An Analysis of Intelligent Agent Framework for Proxy Mobile IPv6 (IAF-PMIPv6)
Chander Kailash
Research Scholar, Maharishi Markendeshwar University, Mullana, Haryana, India.
E-mail: kailashchd@gmail.com

Abstract

This paper begins with analyzing the metrics desired for evaluation of an agent based frameworks in PMIPv6 domain. Further, it evaluates an Intelligent Agent based framework for Proxy Mobile IPv6 (IAF-PMIPv6). Although there exist various metrics such as performance, scalability, stability, security and trustworthiness etc. for evaluating an agent based framework but the usage of metrics is application dependent. This paper initially analyses & explains the metrics for the framework under consideration and later evaluates the same on the mentioned criteria; it was found that IAF-PMIPv6 meets the standards of evaluation.

Keywords: PMIPv6, Vertical Handover, Mobile Agents, Intelligent Agent

1. Introduction

PMIPv6 is employed for heterogeneous networking environment supporting various access network technologies such as 3G/4G cellular network, WLAN and WiMAX network. In Proxy Mobile IPv6 when a MN moves and attaches itself to a MAG then that MAG should detect the MN's attachment and initiate the necessary procedures to authenticate and authorize the MN by giving it access to the network. Figure 1 shows the initial attachment and handover process.

1. Initially on attaching to MAG, MN-Identifier (MN-ID) of Mobile Node is authenticated via access security protocols on the network to be accessed.
2. After successful access authentication MAG obtains the MN’s profile.
3. MAG sends a Proxy Binding Update (PBU) to the LMA of Mobile Node regarding its current location.
4. After receiving the PBU message, LMA assigns a MN-HNP (Home Network Prefix) and creates a Binding Cache Entry (BCE) that binds the Proxy-Care of Address of MAG with MN-HNP.
5. It also establishes a bi-directional tunnel to MAG and sends a Proxy Binding Acknowledgement (PBA) message including the MN-HNP.
6. After receiving the PBA message, tunnel is being sets up between MAG and LMA. After that it sends Router Advertisement (RA) message to MN on the access link to advertise the MN-HNP as the hosted on-link-prefix.
7. On receiving RA messages Mobile Node configures the IP address. After completion of configuration procedure, MN uses this address for all future packets delivery.
8. When MN moves in the area of MAG_{new}, MAG_{prev} sends a DeRegistration PBU (DeReg PBU) to LMA.
9. Once LMA received DeReg PBU, it sends a PBA message to MAG_{prev}.
10. On the other hand when MAG$_{\text{new}}$ detects the attachment of MN, it obtains the MN-profile using MN-ID. It uses similar authentication steps from 1 to 7 as performed during initial attachment.

The Traditional PMIPv6 is represented as the function defined by the equation (1) given below.

$$PMIPv6 = f \left( LS, AAA, Reg, RS - RA \right)$$  \hspace{1cm} (1)

$LS$ = Link Switching  \hspace{1cm} $AAA$ = Authentication, Authorization and Accounting

$Reg$ = Registration  \hspace{1cm} $RS-RA$ = Router Solicitation and Router Advertisement

PMIPv6 suffers from unacceptable handover latencies and packet losses. Where, Handover latency means the maximum time interval in which mobile node does not receive any packet due to the process of handover. On the other hand, packet loss means number of downstream packet lost at the mobile node during handover period. Therefore it is necessary to design and specify PMIPv6 extension that enhances its performance. The solution of these problems is addressed by
implementing Intelligent Agents and the process significantly reduces the handover latency and packet loss in PMIPv6.

Today agent technology is associated with various expectations and used in many different contexts. In particular mobile agents enable new ways of implementing telecommunication services. A mobile agent is a computational entity that can migrate in a heterogeneous environment. Majority of the work in mobile agent system deals with the development of mobile agent system based on mobile computing, mobile communications and mainly in telecommunications [25].

Numbers of frameworks were proposed, which uses the intelligent agents in telecommunications sectors like in Intelligent Call Management [27] solution for convergent voice and data service, which gives the user more flexibility in order to respond to an incoming call. Intelligent Call Management System (ICM) allows the call to be handled intelligently using intelligent agents. Mobile AGeNt Architecture (MAGNA) [25] is another approach being developed by GMD-Fokus to enhance the concepts of TINA and similar architectures (e.g., OMG’s OMA architecture) toward a generic architecture for future telecommunications applications, in which the traditional client/server concept is harmoniously complemented by agent concepts. However, general concept of agents is already being used by Telecommunication Information Networking Architecture (TINA) [26]. Framework like Advanced Communication Technologies and Services (ACTS) formed a new ACTS Cluster for Intelligent Mobile Agents for Telecommunication Environments (CLIMATE)

With the emergence of agent-oriented technology, now telecommunications support better distribution and management. In order to do their job, agents may be transported throughout a network to remote sites in order to perform specific tasks [24].

The paper is structured as follows: Section II provides the related work in the area of evaluation of agent based framework. Section III provides brief introduction of IA-PMIPv6. Section IV explores the evaluation parameters from the existing works and in parallel, evaluates the proposed framework. Section V concludes the paper.

2. Related Work

The issue of performance, in particular, is very important in emerging next generation mobile communications. Numerous studies have shown that performance of systems depends upon user-perceived Quality of Service. The performance of traditional software systems is conducted through experimentation and monitoring, simulation and combinations thereof. However performance of mobile agent systems is more complicated than for traditional systems. Because of variety in distributed computing, dynamic nature, mobility and agile configuration for mobile agents, makes it hard to establish a concise and stable representation. Usage of mobile agents in next generation heterogeneous environment opens numerous problems related to security, communication, intelligence, performance and scalability [19].

The MT/MC/d assignment algorithm [29] reduces the MF (Mobile Floating) agent assignments and thus reduces the background overhead cost of high mobility density users as compared to
radius-d algorithm. Analysis and results of the proposed scheme in terms of latency reduction and system overhead cost increment versus mobility density are given. The evaluation results show that the MF-agent scheme can improve overall performance by more than 20% for any mobility density of users.

The hierarchical performance analysis framework [20] can be used to study the performance characteristics of different mobile agent platforms. Three layers named Micro-benchmarks, Micro-kernels and Micro-Applications are used to investigate the performance of mobile agent based applications. At first layer, it explores and checks the performance of mobile agent platforms while at second layer, it investigates implementations for popular application kernels upon simple workloads. In third, it studies particular functionalities running on realistic workloads.

For instance, in [21], Strasser and Schwehm compared analytically the performance of agent migration and remote execution using Mole platform [22] with the more traditional approach of remote procedure calls (RPC). Their performance model takes into account issues, such as network throughput, communication latency, and network load. Furthermore, the use of model is to identify situations where agent migration has performance advantages over remote procedure calls.

Another approach used by Kotz et. al is mathematically modeled to study a more extended mobile-agent application scenario in [23]. The authors used two performance metrics for their comparison study: computation and bandwidth requirements. They concluded that the mobile agent approach trades server computation and cost for savings in network bandwidth and client computation, which is an important remark in the context of “thin” clients used in mobile-computing applications.

Another thread to examine performance properties of mobile-agent systems is used by Rana in [30] to investigate the performance properties of two agent design patterns, “Task” and “Interaction," and a combination thereof. The study is performed in the context of an agent based, e-commerce application.

Further, Juneja et. al [13] have evaluated their works using fuzzy logic. Sharma and Juneja [14] explored evaluation metrics for Multi-agent framework employed in cellular networks. Singh et. al [15] also presented metrics for Multi-agent framework employed in semantic web. Ibrahim et. al. [28] agent based independent test-bed model is proposed for knowledge aware environments. It is proved that when agents work in a team they can achieve maximum output as a whole. Their work highlighted the non-functional issues like coordination, performance, scalability and security on which agent based frameworks can be evaluated.

Hexmoor et. al in [16] provided metrics such as autonomy, timeliness and purposefulness, robustness and fault-tolerance as general evaluation parameters for agent based frameworks. Karageorgos et. al. in [17] proposed a framework to evaluate Agent Based System (ABS) engineering methodologies against a number of criteria related to design complexity.

The insight into the existing literature [10, 11, and 12] reflects that there are no well defined or standard metrics in literature on which agent-based frameworks can be evaluated and also not
many agent based applications are available for such evaluation and analysis. Upcoming subsection explains the Intelligent Agent Framework for Proxy Mobile IPv6 (IAF-PMIPv6).

3. Intelligent Agent Framework for Proxy Mobile IPv6 (IAF-PMIPv6)

IAFPMIPv6 framework provides a complete solution for improving the performance of PMIPv6 domain. The intelligent agents were deployed at different levels of PMIPv6 environment and named as MN\textsubscript{agent}, MAG\textsubscript{agent} and LMA\textsubscript{agent}. These agents are collectively used for selecting best network, emigration solution for critical situations, reducing packet loss during handover, reducing and balancing load etc. The proposed agents also carry trust certificate while communicating with each others. Figure 2 shows the integrated view of the proposed framework.
As shown in the figure 2 the agents are deployed at MN, MAG and LMA in PMIPv6 domain. These agents known as MN_agent, MAG_agent and LMA_agent interact with each other to perform seamless communication. At the initial stage MN_agent maintains the user preference list which is updated periodically based on message received from MAG_agent. Once the MN connects to the preferred MAG, MAG_agent sends the Proxy Binding Update (PBU) to its LMA_agent. It also maintains the MN_log for the future interaction and starts buffering the packets. So that in case of handover situation the buffered data can be transfer to the target MAG without loss of packets.

MAG_agent also roam to its neighboring MAG_agent to select the best available network. Cost of service, data transfer rates, mobility of node, signal strength, network traffic, security and drainage rate of battery [6] are the major factors for selecting the best network. Sorting of the available network as per the above factors is also done by MAG_agent and the same will be transfer to MN_agent for updation of MN preference list.

On the other hand MN_agent also computes the Remaining Battery Life (RBL) [5] for shifting the device to less power consumption networks in case of critical situations. When more than the desired number of mobile nodes is attached to the MAG, it gets overloaded, causing the end-to-end transmission delay. LMA_agent plays a very vital role to reduce the load [8] of MAG and balance it if required. LMA_agent computes the load of each MAG by comparing the threshold value of each MAG and transfer the MNs to another least loaded MAG in overload situations.

To perform the above assigned duties mobile agents travel from one system to another in a network transferring their code, data and execution state. Therefore, reliability is a vital issue for deploying the mobile agent system. The reliability can be gained by implementing trust certificate for mobile agents. Although, an X.509 certificate is good enough to identify the identity of a mobile agent but it fails to confirm the reliability and credibility of the agent under consideration. Thus every deployed agent is added with a trust certificate. The trust certificate has been weighted on the basis of factors like Direct Experience, Third Party References, Confidentiality, Persistence Execution Trust [7] etc. The unique contribution of this part has been computing reliability and then generating trust certificate which has been mathematically modeled & proved to inconsistency with existing models.

In IAFPMIPv6, the functioning of traditional PMIPv6 protocol is modified with an addition of more agent based functionalities. So the composite function of proposed framework is defined below.

\[
IAFPMIPv6 = f(\mu, \alpha, tw, \omega) \\
\text{Where,} \\
\mu = f(s) \left\{ \begin{array}{ll}
(s = LS, AAA, REG, AAM), & \text{when new user} \\
(s = LS, REG, AAM) & 
\end{array} \right. \\
\alpha = \sum_{i=1}^{n} W_i \\
tw_i = \frac{\sum_{i=1}^{n}(cw_i + pw_i)}{2} \\
\omega = \gamma_1 + \sum_{i=2}^{k} \pi_i \gamma_i 
\]
After a careful investigation of the evaluation parameters explored in the previous section, this work considers performance; effectiveness, robust, scalability and security are the most suitable parameters for evaluating IAF-PMIPv6. Upcoming subsections evaluate the proposed work on the basis of these parameters only.

4. Evaluation Parameters of IAF-PMIPv6

As explained the equation 2 that IAFPMIPv6 is a function of four different parameters i.e. Performance ($\mu$), Effectiveness ($\alpha$), Trustworthy ($tw$), Scalable ($\omega$). These parameters make that framework more efficient.

4.1. Performance ($\mu$)

Literature explored in the previous section clearly indicates that performance is the most suitable evaluation parameters for evaluating IAF-PMIPv6.

Handover latency is the vital factor to estimate handover performance of one protocol. In PMIPv6 handover latency [9] is the sum of link-switching (LS), AAA-Auth, Registration (Reg) and RS-RA as shown in equation 1. Link Switching is a delay during layer 2 handover and AAA-Auth is a delay during authentication of an MN through AAA infrastructure. Registration is a delay during binding updates to LMA. RS-RA is delay during exchanging of a router solicitation (RS) and router advertisement (RA) messages between the MN and the MAG.

In IAF-PMIPv6 the AAA-Auth processing will be done through MN$_{log}$, maintained by Mobile agent earlier. MAG$_{agent}$ is maintaining the log for visited mobile nodes enabling the network to skip the authentication again and again. However, AAA-Auth infrastructure is applicable for new mobile nodes. Apart from the AAA-auth, RS-RA is also replaced with Agent Advertisement Message (AAM), which significantly reduces the delay caused by RS-RA. Now, the handover latency is given as in equation 3.

Where, Handover latency is considered as the time between last packet from previous MAG and first packet from new MAG as handover latency. An implementation & simulation of the same resulted into slight reduction in handover latency of IAF-PMIPv6 compared to traditional PMIPv6. This scheme slightly improves the total handover latency. In addition to this smart buffering technique prevents the packet loss during the handover period.

4.2. Effectiveness ($\alpha$)

The key factors influencing the agent based framework’s [6] effectiveness is to select best network for mobile node. Each network would be assigned a weight which is based on the QoS parameters i.e. cost of service offered, data transfer rate, mobility of node, signal strength, network traffic, security and drainage rate of battery that it provides and satisfies the end user requirements as per equation(4). MAG$_{agent}$ in IAF-PMIPv6 computes the sum of all weight assigned to a particular network which is then normalized within the range of 0-1. This helps the MN$_{agent}$ to prepare preference list for mobile node as per the efficiency of networks.
4.3. Security (tw)

Robust system should have a feature of handling abnormal and exceptional situations like fault tolerance, robustness against Denial-of-Service (DoS) attacks, Shared data remains confidential at any time. IAF-PMIPv6 handles those situations by merging trust certificate [7] of agents deployed in the domain. This trust certificate can be computed some of the parameters like confidentiality, persistence, and execution trust along with X.509 certificate. Based on previous weight \( (pw_i) \) and current weight \( (cw_i) \), a trust weight \( (tw_i) \) of a mobile agent is calculated. To calculate the \( cw_i \) every parameter (Direct Experience, Third Party References, Confidentiality, Persistence, Execution Trust) is assigned values in the range of \([0, 1]\). For instance, \( cw_i = 0 \) means distrust i.e. agent is non-trustworthy and \( cw_i = 1 \) implies agent is fully trustworthy. If the value assigned between 0 and 1 it means two entities trust each other up to an extent. In order to compute, \( tw_i \) between two interacting agents, an average of \( pw_i \) and \( cw_i \) is calculated as given by equation (5).

The trust certificate generated by the proposed model not only improved the reliability and trustworthiness of agents but also added to improving security of the whole agent-based framework.

4.4. Scalability (\( \omega \))

Scalability metric balances timeliness, cost and volume and other attributes in the system, as a function of its size. As a system is scaled up, the average tour increases and this has a major effect on performance and scalability limits [31].

Scalability issue of an IAF-PMIPv6 refers to the performance of the system at different size i.e. whether an increase in the number of mobile nodes along with agents in a PMIPv6 domain adversely affects the functioning of already existing agents or conversely it might improve the performance of IAF-PMIPv6. In IAF-PMIPv6 two algorithms proposed [8] initially for reducing the load & later for balancing the load, if still required. The work proposes mathematical equations that are able to compute the reduction in load & hence the amount of load to be balanced.

\( \text{MAG_agent} \) initially moves to one mobile node to gather the data and then traverses the complete list of all mobile nodes attached to the respective MAG at that particular instant of time. If for the first node the size of reduced data is given by \( \gamma_1 \), then the total Load Reduced (\( \omega \)) at MAG is given by equation(6)

\[
\text{Load Reduced at MAG} = k \times (1 - \pi) \times \gamma_1
\]

Where \( k \) is the number of mobile nodes connected with MAG and \( \pi \) (0< \( \pi \)<1) is the aggregation factor multiplied with the sequence of reduced data.

Further, the communication overhead can also be reduced if the shorter packets can be combined to form bigger packet before transferring on to the network. Now, \( \omega \) shall be considered as the current load of MAG and is compared with \( \theta \) i.e. the threshold load set for the MAG. If \( \omega > \theta \)
then the load LMA\textsubscript{agent} is activate
d and executes the load balancing strategy as proposed in next
section else the MAG continues to handle the tasks allocated to it.

4.5. Robustness

Persistence of a system’s characteristic behavior under conditions of uncertainty makes a
framework robust. IAF-PMIPv6 proactively migrate data from one network to another especially
when the remaining time of current transaction is more the remaining battery life (RBL) of
current device. However, if no suitable network is found then it gives an option of moving data
from current device to other alternative device.

MN\textsubscript{agent} computes Remaining Transaction Time (RTT) and Remaining Battery Life (RBL) based
on the data provided by MAG\textsubscript{agent}. The intelligence of the framework [5] is that MN\textsubscript{agent}
computes RTT intelligently by learning from the pattern of data sizes utilized in past few cycles.
Mobile agent initiates the handover to target network which process data at high rate with low
power requirement at compare to current network. Moreover if current available networks are
unable to process the transaction within Remaining Battery Life (RBL) then remaining data
should be migrated to an alternative device. So that transaction should not be abandon. This
emigration process makes the IAF-PMIPv6 a robust framework.

As shown the figure 3 the cost of handover is slightly improved in IAFPMIPv6. The additional
weightage is given to IAFPMIPv6 because of smart buffering, load reduction; load balancing,
trust certificate, emigration framework, and selection of best network etc. All these tasks are
performed by agents during ideal time. So when the actual handover process starts the overall
cost reduced and efficiency of the framework improves. The work has been simulated using Java
SDK and reflects that results are in comparison with other existing strategies.
5. Conclusion

The strong notation of agents is the thrust behind implementation of PMIPv6. Cooperation among agent societies employed in various domains (homogenous or heterogeneous) gave rise to many issues pertaining to agent communication, which needed attention. This work proposed a novel Intelligent Agent Framework for PMIPv6 (IAF-PMIPv6) supporting almost all major issues concerning handover process. Also this work evaluated the proposed framework on metrics meant for evaluation of agent based frameworks. Empirical results found are sound and meet the desired expectations. This framework has been partially implemented in Java Agent Development Framework (JADE).

References:


[8] Dimple Juneja, Kailash Chander “A Load Reduction and Balancing Scheme for MAG Operating in PMIPv6 Domain”, communicated to The International Journal for the Computer and Telecommunications Industry, ELSEVIER


