HYWINMARC: An Autonomic Management Architecture for Hybrid Wireless Networks

Shafique Ahmad Chaudhry, Ali Hammad Akbar, Ki-Hyung Kim[1],
Suk-Kyo Hong, and Won-Sik Yoon

Graduate School of Information and Communication
Ajou University, Suwon, Korea
{shafique, hammad, kkim86,skhong,wsyoon} @ ajou.ac.kr

Abstract. The envisioned realization of ubiquity has resulted into the emergence of new kinds of the hybrid networks. The modern hybrid networks, e.g. combination of wireless mesh and Mobile Ad-hoc Networks (MANETs), help realize ubiquity through spontaneous networking. The network management for these hybrid networks is different from conventional and infrastructure based network management. Heterogeneity, mobility, dynamic topologies, physical security, and survivability make the challenge hard. A new class of management called self-management can effectively be used to cater for the autonomous behavior of hybrid networks**.** We present HYbrid WIreless Network Management ARChitecture (HYWINMARC), a three-tier framework, covering all the management levels, for autonomic network management for hybrid networks. We integrate policy-based network management with mobile-agent technology and design a prototype for a context-aware and self-managing architecture. The context information is collected, from all levels in network hierarchy through monitoring agents, and is used to apply needed self-management operations that include self-optimization, self-healing, self-configuration, and self-growing.

1 Introduction

Ubiquitous or pervasive computing means to embed the computing into environment. One of the main aspirations for ubiquitous computing is to enable the devices to sense changes and adapt accordingly. The envisioned realization of ubiquity has resulted into the emergence of new kinds of the hybrid networks [1] [2], i.e. result of integration of the different network technologies, demands for new paradigms for network management. The u-Zone Network is a hybrid of wireless mesh and MANETs. Unlike fixed wireless networks such as cellular networks or wireless local area networks (WLANs), mesh networks provide robust wireless connectivity to heterogeneous wireless devices and take less time to set up. Applications of mesh networks range from emergency services such as fire brigade network to intelligent transportation i.e., making car to car communication possible etc. Ad hoc networks, on the other hand, are formed by a group of wireless enabled devices that connect together and form a network, without the assistance of a pre-existing infrastructure. MANETs are characterized by heterogeneity, mobility, dynamic topologies, limited

[1] Corresponding author

physical security, and limited survivability. The applications of MANET include search and rescue operations, natural disaster recovery, the battlefield, spontaneous meetings and rendezvous between people of similar interests etc. We believe that a much more common use of multi-hop MANET concepts will emerge in mesh network configurations. In these scenarios, the ad hoc network is an extension of the existing fixed telecommunications infrastructure. Mesh networks are a suitable choice for such infrastructure to provide ubiquitous and robust connectivity to users. An example of such a setup is experimental metropolitan area networks or city wide mesh networks set in certain cities with wireless mesh routers on street lamps to provide seamless Internet connectivity to public.

To keep such networks always operational, a robust network management architecture is needed. There are many valuable efforts for network management of mesh networks [3] [4] as well of MANETs [5] [6]. All these works address their own domains, i.e., mesh or MANET and do not discuss the hybrid networks. The existing approaches cannot be implemented directly to the u-Zone networks due to the fact that u-Zone networks posses the characteristics of mesh networks as well as mobile ad-hoc networks. The catering of such challenges, plus the continuous growth factor, demands autonomous or self-management architecture which would support self-configuration, self-healing, self-security management, self-performance management, and self-accounting features. So far, the self-management architectures proposed are poised for wired networks especially for high end computing devices. A comprehensive architecture with granular architectural details, for a hybrid wireless network is still needed.

In this paper, considering the special nature of hybrid networks like u-Zone networks, we propose an architecture that is adaptable and robust to network variations, failures and changing user requirements. We present a three-tier, policy-based management architecture that uses the mobile-agent technology to deploy a robust self-managed framework.

The remainder of the paper is organized as follows. In section 2, we present related work which is followed by our proposed framework architecture in section 3. Self-management module is discussed in section 4. Section 5 is used to elaborate some proposed functionality. We conclude this paper with a summary in section 5.

2 Related Work

We state that autonomic network management of dynamic, on the fly, small scale ad hoc and mesh networks is in its infancy and there exists not much published work in this area. However, in this section we summarize the valuable efforts that relate to policy based and autonomic network management.

Ad-hoc Network Management Protocol (ANMP) is presented in [7] as an extension to Simple Network Management Protocol (SNMP) [8] has been presented. This work considers data collection, fault management, and security management as basic goals. As an extension of SNMP, it inherits certain limitations like difficult synchronization between SNMP manager and agent, difficult maintenance, limited support for third party RDBMS and polling. The Internet Engineering Task Force (IETF) [9] and the

Distributed Management Task Force (DMTF) [10] are currently working for the definition of standards for PBNM. The IETF has adopted the CIM (Common Information Model) [11] from DMTF to describe the network information. CIM is an implementation neutral schema for describing overall management information. The core model of CIM was extended to describe the policies to be applied in the PBM.

An intelligent agent-based framework that implements monitoring, configuration, and reporting policies through agents has been presented in [12]. Although this work covers most of the needs required by MANETs, it does not provide any provisions for hybrid networks. The k-hop clustering [13] has been one of the first of its kind policy-based QoS management frameworks for MANETs. Aside, it uses IETF policy framework and Common Open Policy Service (COPS) protocol. The Dynamic Re-Addressing and Management for the Army (DRAMA) [14] project, although primarily being a military application related project, explores the automation and distribution of policies and policy-decisions. Other known and accepted works apart from above mentioned ones are Guerrilla Management Architecture [15] and Management of Active Networks Based on Policies (MANBoP) [16]. We have summarized and compared the important features provided by these architectures with our architecture. Table 1 shows this comparative summary.

	K-hop Clustering	**DRAMA**	**Guerrilla Management**	**MANBoP**	**HYWINMARC**
Supported networks	MANET	MANET	MANET	Active and Programmable Networks	MANET and Wireless Mesh Networks
Tiers	2 tiers	3 tiers	3 tiers	2 tiers	3 tiers
Technology	IETF Policy based	Policy and Agent based	Agent based	Policy based	Policy and Agent based
Components	PDP and PEP	GPA, DPA and LPA	Nomadic Manager and Active Probe	MANBoP module	GMS, CM, GPM, MSR, DPM, SLA and ELA
Node Types	Simple Nodes (PEP)	Simple nodes (LPA)	Simple nodes (SNMP), Capable nodes (SNMP + Probe Processing Module)	Programmable nodes (Active nodes)	Simple nodes (SLA), Extended nodes (ELA)
Mobile Code	No	No	Yes (Active Probes)	No	Yes
Service Repository	No	No	No	Yes (code server)	Yes (MSR)

We realize that up to now there are not any well-known efforts that analyze the synergies that can be obtained from joining agent and policy-based network management technologies, for autonomic network management of hybrid networks.

3 Overall Architecture

The u-Zone Network is a hybrid of wireless mesh and MANETs. A wireless mesh network makes high-speed backbone whereas zero or more MANET clusters are attached to mesh nodes. Each mesh node, known as Zone-Master (ZM), is a multi-homed computer that has multiple wireless interfaces that make it capable of connecting to its peers as well as with MANET cluster(s). Each MANET cluster has a

cluster head (CH). As a ZM is connected to many peers, there are alternate paths to access wired network.

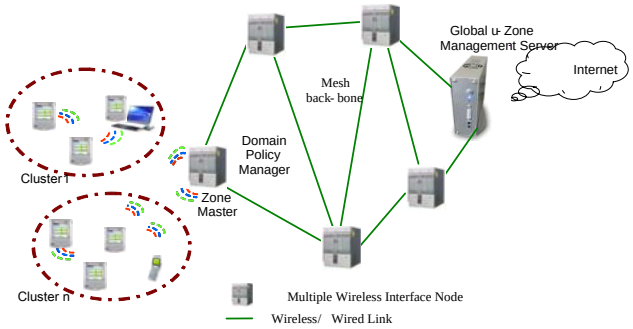

Figure 1 : The u-Zone Network

A hierarchical model of manager-agent configuration is followed to cover the whole network level, cluster level and node level management activities. At the whole network i.e., u-Zone level, Global u-Zone Management Server (GuMS), the central control entity, monitors the overall status of all u-Zone network elements. It provides an environment to specify the u-Zone level parameters through policies for ZMs' management. GuMS also manages context for the u-Zone and facilitates the mechanism to provide feedback control loop to achieve autonomic features. For cluster level management we have Domain Policy Manager (DMP) which performs the management operations with a scope limited to the cluster. At the node level we propose Simple Local Agents (SLAs) or SNMP agents and Extended Local Agents (ELAs). These components are installed on the managed entities to execute the management services. Management services are executable management functions in the form of mobile code. Mobile code can be defined as, "software modules obtained from remote systems, transferred across a network, and then downloaded and executed on a local system without explicit installation or execution by the recipient". ELAs are equipped with Mobile Code Execution Environment (MCEE) that executes the mobile code modules. This feature allows performing management operations autonomously.

3.1 Global u-Zone Management Server

Global u-Zone network server (GuMS) is the central control entity that monitors the overall status of all u-Zone network elements. It comprises of Context Manager (CM), Global Policy Manager (GPM), and Management Services Manager (MSM).

The Context Manager (CM) is responsible for aggregation and analysis of context information. CM takes filtered data collected at CHs and prepares a global network perspective. This perspective describes the overall status of u-Zone network. This perspective is updated whenever major changes as cluster splitting occurs in the network. CM keeps updating the Global Policy Manager (GPM) by sending the latest

perspective so that appropriate policy decisions can be taken. Context manager also manages a context database that contains these perspectives.

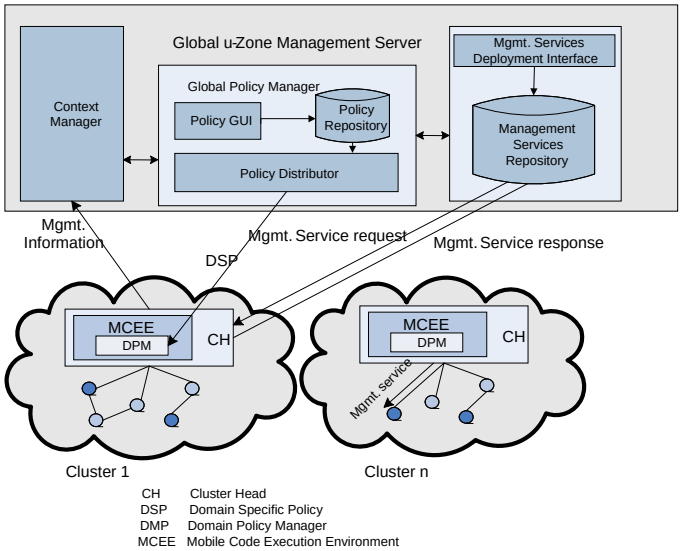

Figure 2 : Proposed Architecture

Global Policy Manger (GPM) is an essential component that provides an environment to create, modify, update, and delete network level policies. It also distributes the policies to the CHs. It comprises of a policy graphical user interface GUI, Policy Repository, and policy distributor. *Policy Graphical User Interface (PGUI)* is the environment to perform policy creation, modification, and deletion operations. It also provides a mechanism for conflict resolution. *Policy repository (PR)* is a centralized database that records policies provided by the network administrators. *Policy Distributor (PD)* distributes the policies to the appropriate domain policy managers (DPMs) on cluster heads.

Management Services Manager (MSM) provides environment to publish and deliver management services. It has Management Services Repository (MSR) that contains the network management services. A management services deployment interface is also provided to add new services for up-gradation to the self-management system.

3.2 Domain Policy Manager

Lightweight DPMs are installed on ZMs and any other nodes acting as CHs. DPMs are same in functionality as the global policy managers but with a cluster-wide scope. DPMs acquire global policy and adapt it according to the cluster dynamics. A DPM provides an interface for the users to directly specify local cluster policies and monitor cluster status. It also distributes policies and intelligence to the sufficiently capable MANET nodes via mobile code for local execution and enforcement of

policies. Furthermore these managers are themselves mobile and can migrate to other capable nodes in case the cluster-head changes due to changing node characteristics. The main purpose of these managers is to manage the cluster, query the cluster-wide network parameters, and prepare a MANET perspective locally for decision-making. The detailed architecture of DPM is given in Figure 3.

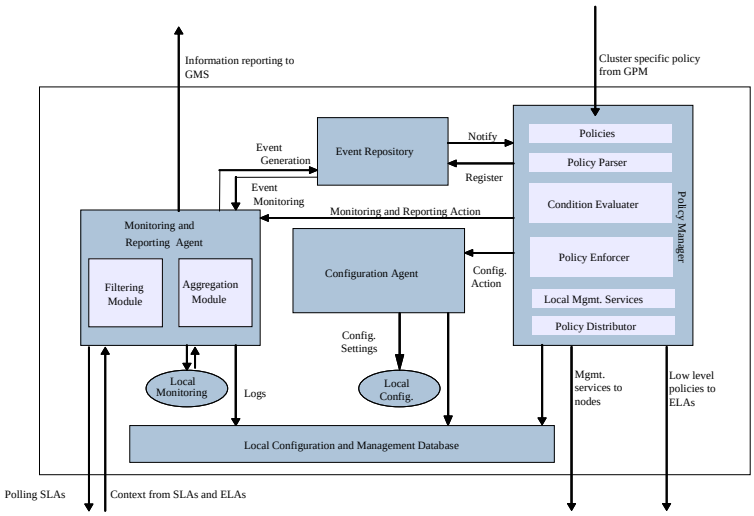

Figure 3 : DPM Architecture

3.3 Local Agents

The local agents are installed onto common MANET nodes that are required to be monitored and controlled continuously for optimized performance tuning. Local agents can further be classified into ELAs and SLAs. ***Extended Local Agents*** are equipped with a customizable MIB and management intelligence to process and implement lightweight policy received from the cluster-head. ELAs are installed on sufficiently capable MANET nodes for local policy enforcement and to minimize the communication overhead in frequent polling from the cluster-head. Such extended nodes are also provided MCEE to support mobile code execution. These agents get the policy from the cluster-heads and implement it on the local node. In this way, extended local agents achieve autonomous node management. These agents also send local information to the cluster-head every specific interval of time as specified in the policy. ***Simple local agents*** are analogous to the agents defined in SNMP (Simple Network Management Protocol). However they are attuned to wireless and mobile environments. These agents maintain a MIB to retrieve, update, and communicate a predefined list of MANET-node parameters and localized network information.

3.4 Operational Details

The monitoring agents at each node continuously send the context information to DPM at their respective CH. CM interacts with the DPMs to retrieve cluster-wide information, prepares a network-wide view and shares it with the policy manager. In case there is a change in scenario and interference is needed, GPM downloads a specific policy at the DPM which in turn, downloads respective services from management services repository and executes them. We categorize the policies as configuration policies, optimization policies, healing policies, and general policies.

Configuration Policies are provided to perform configuration operations on a node level, cluster head level, mesh router level, and u-Zone level. Examples of configuration policies are CH bootstrap policy, node bootstrap policy, new node join policy, and CH delegation/migration policy. ***Optimization Policies*** define how utility functions can be applied efficiently to get optimum performance. ***Healing Policies*** provide reliability and robustness for the all levels of network management. They provide reliability and robustness to all the levels of u-Zone network.

For policy communication and management we propose a hybrid system. We use XML policies from GuMS to CH and from CH to nodes with ELAs. However the nodes that cannot support XML due to their limited resources can work with their SNMP. We propose to have an XML / SNMP gateway on CH for XML-SNMP translations. CH will make the necessary translations to communicate with these nodes. A similar gateway is proposed in [17]. This architecture provides robustness to support heterogeneous nodes. Thus a node that does not have an ELA can still work well in the u-Zone.

4 Self-management Architecture

Autonomy is the most important design goal of our architecture that provides self-management functions to u-Zone Network. Self management functionality helps components to self-organize into composite entities, optimally providing required, often complex functions. It makes the systems manage themselves according to an administrator's goals with least human intervention. Our solution makes the network to optimize itself by monitoring itself, estimating its new transitions and act accordingly.

Network sate Management is the most essential component of our architecture to provide the required information to self-management modules. By network state we mean various network metrics e.g. nodes in the network, traffic load on all communication links etc. Certain events happening in the network e.g. node movement, link breakage, number of applications running over the network and traffic load on all the links etc, change the network state. We need to manage the network according to the current network state in order to plan future policies based on network state statistics. We maintain a network state graph that is then used to implement various QoS provisioning, resource management, self-healing and self-configuration functions. The graph maintains the information about all nodes', in

mesh and MANETs, including node's resources, applications etc. It also maintains the link information about all the communication paths. As the mesh back-bone does not change so rapidly the majority of operations, to maintain the network state graph, are related to the MANETs.

To fully visualize the realization of self-management architecture in u-Zone network we focus on different aspects including:-

- ***Self Configuration*** of a node determines its operational and maintenance characteristics as well as application execution, data communication, and data forwarding. Self-configuration module makes managed components to configure themselves automatically in accordance with high-level policies.
- ***Self Fault-management*** relates to recovering from the network and component failures automatically.
- ***Self Optimization*** continually seeks the ways to improve the network performance. It keeps identifying and seizing opportunities to make network performance more efficient.
- ***Self Healing*** describes the property that each node has the ability to perceive that it is not operating correctly and, without human intervention, makes the necessary adjustments to restore itself to normal operation.

To better illustrate the different aspects of self-management we present one scenario each for self-fault management, self-configuration, and self-healing.

4.1 Scenario 1: Self-Management

In this scenario we consider a cluster where node 'A' is working as a CH. At any given situation the owner of the node installs an application that needs a reboot of the node. Before executing the reboot process the CH will delegate its role, temporarily to another capable node that can act as CH. The sequence of actions is given below:-

1. Node 'A' checks the most suitable node from its cache by looking at the nodes who have MCEE installed on them
2. Node 'A' finds that node 'B' is the best suitable candidate
3. It send 'B' a request message to accept the role of CH
4. 'B' send the acceptance message
5. Node 'A' transfers its cluster-wide context information to node 'B'
6. Node 'B' starts to work as temporary cluster head
7. Node 'A' restarts after x seconds
8. Node 'A' resumes back to normal operation
9. It notifies 'B' that it is operational now
10. Node 'B' transfers updated context information on node 'A'
11. Node 'A' notifies itself as CH

4.2 Scenario 2: Self-Configuration

In this scenario we consider a hierarchical topology. The Cluster heads retrieve, process (aggregate and filter) data from u-Zone leave nodes, and forward this information to GuMS. In case the number of nodes in a cluster increases more than 'n' we prefer a cluster splitting. This will distribute the overhead of one CH into two CHs and will also help in maintaining optimum k-hop clusters. Following operations will be taken to tackle the situation:

1. Cluster Head 'A' reports an overload to the context awareness server
2. Context awareness server prepares a perspective for entire u-Zone
3. CM sends this perspective to the GPM
4. GPM issues an appropriate policy according to the perspective
5. Management service is downloaded from MSR to cluster 'A'
6. 'B' is chosen as second CH, as mentioned in scenario 1
7. 'B' notifies itself as CH and nodes 'close' to it are attached to 'B'
8. 'B' helps reconfiguring the nodes attached to it

4.3 Scenario 3: Self-Healing

This scenario is to maintain stability of the node. When a node becomes flooded with traffic and becomes vulnerable to crashing, the self-healing module becomes active and roll backs the already running processes. This rolling back is done by migrating some processed to other nodes and in some cases, by killing unused processes. We have designed a self-healing engine, with four different levels of defense, in order to incorporate a node resident component to handle network faults. We use normal functionality model to realize the self-healing. The activity of each node is monitored and is matched with a normal range of parameters. If any abnormal behavior observed, a solution (vaccine) is searched, locally or globally, for fault removal. The details of that engine are out of scope of this paper.

5 Conclusion

In this paper we have presented an autonomic management framework that will help realizing ubiquity, by managing the hybrid wireless networks. Our architecture supports context-aware policy-based work management at all levels of a hybrid wireless network. The autonomic sense-and-control loop exists at node level, at CH level, as well as at the whole network level. An XML / SNMP gateway is proposed at CH that makes it more robust to deal with heterogeneity. We describe the scenarios to visualize the realization of self-management operations in hybrid networks.

References

1. Mathew W.D.L., Miller J., Vaidya N.H.: A hybrid network implementation to extend infrastructure reach. UIUC Technical Report (2003)
2. Hsieh H.-Y., Sivakumar R..: Towards a Hybrid Network Model for Wireless Packet Data Networks. IEEE Symposium on Computers and Communications (ISCC), Taormina, Italy, (2002)
3. Oh M.: Network management agent allocation scheme in mesh networks. Communications Letters, IEEE Volume 7, Issue 12, (2003) pp:601 – 603
4. Kishi Y., Tabata K., Kitahara, T., Imagawa, Y., Idoue, A., Nomoto, S.: Implementation of the integrated network and link control functions for multi-hop mesh networks in broadband fixed wireless access systems Radio and Wireless Conference. (2004)
5. Yong-Lin S., DeYuan G., Jin P., PuBing S.: A mobile agent and policy-based network management architecture. Fifth International Conference on Computational Intelligence and Multimedia Applications ICCIMA (2003), pp: 177- 181.
6. Policy-based Management of Ad-hoc Enterprise Networks. HP Openview University Association 9th Annual Workshop, HP-OVUA, (2002)
7. Chen W., Jain N., Singh S.: ANMP: Ad hoc Network Management protocol. IEEE Journal on Selected Areas in Communications (1999) pp: 1506-1531.
8. Case J., McCloghrie K., Rose M., and Waldbusser S.: Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2). RFC 1902, IETF, (1996)
9. Internet Engineering Task Force. http://www.ietf.org
10. Distributed Management Task Force. http://www.dmtf.org
11. Common Information model. http://www.dmtf.org/standards/cim/
12. Chadha R., Cheng H., Cheng Y., Chiang J., Ghetie A., Levin G., and Tanna H.: Policy-Based Mobile Ad Hoc Network Management. Fifth IEEE International Workshop on Policies for Distributed Systems and Networks. (2004) pp: 35-44
13. Phanse K.S.: Policy-Based Quality of Service Management in Wireless Ad Hoc Networks. PhD. Thesis, Virginia Polytechnic Institute and State University, (2003)
14. Chadha R., Cheng Y.H., Chiang C.Y., Levin G., Li S., and Poylisher A.: DRAMA: A Distributed Policy-based Management System. Mobisys (2005)
15. Shen C.C., Srisathapornphat C., and Jaikaeo C.: An Adaptive Management Architecture for Ad hoc Networks. EEE Communication Magazine, vol. 41, no. 2. (2003)
16. Vivero J.: Proposal of a Model for the Management of Active Networks Based on Policies. Ph.D. thesis, Universitat Politècnica de Catalunya, (2003) http://www.tdx.cesca.es/TDX-0113104-100019/
17. Klie T., Straub F.: Integrating SNMP Agents with XML-Based Management Systems. IEEE Communication Magazine (July 2004) pp: 76-83.