PERORMANCE ANALYSIS OF A NODE MONITORING PROTOCOL IN UBIQUITOUS NETWORKS

Sarada Prasad Gochhayat and Pallapa Venkataram
Department of Electrical and Communication Engineering,
Indian Institute of Science,Bangalore, India,
{ecesarada,pallapa}@ece.iisc.ernet.in.

ABSTRACT

The proliferation of the ubiquitous applications in different sphere of life like ubiquitous health monitoring system, and ubiquitous learning system demands the network monitoring, to check whether the nodes are getting the promised resources and inputs. Dynamicity and heterogeneity of the ubiquitous network adds challenges to the monitoring of a node because of node's mobility and migration between heterogeneous networks. The network monitors the health condition of a node by supervising the network status, like data rate, delay, energy and throughput, as well as events such as node failure, network failure and fault in the network. In this paper, we propose a node monitoring protocol to monitor the health condition and behavior of a Unode (Ubiquitous application node) in a ubiquitous network using agents (both static and mobile). The mobile agents follow the Unode by migrating to the subnetwork (where the Unode is about to migrate), and collect network resource availability status and requirement of the Unode. We deploy a static agent which has complete information about all Unodes on their mobility and health conditios, etc. The past behavior of the applications and networks, and history of the Unode and the predecessors are taken into consideration to help static agent to take appropriate decision during the time of emergency situations like unavailability of resources at the local administration, and to predict the migration of the Unode. We have simulated different ubiquituous applications with Unodes range from 50 to 100 and tested the designed node monitoring protocol. We have built the analytical model of the node monitoring protocol using Markov chain concept. The results obtained reflects the effectiveness of the technique.

Keywords: Ubiquitous computing, Ubiquitous network, Static Agent, Mobile Agent.

1 INTRODUCTION

Currently, the applications of the ubiquitous computing are an interesting and important research topic. Projects like UbiMon [1], [2] to provide ubiquitous heathcare monitoring, and ubiquitous learning environment [3] to make education ubiquitous demand ubiquitous network monitoring for their efficient implementation.

The ubiquitous network, which provides uninterrupted connectivity to the users anytime, anywhere, any device on any network technology, is a part of ubiquitous computing environment. Generally, it is a set of heterogeneous subnetworks such as WLAN, Bluetooth, WMax, cellular network and sensor network. Teruyasu Murakami [4] defined ubiquitous network from the user's perspective, saying that the ubiquitous network must provide access to the broad-band network to the user from literally anywhere.

The nodes running ubiquitous application requires ubiquitous connectivity unobtrusively to provide the service to the user. These nodes are generally highly mobile, and require sustained Quality of Service (QoS) in the presence of network problems like scarcity of resources, and during their migration into a subnetwork, to continue the operation unobtrusively.

1.1 Necessity of Ubiquitous Node Monitoring

To provide efficient network access to the user in a ubiquitous computing environment with the mobility to allow for always-on connection regardless of the location of usage, and to check the bottleneck, that may arise because of buffer occupancy or faults in the system [5], [6], the ubiquitous network needs to be monitored continuously.

The work given in [7] explains continuous

monitoring of the network status and its resources are necessary to ensure proper network operation. QoS monitoring is becoming crucial to Internet Service Providers (ISPs) for providing quantified QoS-based services. By QoS, they refer to a service offering where one or more performance parameters (i.e., throughput, delay, loss and delay variation) are quantified.
Problems like degradation of throughput and energy, packet loss and buffer overflow may arise in different subnetworks, which degrade the network performance. So, it is required to monitor the network resources and their status continuously to ensure the healthy condition of the network (the expected QoS from network without any abnormality), and to guarantee functionality of the rendered services and improve reliability.
The goal of this work is to develop an agent based ubiquitous node monitoring protocol. The protocol periodically collects network performance information, accumulates the statistics about Unodes' (ubiquitous application node) network resource utilization and informs the higher level statistical information to the application manager. It will also attempt to notify the management if any exceptional events occur. Conventional network monitoring protocols like SNMP (Simple Network Management Protocol) can have information bottleneck and processing load at manager, mobile agent based network monitoring can be considered as a solution. While RMON (Remote network MONitoring) capable of monitoring subnetwork wide behavior, our work is mostly focus on the behaviour of a Unode in a subnetwork, which consist of both local nodes and Unodes, in a ubiquitous computing environment.
In this paper, we propose a novel method to monitor Unode to observe its behavior and to ensure that it gets the promised resources and the QoS. Monitoring is essential to provide best service to the Unode in a ubiquitous network in coordination with the local administration (LA) (i.e., the Base Station (BS) in GSM/GPRS, and Access Point (AP) in WLAN.)
The rest of the paper is organized as follows: Section 2 discusses some of the existing works; the importance of mobile agent based network management is discussed in Section 3; Section 4 presents the monitoring protocol architecture; Section 5 presents the analytical model of the proposed protocol. Section 6 discusses the simulation procedure and followed by results and conclusion in Section 7 and 8 respectively.

2 SOME OF THE EXISTING WORKS

Monitoring the health condition of a node or an application is the central part of a management system in a ubiquitous computing environment. In the ubiquitous computing environment, monitoring of the function and performance of large scale service objects are essential to guarantee functionality of the service and improve reliability. Designing a monitoring system for ubiquitous computing environment has been an active area of research. But not much work is done on the network monitoring in ubiquitous computing environment.
The work discussed in [8] focuses on detecting fault in a ubiquitous computing environment by monitoring the state of the monitored objects and detecting occurrence of fault. The work is based on deciding the appropriate timeout value in asynchronous distributed systems like ubiquitous computing environment as the detection mechanism solely based on the timeout event, which occurs when response does not come in time after monitor sending a request message to the monitored object. [9] proposes a monitoring system to monitor the application which participates in collaboration to achieve a service goal in a ubiquitous computing environment. The system collects and visualizes the state of the member applications for collaboration.
Few attempts have been made to by using sensors. In [10], Ji-Hye Bae et al presented MONETA (MONitoring system for the Embedded TArget device), which is an embedded monitoring System for ubiquitous network environments. The system collects, analyzes, and displays the state information from the wireless sensors. DRMonitor [11] is a system for monitoring usage of computing resources of networked heterogeneous (Linux and Windows NT's derived) personal computers. This is aimed to serve resource-monitoring applications and to assist load-balancing policies by providing performance and load data about each monitored machine included in the system.
In [12], Remos system has been described. Remos provides resources information to distributed application. The data rate and buffer occupancy in each element may be required for diagnosis purposes when there is a perceived bottleneck. A scalable real-time monitoring system is given in [13]. The system is required to supply smart services to customers, on time in ubiquitous smart space.
Conventional centralized network management, based on SNMP, can have information bottleneck and processing load at manager, which can be solved by distributed agent based management. The traffic between managed nodes and management station reduces as the mobile agents perform the predefined management task at the managed devices with minimum interaction with the management station. Similarly, the execution of certain functionalities at the MA reduces the workload of the management system.

3 MOBILE AGENT BASED NETWORK MANAGEMENT

Mobile agents are autonomous programs that execute on a node and travel from nodes to nodes resuming their execution there, and performing actions on behalf of the user (or application). Hence, they are not bound to the node where they were created or started execution for the first time. The interesting features of mobile agents, which attracted most of its admirers, are autonomy, adaptability and capability to move to move to different nodes.

There are many mobile agent platforms available those offer services like execution, communication, mobility, tracking, directory, persistence, security, to their agents. An execution environment provided by the platform is the most basic service which allows agents to run their code. The mobility service enables an agent to move among the execution environments. JADE and Voyager are most famous mobile agent platform which provide the services.

The advantages of MA based network management with regard to functional area are performance, configuration, accounting and fault management [17]. MA based network management system is also the answer to the scalability problem faced by centralized network management. The management task is assigned to an agent, which executes the management logic in a distributed and autonomous fashion [14], [15]. The superior efficiency of the MA-based network management has already been established [16]. Advantages of MA based network monitoring are less delays and less bandwidth utilization in observing the network traffic and higher level data (relevant information) communicating to the manager. The functionality of MAs can be easily modified or added, by the manager, by adding modified software code which can be easily send from the SA. The fault can be easily detected using mobile agents by analyzing the devices connected to a network.

The work in [18] analyzed the characteristics of four design paradigms, client/server, code on demand, remote evaluation and mobile agent, and calculated their performances with respect to network traffic, response time and energy consumption in ubiquitous computing environment, and showed the superior performance of the mobile agent over other service design paradigm. The rate of increase of network traffic and response time in case of MA is slowest than the other service design paradigm when the number of node count increases.

Mobile agents are useful tools to be used in mobile environment [19], [20]. The mobile agents are secure, as the mobile agent platform provides encryption, code integrity for transmission, code integrity for execution, and authentication and trust. Mobile agent platforms like JADE and Voyager use SSL connection which assure data flow encryption, and to detect any attempt to tamper with an agent. In mobile agent platform code integrity is verified before starting its execution. Danny B. Lange in [21] describes the role of mobile agents in distributed computing. The important characteristics of mobile agents [22], [23], [24], [25], which motivated us to use it for monitoring are asynchronous autonomous interaction, robustness and fault tolerance and support for heterogeneous [26].

4 UNODE MONITORING PROTOCOL

In this section, we describe the proposed Unode monitoring protocol, which collects health condition of Unodes of a ubiquitous application in different subnetworks of ubiquitous network using agents.

The principle of monitoring protocol is solely based on two types of agents: Static Agent (SA) and Mobile Agent (MA). The SA resides at the central monitoring location, i.e., where the main ubiquitous application is activated. This not only monitors the health conditions of the Unodes but also generates and dispatches MAs to the required subnetwork. The SA uses the past migration information of the predecessor to decide the subnetwork to which Unode might migrate. MA runs on the agent platform in a node (which is rich in resources) in the subnetwork where Unode is about to migrate.

Keeping a static agent in each and every subnetwork might not be a good approach as they consume node resources in which they reside even if there is no migration of a Unode to the subnetwork. As in ubiquitous computing environment, the nodes are usually mobile, and have less computational resources compared to the static nodes; using static agent might exhaust their resources quickly.

4.1 The Protocol Architecture

The monitoring protocol designed with two segments: 1. Central monitoring system: The main segment which runs at the subnetwork where the main ubiquitous application is initiated, for example, in a ubiquitous museum guide system it might reside at the central ubiquitous system where the Unode (or the user) registers first to avail the ubiquitous museum guide services, and 2. Mobile Monitoring system: It is a minor segment of the system which runs in the migrated MA. The MA indeed runs on a node, which is rich in resources, in the subnetwork, and migrates (if required) to another node (in the same subnetwork) if the resources of the node where MA is residing are heavily used. Contract Net Protocol and the Dutch Auction Protocol are the agent interaction protocols standardized by FIPA (Foundation of Intelligent Physical Agents), which can be used between FIPA complaint agents for communication.

Both SA and MA monitor the Unode health conditions in coordination with LA. In the absence of agent platform in the subnetwork, the MA's can communicate with the LA of different subnetwork by using remote procedure call (RPC) mechanism. MA uses the appropriate network management protocol, like SNMP (in data network) or CMIP (in telecommunication network) to interact with LA.

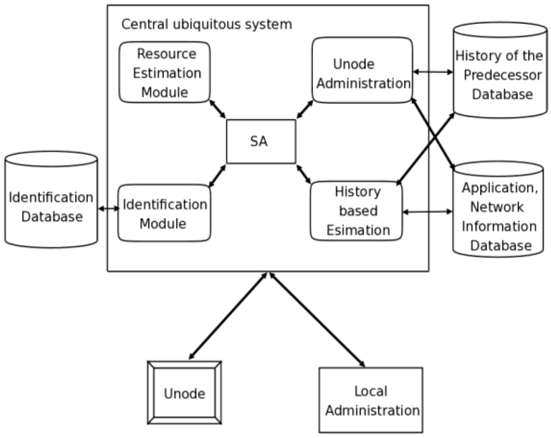

Figure 1: Architecture of the main segment of the monitoring protocol which runs at the subnetwork where ubiquitous application is initiated

Architecture of the main segment of the monitoring protocol of a ubiquitous network is shown in Fig. 1. The central subnetwork could be one of wireless or wired segment where ubiquitous application has been started.

We discuss the functioning of the main segment of the protocol by describing its components.

4.1.1 Identification Module

The function of this module is to verify the authenticity of a Unode. When a Unode register to avail a service (as discussed earlier in the context of ubqitous museum guide system), the central ubiquitous system assigns a security certificate to the Unode. When the Unode goes to the subnetwork where the ubiquitous application is initiated, Unode sends the security certificate to the central ubiquitous system over the subnetwork technology. When SA (which is in the central ubiquitous system) receives the security certificate, it forwards that to the Identification module. This checks the security certificate provided by Unode with the existing authentication information in the identification database and communicates the same to SA.

4.1.2 History based Estimation Module

It analyzes the past behavior of the application and network, and history of the Unodes. The history helps SA to take appropriate decision during the time of emergency situation like unavailability of resources at the LA, and to predict the migration of the Unode based on the previous node history. From the history of the predecessor, SA can predict the possible network to which the Unode may migrate. The history of the behavior of the Unode also helps SA to know about the Unode resource requirement.

We define, σ_i, as the parameter which shows the percentage of resources used by the predecessors, and which will be used by the resource estimation module to estimate the required resources to provide the service,

$$\sigma_i = \frac{1}{N}\sum_{k=1}^{N}\frac{R_{i,k}(\max) - R_{i,k}(allocated)}{R_{i,k}(\max) - R_{i,k}(\min)} \quad (1)$$

where N represents the number of predecessors who are running the same application, $R_{i,k}(\min)$ and $R_{i,k}(\max)$ are the minimum and maximum i_{th} resource requirement of k_{th} Unode, and $R_{i,k}(allocated)$ represents the amount of i_{th} resource allocated to the k_{th} Unode.

4.1.3 Required Resource Estimation Module

It provides the required resource information to SA based on the task to be performed. The function of the resource estimation module is to analyze the personalized information of the user and the context information, the resource requirement for the service, etc., and decide/predict the resource requirement for the provision of the service. The resources are estimated based on average predecessors resource utilization, type of QoS required to provide the service and the current resources used by Unode.

We estimate the lowest and highest i_{th} resource requirement of j_{th} Unode, $R_{i,j}(\min)$ and $R_{i,j}(\max)$ respectively, by taking the weighted average of the lowest and highest resource requirement of the predecessor nodes, $R_{i,j}(avg,\min)$ and $R_{i,j}(avg,\max)$ respectively, and the minimum and maximum resource requirement of the j_{th} Unode in the previous subnetwork, $R_{i,j}(pre,\min)$ and $R_{i,j}(pre,\max)$ respectively.

$$R_{i,j}(\min) = \frac{R_{i,j}(pre,\min) + R_{i,j}(avg,\min)}{1+\delta} \quad (2)$$

$$R_{i,j}(\max) = \frac{R_{i,j}(pre,\max) + R_{i,j}(avg,\max)}{1+\delta} \quad (3)$$

where

$$R_{i,j}(avg,\min) = \frac{1}{N}\sum_{k=1}^{N} R_{i,k}(\min)(\sigma_i) \quad (4)$$

$$R_{i,j}(avg,\max) = \frac{1}{N}\sum_{k=1}^{N} R_{i,k}(\max)(\sigma_i) \quad (5)$$

$$\delta = \begin{cases} 0 & \textit{for the first network} \\ 1 & \textit{for other networks} \end{cases}$$

Algorithm 1 shows the proposed algorithm of the Resource Estimation module.

Algorithm 1 Resource Estimation module

1: **Begin**

2: SA informs Resource Estimation module to give the resource requirement of Unode

3: History Estimation module calculate σ_i

4: SA collects σ_i, $R_{i,j}(pre, \min)$, $R_{i,j}(pre, \max)$ and history of predecessor resource utilization and provides to the Resource Estimation module

6: the Resource Estimation module calculates $R_{i,j}(\min)$ and $R_{i,j}(\max)$ and informs to SA

5: **End**

4.1.4 Unode Administration

This module monitors the Unode activities by collecting and analyzing the network information provided by the SA (directly) or MA (indirectly, through SA). MA informs the Unode administration module, via SA, the network information periodically, like the resources that have been provided by LA to Unode. The Unode administration module monitors the activities of the Unode, and informs the SA to take some preventive steps if any symptoms of a fault are notified. Unode administration module stores all the analyzed data in the history of the predecessor database and in the application and network information database.

4.1.5 Static Agent (SA)

It has significant role to play in monitoring all the Unodes wherever they are migrating. The main objective of using SA is to coordinate among different modules of the monotoring system at the central ubqitous system, and create/destroy MAs as and when required.

The functions of a static agent at the central ubiquitous system are:

- It collects the identification information from the Unode.
- It verifies the identity of a Unode with the help of the identification module.
- It contacts resource estimation module seeking for the resource information to allocate network resources to the Unode.
- It collects and analyzes the network resource information from the resource allocation module.
- It contacts LA to get/negotiate the required network resources for the Unode.
- It provides the network resource status information to the Unode.
- It monitors the resource utilization of the Unode by interacting with LA.
- It informs the Unode administration module about the resource requirement of the Unode and the network resource status infomation.
- If the LA does not have required resources, it checks the history of the application, and takes effective actions like seeking less resources from the LA if the application runs for a short period.
- It predicts the Unode migration based on the Received Signal Strength Indicator (RSSI)-from the LA, and the migration history of predecessor.
- It generates and dispatches the MA to the next subnetwork where the Unode is about to migrate.
- It interacts with the MA to get the current status of the Unode(s).
- It informs respective MA about the updated resource requirements.
- It informs the monitored information received from the mobile agent present in other subnetwork to the Unode administration module.

4.1.6 Mobile Agent (MA)

The MA (see Fig. 2) has two modules: monitoring module and interface module, that interacts with LA. MA is created by SA in anticipation of a Unode migrating to a subnetwork segment of the Ubiquitous network.

The functions of the mobile agents at the other subnetworks of a ubiquitous network are:

- It migrates to the subnetwork carrying the relevant required resource information about the Unode.
- It communicates with LA of the subnetwork segment to reserve network resources for the migrating Unode.
- It informs the Unode regarding the reserved network resources as soon as the Unode enters into the subnetwork.

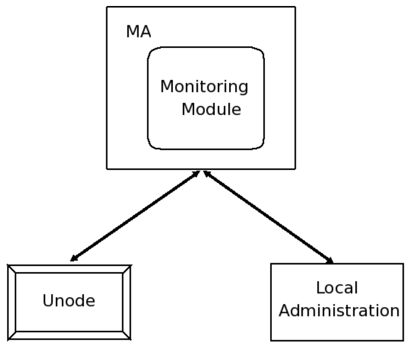

Figure 2: Mobile monitoring system segment of Unode monitoring protocol

- It informs the network condition to the SA present at the central ubiquitous system.
- It monitors resource utilization of the Unode by collecting the relevant information from LA.
- It analyzes resource information of the Unode administration using the monitoring module.
- If it finds LA is not providing the promised resources it accepts resources for the Unode in an incremental manner.
- It predicts the Unode migration to the next subnetwork and reports to the SA for migration preparation.

4.1.7 Monitoring Module

This (see Fig. 3) is a part of the MA, whose functions are to collect the network resource information provided by MA, to analyze those, and to inform MA about the noticed abnormalities. The resource status collects the network information from the MA and provides to the resource analyzer. The resource analyzer checks whether the Unode is getting the promised resources from the LA. If it finds that LA is unable to provide the required resources it informs the resource information reporter to inform the same to MA. Algorithm 2 shows the proposed algorithm of the Monitoring module.

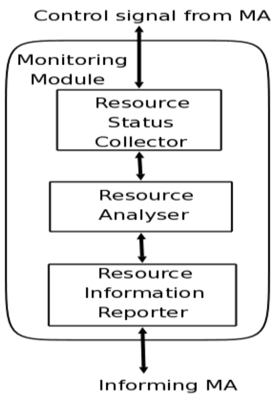

Figure 3: Monitoring module architecture

Algorithm 2 Monitoring Module

Begin
MA provides information from LA to Resource Status
if Resource_Status: OK **then**
  Resource_Analyser: % of resources allocated to the Unode
  Resource_Reporter:<OK, % of resources>
if Resource_Status: Not satisfactory **then**
  Resource_Analyser:% of resource allocated
  Resource_Analyser:% of resource not allocated
  Resource_Reporter:<Not satisfactory, % of resource provided, and % of resources not provided>
end if
End

4.2 Case Study

Fig. 5 shows a case study of proposed monitoring system. We consider three different subnetworks: GSM/GPRS, WLAN, and Bluetooth in a ubiquitous network. Initially when a Unode enters into the GSM/GPRS, the SA present in the subnetwork verifies the authenticity of the node. SA gets the information of the required resources from the resource estimation module, negotiates with LA, i.e., the GSM/GPRS service provider, to allocate/provides the required resources to the Unode, and monitors the activities of the Unode continuously by gathering network resource utilization information about Unode from LA. This process continues till the Unode migrates into another subnetwork, i.e., the WLAN network. When SA predicts that the Unode is about to migrate to the WLAN subnetwork it generates and dispatches a MA to monitor the Unode.

From the history of the Unode in the WLAN, the SA provides the resource requirement information to the MA, and dispatches it to the WLAN subnetwork before actual migration takes place.

The main objective of the MA is to negotiate with the LA to reserve the required resources for the Unode and to monitor the healthy condition of the networks, using the monitoring module.

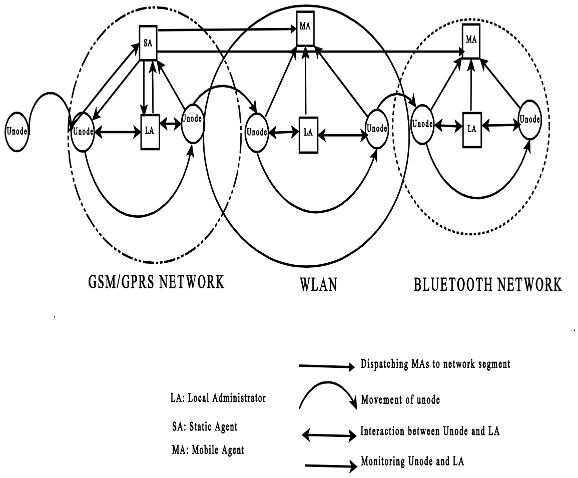

Figure 4: Scenario of a case study

In the WLAN subnetwork MA contacts and informs the Unode that it (MA) is the representative of the SA, and provides resources from the LA, i.e., the WLAN access point. Then, the Unode and the LA communicate with each other and MA collects the network resources utilized by the Unode from LA. Similar activities take place when Unode moves from WLAN subnetwork to the Bluetooth subnetwork. Fig. 5 shows the sequence diagram of the node monitoring protocol in ubiquitous network.

The next section explains the mathematical modeling of the proposed ubiquitous node monitoring system to study the behavior of the system by analyzing the steady state information of the system.

5 ANALYTICAL MODEL

We have built the analytical model of the node monitoring protocol using Markov chain concept. Based on the model, we analyze the system at the steady state by giving mathematical representation of the steady state.

5.1 Analysis of Mobile Monitoring System Segment

The subsystem in mobile agent monitors the healthy

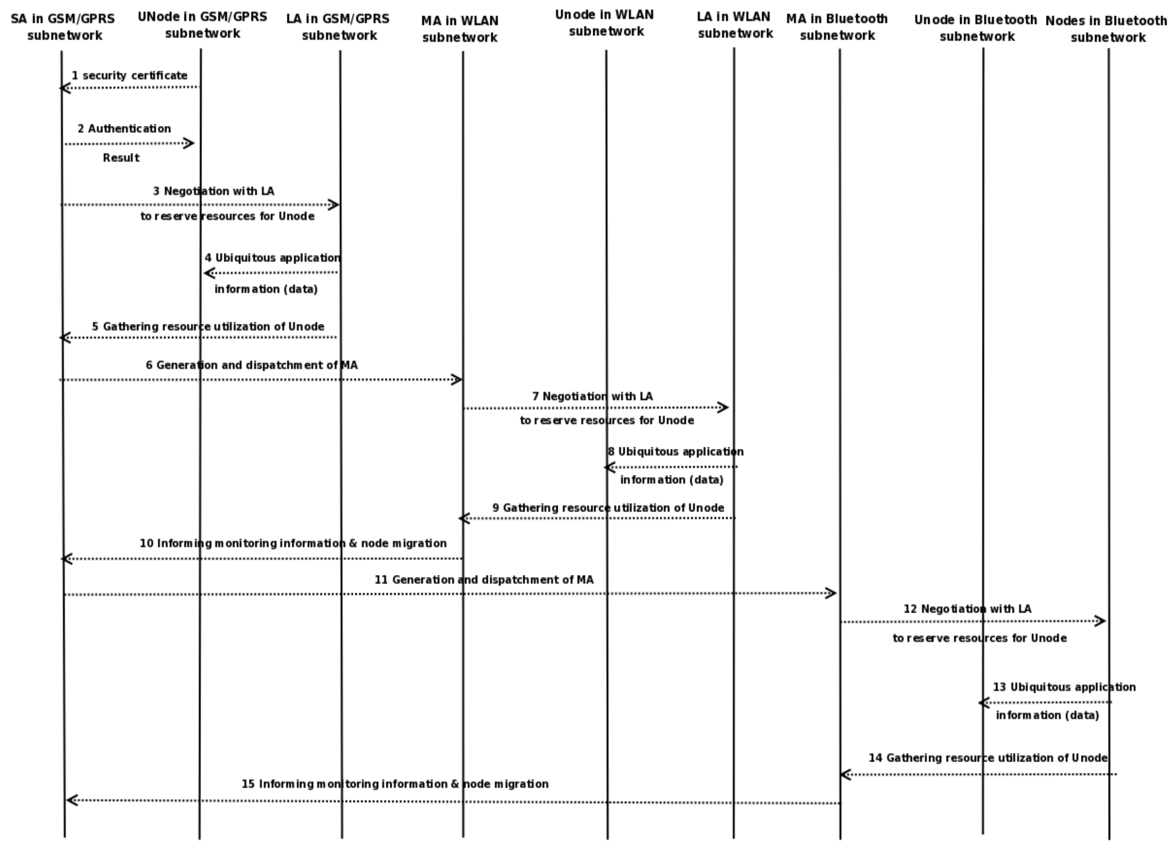

Figure 5: Sequence diagram of the node monitoring protocol in the ubiquitous network

conditions and identifies the activities like getting the required information, producing the effective results and identifying required local resources like bandwidth, buffer and processing time. We consider the discrete time Markov model for system analysis. The Markov chain of the subsystem is shown in the Fig 6.

At any point of time, the system can be any of the below mentioned states. The states are

S_1 = Initial or idle

S_2 = Identification of a Unode

S_3 = Monitoring the Unode

$S_{i,j}$ = i resource degradation state of the j Unodes

Where it shows that j numbers of Unodes are not getting the promised i resources

S_4 = Finish

Where $i = 1,2,\cdots,m$ and m is the number of different resources, and $j = 1,2,\cdots,n$ and n is the number of Unodes.

The MA keeps on observing and monitoring the resources of the Unodes. The occurrence of the events like the throughput degradation and energy degradation are random. The S_3 state of the Markov chain represent the monitoring state. At this state the monitoring subsystem, i.e., the MA, inquiries about each Unodes promised resources which includes checking whether the local administration provided sufficient resources or not by checking whether the Unode is getting the promised resources or not by analyzing the throughput, packet loss, energy and buffer degradation. The main objective of the monitoring system is to monitor that the Unode is getting the promised resources or not and report the same to the Ubiquitous central system where SA resides. The MA monitors the Unodes one by one to check the resources. While monitoring if MA finds that some event such as throughput degradation of that particular Unode happened it reports this to central ubiquitous system. In Markov chain, we can show this by the transition from the monitoring state (S_3) to the respective states, i.e., $S_{4,1}$ when the Unode don't get the promised resources, and $S_{5,1}$ when the throughput of the Unode degraded. When MA checks the next Unode, and finds that it is facing similar problem it informs the SA. In the Markov chain it is shown in the transition from the $S_{4,1}$ to $S_{4,2}$ if the second Unode also does not get promised resources. Similarly, the one-step transition probabilities are:

$P\{S_1 \mid S_1\} = 1 - q$; $P\{S_2 \mid S_1\} = q$;

$P\{S_1 \mid S_2\} = 1 - P_{id}$; $P\{S_3 \mid S_2\} = p_{id}$;

$P\{S_3 \mid S_3\} = 1 - p_{mo}$; $P\{S_{i,1} \mid S_3\} = p_i$;

$P\{S_{i,j} \mid S_{i,j-1}\} = p_i$; $P\{S_{i,n} \mid S_{i,n-1}\} = p_i$;

$$P\{S_{i,n} \mid S_{i,n}\} = p_i\,;\ P\{S_{i,n-1} \mid S_{i,n}\} = 1 - p_i\,;$$
$$P\{S_{i,j-1} \mid S_{i,j}\} = 1 - p_i\,;\ P\{S_3 \mid S_{i,1}\} = 1 - p_i\,;$$
$$P\{S_4 \mid S_3\} = p_s\,; P\{S_1 \mid S_4\} = 1 \qquad (6)$$

The subsystem starts with the ideal state (S_1). The q shows the probability that the new arrivals or new Unodes come or enter into the subnetwork. The p_{id} is the probability of the successful identification of the Unodes. After the monitoring system identifies the Unode, it goes into the monitoring system. Where it monitors the behavior of the Unode by observing resource related information. While monitoring, the system can enter into any state.

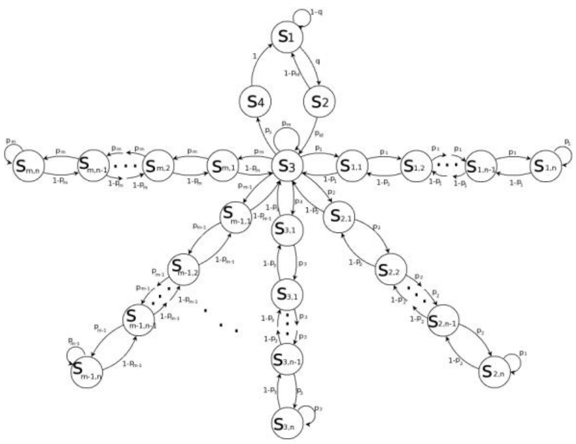

Figure 6: Markov chain of the mobile monitoring system segment

5.1.1 Multiple Nodes

Here we consider more than one Unode exist in a network segment. The steady state probabilities of the Markov chain with n number of Unodes are calculated. They are:

$$\pi_1 = [1 + q + qp_{id} + \frac{qp_{id}}{p_s}\{1 + \sum_{i=1}^{m}\sum_{j=1}^{n}(\frac{p_i}{1-p_i})^j\}]^{-1}$$

$$\pi_2 = q\pi_1\,;\ \ \pi_3 = \frac{qp_{id}}{p_s}\pi_1$$

$$\pi_{i,j} = (\frac{p_i}{1-p_i})^j\pi_3\,; \pi_4 = p_s\pi_3 \qquad (7)$$

5.1.2 Single Node

Consider a case of single Unode migrated into a subnetwork. The steady state probabilities of the Markov chain with n number of Unodes are calculated. Those are:

$$\pi_1 = [1 + q + qp_{id} + \frac{qp_{id}}{p_s}\{1 + \sum_{i=1}^{m}(\frac{p_i}{1-p_i})\}]^{-1}$$

$$\pi_2 = q\pi_1\,;\ \pi_3 = \frac{qp_{id}}{p_s}\pi_1$$

$$\pi_{i,1} = (\frac{p_i}{1-p_i})\pi_3\,; \pi_4 = p_s\pi_3 \qquad (8)$$

5.2 Analysis of Node Monitoring System

In this work, we developed an analytical model of the proposed node monitoring protocol in ubiquitous computing environment. At any point of time, the system can be any of the below mentioned states. The states are:

S_1 = Initial or idle

S_2 = Identification of a Unode

S_3 = Seeking resources from the LA (by SA)

S_4 = Obtaining resources from the LA (by SA)

S_5 = Allocating resources from the LA (by SA)

S_6 = Monitoring the Unode and predicting its migration (by SA)

S_7 = Generation and dispatching of MA (by SA)

S_8 = Seeking resources from the LA (by MA)

S_9 = Obtaining resources from the LA (by MA)

S_{10} = Allocating resources from the LA (by MA)

S_{11}= Monitoring the Unode and predicting its migration (by MA)

S_{12} = Finish

Fig. 7 shows the Markov model of the system monitoring protocol.

In the Markov chain, the one-step transition probabilities are:

$$P\{S_1 \mid S_1\} = 1 - q\,;\ P\{S_2 \mid S_1\} = q\,;$$
$$P\{S_1 \mid S_2\} = 1 - p_i\,;\ P\{S_3 \mid S_2\} = p_i\,;$$
$$P\{S_3 \mid S_3\} = 1 - p_{ss}\,;\ P\{S_4 \mid S_3\} = p_{ss}\,;$$
$$P\{S_3 \mid S_4\} = 1 - p_{rs}\,;\ P\{S_5 \mid S_4\} = p_{rs}\,;$$
$$P\{S_5 \mid S_5\} = 1 - p_{as}\,; P\{S_6 \mid S_5\} = p_{as}\,;$$
$$p\{S_6 \mid S_6\} = p_{ms}\,; P\{S_{12} \mid S_6\} = p_s\,;$$
$$P\{S_7 \mid S_6\} = p_{ns}\,; P\{S_7 \mid S_7\} = p_{ngd}\,;$$
$$P\{S_8 \mid S_7\} = p_{gd}\,; P\{S_8 \mid S_8\} = 1 - p_{sm}\,;$$
$$P\{S_9 \mid S_8\} = p_{sm}\,; P\{S_8 \mid S_9\} = 1 - p_{rm}\,;$$
$$P\{S_{10} \mid S_9\} = p_{rm}\,; P\{S_{10} \mid S_{10}\} = 1 - p_{am}\,;$$
$$P\{S_{11} \mid S_{10}\} = p_{am}\,; P\{S_{11} \mid S_{11}\} = p_{mns}\,;$$
$$P\{S_{12} \mid S_{11}\} = p_{sns}\,; P\{S_1 \mid S_{12}\} = 1 \qquad (9)$$

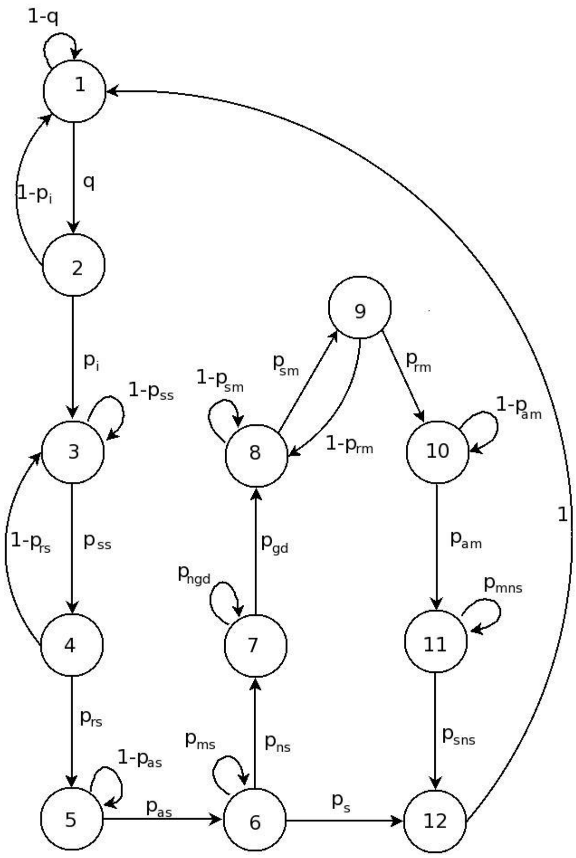

Figure 7: Markov model of the system monitoring protocol

The Markov model is ergodic and the steady state probabilities are:

$$\pi_2 = \frac{den_1 den_2}{(1+q+qp_i)den_1 den_2 + qp_i(num_1 den_2 + num_2 den_1)}$$

& where,

$$num_1 = p_{as} + p_{as}p_{ss} + p_{rs}p_{ss}$$

$$den_1 = p_{as}p_{rs}p_{ss}$$

$$num_2 = p_{sm}p_{rm}p_{am}(1-p_{ngd})(1-p_{mns}) +$$
$$p_{ns}p_{sm}p_{rm}p_{am}(1-p_{mns}) +$$
$$p_{ns}p_{am}(1-p_{ngd})(1-p_{mns}) +$$
$$p_{ns}p_{sm}p_{am}(1-p_{ngd}) +$$
$$p_{ns}p_{sm}p_{rm}(1-p_{ngd})(1-p_{mns}) +$$
$$p_{ns}p_{sm}p_{rm}p_{am}(1-p_{ngd})$$

$$den_2 = p_{sm}p_{rm}p_{am}(1-p_{ms})$$
$$(1-p_{ngd})(1-p_{mns})$$

$$\pi_1 = \frac{\pi_2}{q};$$

$$\pi_3 = \frac{p_i}{p_{ss}}\pi_2 \;;$$

$$\pi_4 = \frac{p_i}{p_{rs}}\pi_2\,; \pi_5 = \frac{p_i}{p_{as}}\pi_2\,;$$

$$\pi_6 = \frac{p_i}{1-p_{ms}}\pi_2$$

$$\pi_7 = \frac{p_i p_{ns}}{(1-p_{ngd})(1-p_{ms})}\pi_2$$

$$\pi_8 = \frac{p_i p_{ns}}{p_{sm}p_{rm}(1-p_{ms})}\pi_2$$

$$\pi_9 = \frac{p_i p_{ns}}{p_{rm}(1-p_{ms})}\pi_2$$

$$\pi_{10} = \frac{p_i p_{ns}}{p_{am}(1-p_{ms})}\pi_2$$

$$\pi_{11} = \frac{p_i p_{ns}}{(1-p_{ms})(1-p_{mns})}\pi_2$$

$$\pi_{12} = p_i \pi_2 \quad (10)$$

6 SIMULATION

In the simulation we have considered three different subnetworks based on GSM/GPRS, WLAN, and Bluetooth technologies as the segments of a ubiquitous network. A ubiquitous application has been activated in the WLAN with 20 Unodes. These Unodes travel in directions as per the application specified and need the local resources to execute the assigned job. We assumed that the Unodes works in these three different networks segments. Initially, Unodes start in the WLAN network segment and send the identification information to the SA. After SA successfully verifies the authenticity of Unodes, it contacts the resource estimation module for the required resources and negotiates WLAN access point to provide resources to the Unode. Here, our assumption is WLAN has the required resources most of the time. We considered buffer space and the throughput are the required resources to be monitored. The Unode then receives the data from the WLAN access point. SA monitors the resource utilization of the Unodes by interacting with LA. After some random period of time, the Unode migrates to the next network segment, i.e., GSM/GPRS subnetwork. As the SA is aware of the migration it generates and dispatches the mobile agent, MA1, to the GSM/GPRS subnetwork. That carries out monitoring process as mention in the previous section. Then Unode moves to the Bluetooth network segment, where it finds the MA2 sent by SA, which (MA2) helps the Unode to get access to the Bluetooth resources, and monitors the resource utilization of the Unode. Algorithm 3 shows the proposed algorithm of the node monitoring protocol.

Algorithm 3 Ubiquitous node monitoring protocol

Begin
SA verifies the authenticity of the Unode
if Authenticated Unode **then**
  SA checks Unode resource requirement
  if LA has required resource **then**
    Unode and LA communicate
    SA collects resource information from LA
    if Unode is about to migrate **then**
      SA Creates a MA and dispatches
      MA seeks the required resources from the LA
      Unode and LA communicate
      MA collects resource information from LA
    end if
  if LA does not have required resources **then**
    MA reports failure to SA
    SA finds required resources from history
    SA uses history analyzer to find the application duration.
    if duration is less **then**
      SA reduces the resource requirement
    end if
  end if
else
  Unauthenticated Unode
end if
End

7 RESULTS

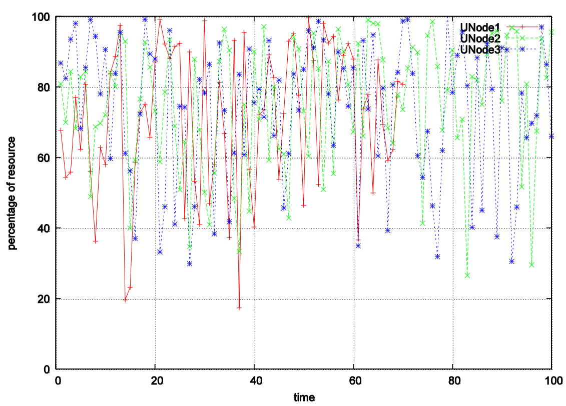

Figure 8: Comparison of the moisture absorption from a free surface by the experimental lime/cement plaster and the restoration (salt transporting) plaster

We have considered a typical scenario where the Unodes start at a WLAN subnetwork, i.e., the main segment of the monitoring protocol, then it migrates into different subnetworks like GSM/GPRS subnetwork and the Bluetooth subnetwork. The time duration is in different networks are considered randomly. Data are transmitted in packets with three different sizes. Initially, while downloading the data the Unode informs the SA about the resource (throughput). As the Unode moves to next sub-networks it provides resource information to the MA. Fig. 8 shows the performance of the ubiquitous monitoring protocol system. By performance we mean to say that what percentage of required resource was provided to the Unodes by the local administration by the help of ubiquitous node monitoring protocol.

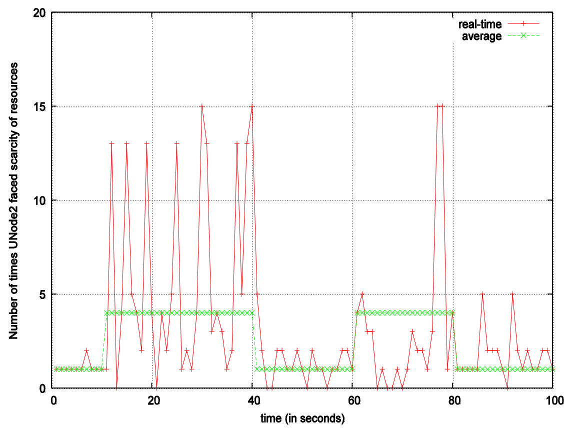

Figure 9: Number of times the UNode2 faced the resource scarcity problem

The number of times the UNode2 faced network resource unavailability are shown in Fig. 9. The big peaks in the figure show that when the UNode2 was Results of the three different Unodes in a ubiquitous network are shown in the Fig. 8. For example, the UNode2 started at the WLAN1 subnetwork where it stayed for 10 sec and after that it migrated in to the GSM/GPRS subnetwork. After spending 30 sec in the GSM/GPRS subnetwork it entered into the BT3 (Bluetooth 3) where it received the data for 20 sec. At last, it migrated to the BT4, through the GSM/GPRS subnetwork, it spent 20 sec inside the GSM/GPRS network, where it stopped the application after 20 sec. in the GSM/GPRS it faced the scarcity of the resources several times compared to the Bluetooth and WLAN subnetworks.

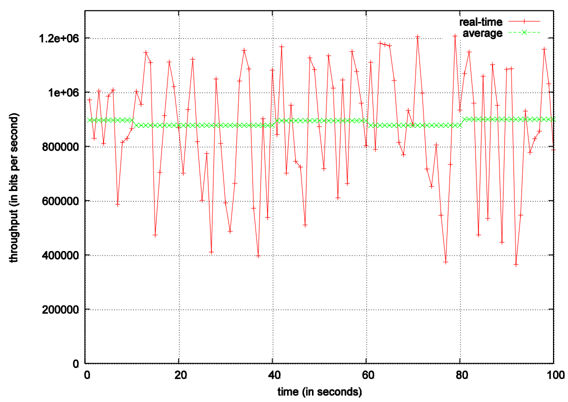

Figure 10: Throughput of the Unode2 in the ubiquitous network

The throughput, and number of packets received by UNode2, which is monitored by different mobile agents and reported to the main segment of the system, are shown in Fig. 10 and Fig. 11. The deep valleys shown in these two figures represent that the decrease in both in the throughput and the number of packets received because of the scarcity of available resources in the GSM/GPRS subnetwork.

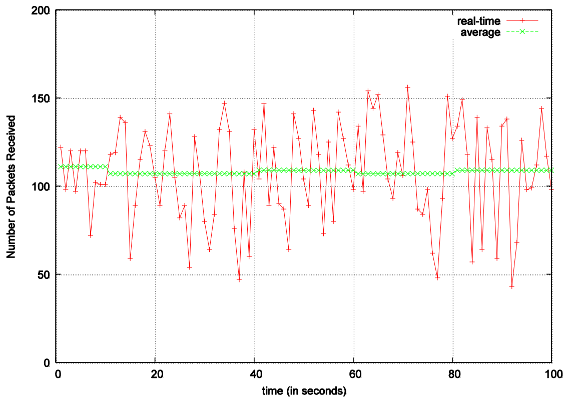

Figure 11: Number of packets received by UNode2 in ubiquitous network

8 CONCLUSION

In this paper, we have presented a novel method to monitor a Unode in a ubiquitous network. We have used mobile agents to monitor the healthy condition of a Unode in a ubiquitous network. The past behavior of the application and network, and history of the Unode and the predecessor are taken into consideration to help SA to take proper decision during the time of emergency situation like unavailability of resources at the local administration, and to predict the migration of the Unode based on the previous node history. The results obtained in the simulation reflect the effectiveness of the technique.

9 REFERENCES

[1] J. W. N. et al., Ubiquitous Monitoring Environment for Wearable and Implantable Sensors (UbiMon), pp.3-4, (2004).

[2] V. L. Kristof et al., "Medical healthcare monitoring with wearable and implantable sensors," in Proc. of the 3rd International Workshop on Ubiquitous Computing for Healthcare Applications, (2004).

[3] V. Jones and J. H. Jo, "Ubiquitous learning environment: An adaptive teaching system using ubiquitous technology," (2004).

[4] T. Murakami, "Establishing the ubiquitous network environment in japan," (2003).

[5] S. Rao, " Operational fault detection in cellular wireless base-stations," Network and Service Management, IEEE Transactions on, vol. 3, no. 2,pp. 1-11, april (2006).

[6] S. Stanic et al., "Acive monitoring and alarm management for fault localization in transparent all-optical networks," Network and Service Management, IEEE Transactions on, vol. 7, no.2, pp. 118-131, june (2010).

[7] A. Asgari, R. Egan, P. Trimintzios, and G. Pavlou, "Scalable monitoring support for resource management and service assurance," Network, IEEE, vol. 18, no. 6, pp. 6-18, nov-dec, (2004).

[8] S. M. Lee and H. Y. Youn, "Dynamic window-based adaptive fault monitoring for ubiquitous computing systems," in Proceeding of the 11th Pacific Rim International Symposium on Dependable Computing, ser. PREC '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 358-368, (2005).

[9] K. Kang, J. Song, J.Kim, H. Park, and W. D. Cho, "Uss monitor: A monitoring system for collaborative ubiquitous computing environment," Consumer Electronics, IEEE Transactions on, vol. 53, no.3, pp. 911-916, aug (2007).

[10] J.-H. Bae, K.-O. Lee, and Y.-Y.Park,"Moneta: an embedded monitoring system for ubiquitous network environments," Consumer Electronics, IEEE Transactions on, vol. 52, no. 2, pp. 414-420, may (2006).

[11] P. Domingues, L. Silva, and J. Silva, "Dronitor-a distributed resource monitoring system," in Parallel, Distributed and Network-Based Processing, 2003, Proceedings. Eleventh Euromicro Conference on, pp. 127-133, feb (2003).

[12] P. Dinda et al., “The architecture of the remos system,” in High Performance Distributed Computing, 2001. Proceedings. 10th IEEE International Symposium on, pp. 252 –265, (2001).

[13] B.-H. Lee, S.-H. Lim, and J.-H. Kim, “Scalable real-time monitoring system for ubiquitous smart space.” Inf. Process. Lett, pp. 294–299, (2010).

[14] D. Gavalas, G. E. Tsekouras, and C. Anagnostopoulos, “A mobile agent platform for distributed network and systems management,” J. Syst. Softw., vol. 82, no. 2, pp. 355–371, Feb. (2009).

[15] D. Gavalas, D. Greenwood, M. Ghanbari, and M. O’Mahony, “Using mobile agents for distributed network performance management,” in 3rd International Workshop on Intelligent Agents for Telecommunication Applications (IATA’99), (1999).

[16] H. H. To, S. Krishnaswamy, and B. Srinivasan, “Mobile agents for network management: when and when not!” in Proceedings of the 2005 ACM symposium on Applied computing, ser. SAC ’05. New York, NY, USA: ACM, pp. 47–53, (2005).

[17] M. Kona and C.-Z. Xu, “A framework for network management using mobile agents,” in Proceedings of the 16th International Parallel

and Distributed Processing Symposium, ser. IPDPS ’02. Washington, DC, USA: IEEE Computer Society, pp. 7–14, (2002).
[18] S. Han and Y. Zhang, “Performance evaluation of mobile service design paradigm in ubiquitous computing environments,” in Proceedings of the 2009 Fourth International Conference on Frontier of Computer Science and Technology, ser. FCST ’09. Washington, DC, USA: IEEE Computer Society, pp. 89–95, (2009).
[19] O. Urra, S. Ilarri, R. Trillo, and E. Mena, “Mobile agents and mobile devices: Friendship or difcult relationship?” Journal of Physical Agents, vol. 3, no. 2, (2009).
[20] C. Spyrou and G. Samaras, “Mobile agents for wireless computing: the convergence of wireless computational models with mobile-agent technologies,” Mobile Networks and Applications, vol. 9, pp. 517–528, (2004).
[21] D. B. Lange, “Mobile objects and mobile agents: The future of distributed computing?” in ECOOP, pp. 1–12, (1998).
[22] A. Bieszczad, B. Pagurek, and T. White, “Mobile agents for network management,” Communications Surveys Tutorials, IEEE, vol. 1, no. 1, pp. 2–9, quarter (1998).
[23] C. Bohoris, G. Pavlou, and H. Cruickshank, “Using mobile agents for network performance management,” in Network Operations and Management Symposium, 2000. NOMS 2000. 2000 IEEE/IFIP, pp. 637–652, (2000).
[24] I. Satoh, “Building reusable mobile agents for network management,” Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 33, no. 3, pp. 350 – 357, aug. (2003).
[25] I. Adhic, C. Pattinson, and E. Shaghouei, “Using mobile agents to improve performance of network management operations,” (2003).
[26] C. Bohoris, G. Pavlou, and A. Liotta, “A hybrid approach to network performance monitoring based on mobile agents and corba,” in MATA, pp. 151–162, (2002).