k=1}^{\infty} p_r(k) C_{S_k} \qquad (11)$$

The above model can be applied to the Event throttling mechanism as well to compute the optimal expiration time between NOTIFY messages. We propose an algorithm next that fits with the PS in IMS to generate the optimal subscription time.

4. ALGORITHM AND OVERHEAD

We assume that an IMS watcher will keep timers separate for its presence publishing time and watcher subscription time. According to the analytical model and the cost computation as discussed above, it is recognized that the total average cost for the NOTIFY messages generated for the presence service with an IMS presentity is a function of several parameters, which characterize the presentities mobility, traffic load at the PS and the hop delays. In practice, the value of the watcher subscribe time, *T*, must be specified in the implementation of network topology. If a watcher is mobile and is visiting a network then, *T* can be defined only based on watchers sojourn time in the visited network as discussed earlier in the literature review section. There are quite a number works available today over mobility management and mobile node's cell residence time. We do not address in this paper the mechanism by which a mobile monitors its location and velocity and such issues. A mobile terminal may determine its location through a variety of methods, including the Global Positioning System, signal triangulation, base-station self identifying beacons, or a combination of the above. Other methods and related references on mobile location and velocity determination can be found in [18], [19]. We recall the sojourn time:

$$T = t_{sojourn} = \frac{9R}{(3 + 2\sqrt{3})V} \qquad (12)$$

However, if the parameters for the cell residence time (*R, V*) or above all, the mobility information of a watcher is not known, then cost computation in (6) may be used with conjunction with *e* or *lnx* to define *T*. The question is when and how a Presence Server will compute the necessary parameters for equation 6. The traffic load (ρ_w) for the joining IMS watchers may be computed any time by the PS. Appendix B shows the traffic intensity for massive number of watchers at a non pre-emptive Presence Server. An IMS watcher may subscribe new presentity with the PS while it joins. The number of presentities *r* is available from the watcher subscription list at any point of time. It is difficult to achieve the accurate value for *i* (average number of intermediate hops for all presentities to notify their corresponding watchers) in mobile environment. The values may fluctuate depending on the cell movement of the IMS terminals. The other parameters for instance different delays, presentity mobility vectors and number of states may be computed using heuristic method. This will require the Presence Server to have extra cache and may introduce slight delay to lookup from its routing table. However, the signalling overhead of the NOTIFY messages is expected to be reduced significantly in return which is shown in the overhead collection later in this section.

From the above discussion, the optimal subscription time, T_{op} is a function of several factors, expressed as:

$$T_{op} = f(\rho_w, v, P, N, r, D_1, D_2, D_3, i) \qquad (13)$$

Our simulation work will show the behaviour of (11) for all the varying parameters. In order to achieve the best performance, the following method may be tested for T_{op}:

Rewriting expression (7) as

$$C(t) = e^{C_T t} \; or \; \ln(C_T t) \qquad (14)$$

Thus the optimal subscription time algorithm can be evaluated as follows:

1: If $t_{sojourn}$ == true
2: $T_{op} = t_{sojourn}$
3: Else
4: If $\rho_w \uparrow$
5: Compute T_{op} from (14)

Line 1-4 of the above algorithm will take *O(1)* time to execute where as executing line 5 will take linear time, *O(r)*. The parameters lookup for expression (6) will take *O(a.b)* time if a PS has to lookup the values mainly for *i, r and N* from a table of size *aXb* which is practical.

Next we evaluate the overhead in terms of extra messages (NOTIFY and SUBSCRIBE) sent for various constant time values of watcher subscription time. Figure 6 and 7 show the

resource wasted area for a constant time set, C_{const} that is not equal to the T_{op} for the two proposed curves. Both the two figures have an intersection point, $t=T$. We argue is the optimal choice points are at the curves.

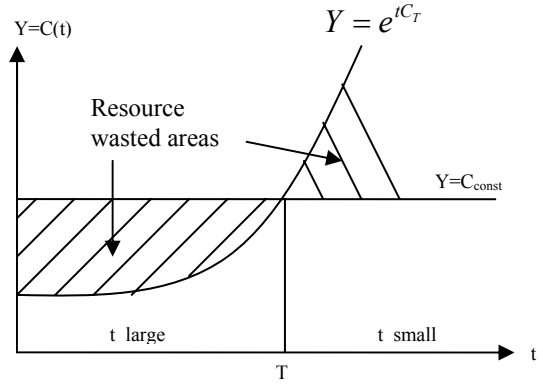


Figure 6-Optimal lifetime of a watcher

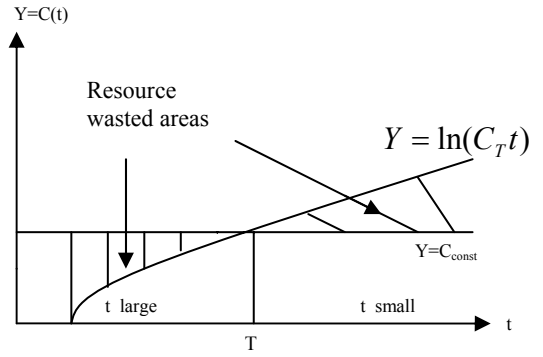


Figure 7-Optimal lifetime of a watcher

We denote smaller subscription time as *t_small* in the $Y=C_{const}$ line if $t>T$ and larger subscription time as *t_large* in the $Y=C_{const}$ line if $t<T$. Since, the model is based on the assumption that the watcher traffic intensity and watched presentities are high in volume, we are particularly interested in the later part of the curves. Analytically, the total average overhead for figure 6 is given by:

$$C_{overhead} = \int_0^T (C - e^{tC_T})dt + \int_T^t (e^{tC_T} - C)dt$$

$$\approx CT - \frac{e^{tC_T}}{C_T}\Big|_0^T + \frac{e^{tC_T}}{C_T}\Big|_T^t - Ct\Big|_T^t$$

$$\approx 2CT - Ct + \frac{1 - 2e^{TC_T} + e^{tC_T}}{C_T} \qquad (15)$$

The overhead for figure 7 can be computed similarly using Simpson's rule. Alternatively, the cost for overhead may be computed quantitatively for a single watcher at the PS. If the watcher subscription time is selected to be smaller than the $T_{op,}$ the watcher will have to subscribe again with the Presence Server and the information of the current presesntities' (which are being watched by the watcher) status will be published to the watcher again as a routine work after it joins the Presence Server. Obviously the watcher will have to wait in the queue once as it tries to re-subscribe. Thus, the average cost of overhead for a watcher, C_{t_small} (in terms of delays) for smaller constant time can be defined as the sum of waiting time in the PS and the extra delays for re-sending the presentities' status to the watcher.

$$C_{t\_small} = \beta(E[W] + C_R) \qquad (16)$$

Where, β (>1) is the ratio between T_{op} and *t_small*; *E[W]* is the waiting time in the M/M/1 queue or in the PS which is defined as

$$E[W] = \frac{\rho_w}{\mu_w(1-\rho_w)} \qquad (17)$$

and, the cost of re-sending the subscribed presentities information is

$$C_R = \sum_s \sum_i (D_{1i} + D_{2i} + D_{3i}) \qquad (18)$$

Where, i is the average number of intermediate hops per message and s is the number of presentities being watched by the particular watcher. It can be easily observed that the inaccurate small constant time for large scale of watchers will be very expensive.

If the watcher subscription time is selected to be larger than the $T_{op,}$ the PS will have to process the signals for the presentities movement for all the watchers during the extra period of subscription time. This cost, C_{t_large} can be retrieved from expression (7) with time interval (t_large- T_{op}).

$$C_{t\_large}(t) = \int_{t\_large}^{T_{op}} C_T dt \qquad (19)$$

5. SIMULATION RESULTS

The Poisson behaviour of presentity subscription (k=1,2 and 3) is shown in figure 8 for growing subscription rate, 50% - 99%.

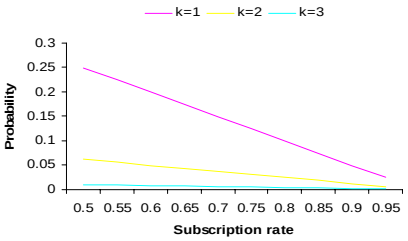


Figure 8-Poisson behaviour

The probability of moving to state j for a presentity is provided in figure 9. The probability of staying at state zero was selected to be *0.1, 0.2, 0.3, 0.4, 0.5, 0.6* and *0.7* where the q_j was derived from $q_j = \frac{1-q_0}{j}$. The figure shows that q_j takes more or less the same value for increasing number of state changes. Similar behaviour follows from figure 10 for v_j. This suggests that the steady state is independent of its

initial position and it is equally likely to be visited in the long run i.e., $v_j = \frac{1}{m}$ for too many status changing states.

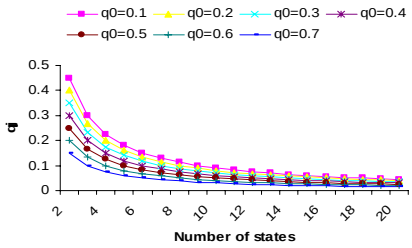


Figure 9-Probability for Growing States

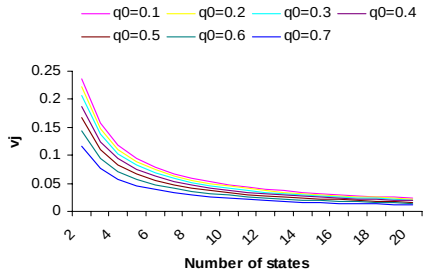


Figure 10-Steady state probability vector

If the watcher subscription rate is constant, then $Y=t\,C_{S_K}\,R$ behaves like a straight line. Thus we centre our simulation over (6). BRITE was used to generate the mobile environment after every interval in a fixed area. Figure 11 shows total cost against number of watchers. The watcher was varied from *50* to *200* for *500* presentities. The message costs were kept constant. We kept q_j as *0.06, 0.09* and *0.17*. All costs go down as the number of watchers increases where the lower transition probability costs reduce to similar values for large number of watchers.

Next we varied *r* for *500* watchers and transition probability vector *0.2*. The message costs were generated randomly. The total cost was found to behave linearly (see figure 12).

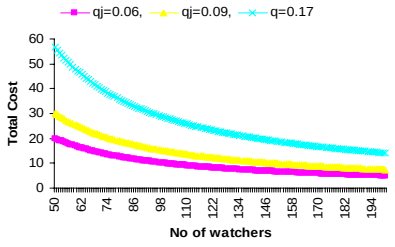


Figure 11-Cost for large watchers

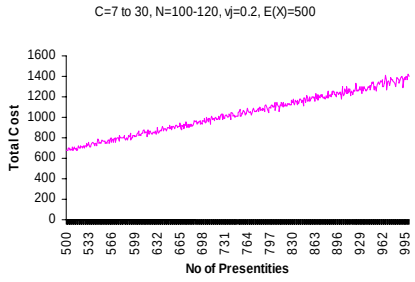


Figure 12-Cost for large presentities

In figure 13, the steady state probability was varied from *0.01* to *0.2* for *500* watchers. The number of messages was randomly generated. The cost goes up slowly with spikes.

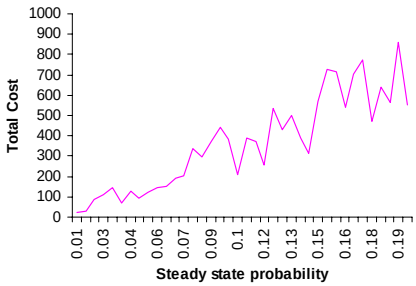


Figure 13-Cost for growing steady state vector

Next we investigate the overhead for *t_small.* The traffic intensity was kept *0.95.* The overhead becomes linear functions of β.

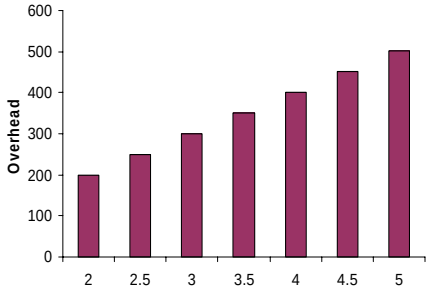


Figure 14-Overhead for varying ß

The overhead for increased waiting time in the PS with fixed C_R is shown in figure 15. The curves go up very slowly.

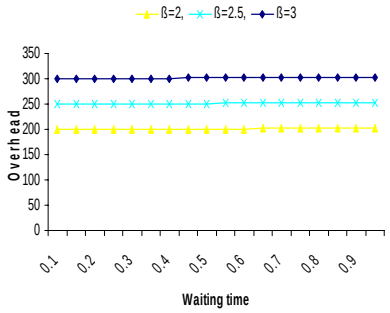


Figure 15-Overhead for increasing wait period

Figure 16 depicts overhead for *t_small* with fixed waiting time *0.5*. C_R was randomly generated between *100* to *950*. The cost for overhead goes up with increasing β.

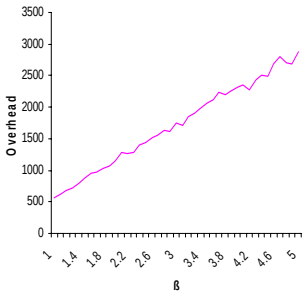


Figure 16-Overhead

The simulation results of overhead computation imply that the ratio between T_{op} and *t_small* and the cost of re-sending the subscribed presentities information are the driving factors than the server waiting time.

6. CONCLUSIONS

It is recognised that a framework for reducing load is essential in IMS for large scale of traffic to facilitate presence service to the terminals. We have proposed an analytical model in this paper to optimize the watcher subscription time in the IMS presence service. By using this model, the cost related to the traffic at the Presence Server can be estimated. The optimal life time of the watcher will reduce the signalling cost for the Presence Server. As an application of the mathematical model in the IMS, an algorithm for dynamically setting the watcher subscription time is proposed in the context of available IMS parameters. Simulation results show the behaviour of the model in terms of cost functions. The overhead is also depicted when the watcher subscription time is not set carefully. Further work is required to test the performance of the algorithm.

APPENDIX A

$$\sum_{j=1}^{m} v_j = \sum_{j=0}^{m} v_j - v_0$$

$$\Rightarrow \quad \sum_{j=1}^{m} v_j = 1 - v_0$$

$$v_0 = v_0 q_0 + \sum_{j=1}^{m} v_j$$

$$\Rightarrow \quad v_0 = v_0 q_0 + 1 - v_0$$

$$\Rightarrow \quad 2v_0 - v_0 q_0 = 1$$

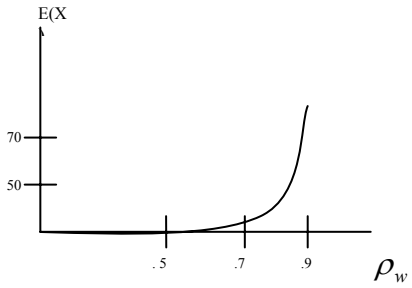
$$\Rightarrow \quad 2v_0 - v_0(1 - \sum_{j=1}^{m} q_j) = 1,$$

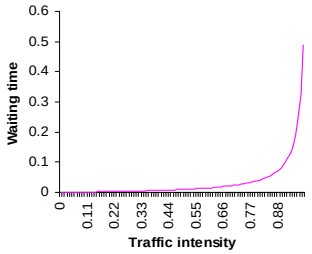
$$[q_0 = 1 - \sum_{j=1}^{m} q_j]$$

$$\Rightarrow \quad v_0 + v_0 \sum_{j=1}^{m} q_j = 1$$

APPENDIX B

The following figures show the behaviour of expected number of watchers and waiting time in the PS assuming that the PS is not idle when there are watchers waiting to subscribe.





REFERENCES

[1] 3GPP TSG SSA, IP Multimedia Subsystem (IMS) – Stage 2 (Release 6), TS 23.228 v. 6.6.0, (2004-06)

[2] 3GPP. Presence service; Architecture and functional description; Stage 2. TR 23.141, 3rd Generation Partnership Project (3GPP)

[3] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler (2002). "SIP: Session Initiation Protocol". *Internet Engineering Task Force,* Internet Draft: http://www.ietf.org/proceedings/02mar/I-D/draft-ietf-sip-rfc2543bis-09.txt. (Work on progress)

[4] D. B. Johnson, C. E. Perkins, J. Arkko (2002). "Mobility Support in IPv6".IETF Mobile IP Working Group, Internet Draft: http://www.ietf.org/internet-drafts/draft-ietf-mobility-ipv6-18.txt. (Work on progress)

[5] C. Perkins (1996). "IP mobility support," Request for Comments 2002, *Internet Engineering Task Force*.

[6] E. Gustafsson, A. Jonsson, C. Perkins (2001). "Mobile IP Regional Registration." Internet draft,

draft-ietf-mobileip-reg-tunnel-05.txt, September, work in progress

[7] H. Sugano, S. Fujimoto, G. Klyne, A. Bateman, W. Carr, and J. Peterson (2003). "Presence Information Data Format (PIDF)." Internet-Draft draft-ietf-imppcpim-pidf-08, *Internet Engineering Task Force*, May 2003. work in progress.

[8] H. Schulzrinne (2004). "RPID – Rich Presence Extensions to the Presence Information Data Format (PIDF)." Internet-Draft draft-ietf-simple-rpid-03, *Internet Engineering Task Force*, March 2004. Work in progress

[9] H. Schulzrinne (2004). "Timed Presence Extension to the Presence Information Data Format (PIDF) to indicate Presence Information for Past and Future Time Intervals." Internet-Draft draft-ietf-simple-future-01, *Internet Engineering Task Force*, March 2004. Work in progress

[10] A. B. Roach, J. Rosenberg, and B. Campbell (2003). "A Session Initiation Protocol (SIP) Event Notification Extension for Resource Lists." Internet-Draft draft-ietf-simple-event-list-04, *Internet Engineering Task Force*, June 2003. Work in progress

[11] G. Camarillo (2003). "Compressing the Session Initiation Protocol (SIP)." RFC 3486, *Internet Engineering Task Force*, February 2003

[12] H. Hannu, J. Christoffersson, S. Forsgren, K.-C. Leung, Z. Liu, and R. Price (2003). "Signalling Compression (SigComp) – Extended Operations." RFC 3321, *Internet Engineering Task Force*, January 2003

[13] H. Hannu, J. Rosenberg, C. Bormann, J. Christoffersson, Z. Liu, and R. Price (2003). "Signalling Compression (SigComp)." RFC 3320, *Internet Engineering Task Force*, January 2003

[14] M. T. Alam, Z. D. Wu (2005). "Comparison of Session Establishment Schemes over IMS in Mobile Environment." *Fifth IEEE International Conference on Information, Communications and Signal Processing (ICICS 2005),* December 6-9, Bangkok, Thailand (to be presented)

[15] N. Banerjee, K. Basu, S. Das (2003). "Hand-off delay analysis in SIP-based mobility management in wireless networks." *IEEE Parallel and Distributed Processing Symposium, Proceedings. International* (22-26 April 2003)

[16] A. Roach (2002). "Session Initiation Protocol (SIP)-Specific Event Notification." RFC 3265, *Internet Engineering Task Force* (June 2002)

[17] R. Guerin (1987). " Channel occupancy time distribution in a cellular radio system." *IEEE Transactions on Vehicular Technology 35(3)*, (August 1987)

[18] M. Hellebrandt, R. Mathar, M. Scheibenbogen (1997). ''Estimating position and velocity of mobiles in a cellular radio network.'' *IEEE Transactions on Vehicular Technology,* vol: 46, pp: 65-71 (Feb 1997)

[19] C. Wu, H. Lin, L. Lan (2002). ''A New Analytic Framework for Dynamic Mobility Management of PCS Networks." *IEEE Transactions on Mobile Computing,* vol: 1, No: 3, pp: 208-219