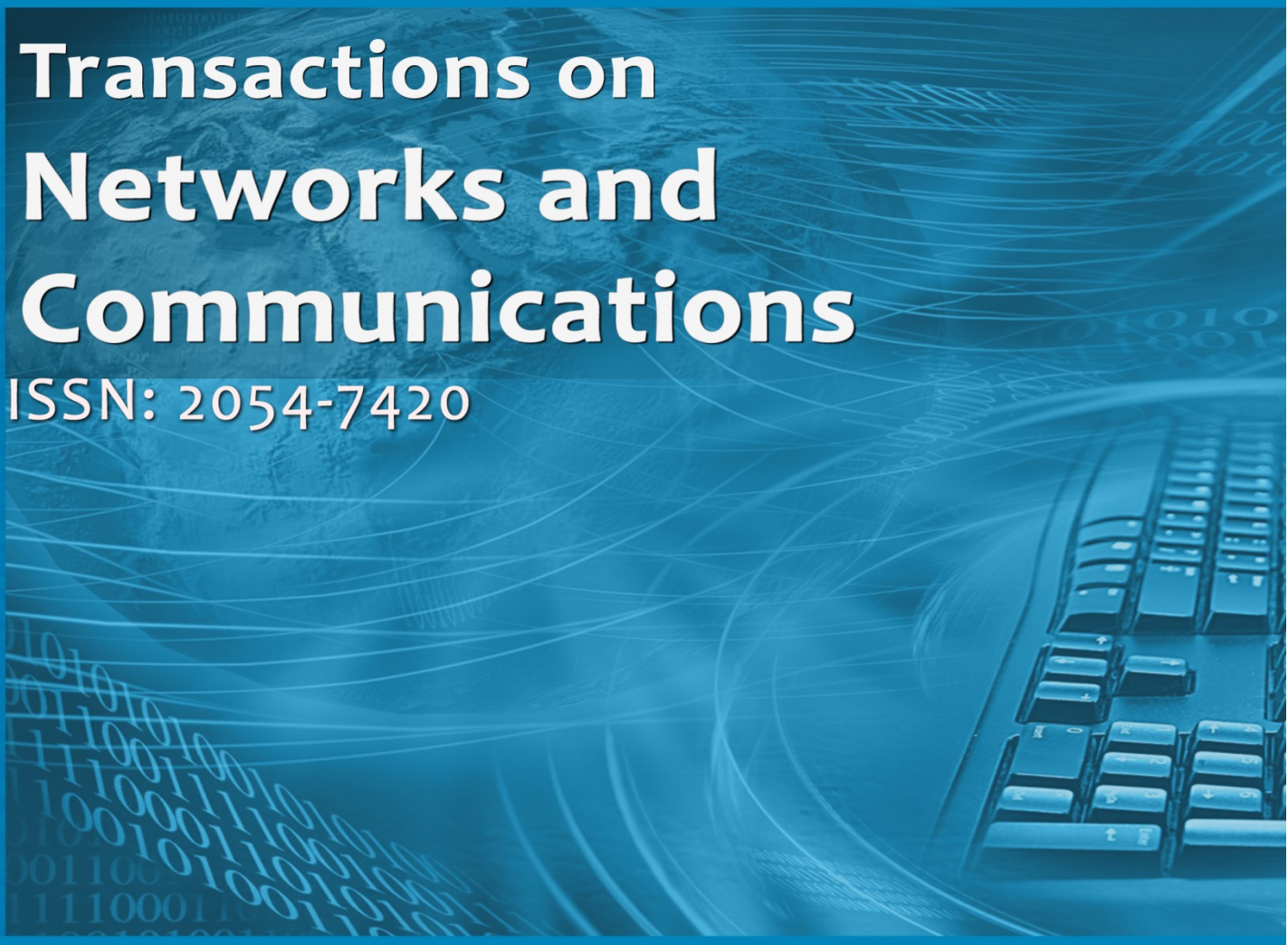
Volume 2. Issue 3 - June 2014

Transactions on Networks and Communications

ISSN: 2054-7420



Society for Sciences and Education - United Kingdom

TABLE OF CONTENTS

EDITORIAL ADVISORY BOARD

DISCLAIMER

All the contributions are published in good faith and intentions to promote and encourage research activities around the globe. The contributions are property of their respective authors/owners and the journal is not responsible for any content that hurts someone's views or feelings etc.

TNC Transactions on Networks and Communications
SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
VOLUME 2, ISSUE 3
ISSN: 2054 - 7420

Policy-based Wide Area Network Management System

[1] Kazuya Odagiri, [2]Shogo Shimizu, [3]Naohiro Ishii
[1] Yamaguchi University ,Yamaguchi, Japan; [2]Gakushuin Women's College, Tokyo, Japan
[3]Aichi Institute of Technology, Aichi, Japan
[1] odagiri@yamaguchi-u.ac.jp; [1] kazuodagiri@yahoo.co.jp; [2]shogo.shimizu@gakushuin.ac.jp
[3]ishii@aitech.ac.jp;

ABSTRACT

In the current Internet-based systems, there are many problems using anonymity of the network communication such as personal information leak and crimes using the Internet systems. This is because the TCP/IP protocol used in Internet systems does not have the user identification information on the communication data, and it is difficult to supervise the user performing the above acts immediately. As a solution for solving the above problem, there is the approach of Policy-based Network Management (PBNM). This is the scheme for managing a whole Local Area Network (LAN) through communication control of every user. In this PBNM, two types of schemes exist. The first is the scheme for managing the whole LAN by locating the communication control mechanisms on the course between network servers and clients. The second is the scheme of managing the whole LAN by locating the communication control mechanisms on clients. As the second scheme, we have been studied theoretically about the Destination Addressing Control System (DACS) Scheme. By applying this DACS Scheme to Internet system management, we realize the policy-based Internet system management. In this paper, we show the DACS system theoretically.

Keywords: Policy-based network management, DACS Scheme, NAPT

1. INTRODUCTION

In the current Internet systems, there are many problems using anonymity of the network communication, such as personal information leak and crimes using the Internet systems. The news of the information leak in the big company is sometimes reported through the mass media. Because TCP/IP protocol used in Internet systems does not have the user identification information on the communication data, it is difficult to supervise the user performing the above acts immediately. Many solutions and technologies for managing Internet systems based on TCP/IP protocol have been emerged, namely, Domain Name System (DNS) [3], Routing protocols, firewall (F/W) [7], and Network Address Port Translation (NAPT) [8] / Network

DOI: 10.14738/tnc.23.192
Publication Date: 12th June 2014
URL: http://dx.doi.org/10.14738/tnc. 23.192

Address Translation (NAT) [9]. However, they are for managing the specific part of the Internet systems, and have no purpose of solving our target problems.

PBNM might be a solution for solving these problems. However, it is a scheme for managing a whole LAN through communication control of every user, and cannot be applied to the Internet systems. It is often used in a scene of campus network management. In a campus network, network management is quite complicated. Because a network administrative department manages only a small portion of the wide needs of the campus network, there are some user support problems. For example, when mail boxes on one server are divided and relocated to some different server machines, it is necessary for some users to update a client machine's setups. Most of computer network users in a campus are students. Because they do not check frequently their e-mail, it is hard work to make them aware of the settings update. This administrative operation is executed by means of web pages and/or posters. For the system administrator, it is difficult to support every student in terms of time and workload. Because the PBNM manages a whole LAN, it is easy to solve this kind of problem. In addition, for the problem such as personal information leak, the PBNM can manage a whole LAN by making anonymous communication non-anonymous. As the result, it becomes possible to identify the user who steals personal information and commits a crime swiftly and easily. Therefore, by applying the PBNM, we study about the policy-based Internet system management.

In the existing PBNM, there are two types of schemes. The first is the scheme of managing the whole LAN by locating the communication control mechanisms on the course between network servers and clients. The second is the scheme of managing the whole LAN by locating the communication control mechanisms on clients. It is difficult to practically apply the first scheme to Internet system management, because the communication control mechanism needs to be located on the course between network servers and clients necessarily. Because the second scheme locates the communication control mechanisms as the software on each client, it becomes possible to apply the second scheme to Internet system management by devising the installing mechanism so that users can install the software to the client easily.

As the second scheme, we have been studied, theoretically, about the Destination Addressing Control System (DACS) Scheme. As the works on the DACS Scheme, we showed the basic principle of the DACS Scheme [28], and security function [29]. After that, we implemented a DACS system to realize a concept of the DACS Scheme [30]. By applying this DACS Scheme to Internet systems, we realize the policy-based Internet system management. In this paper, we show the encrypting mechanism, which is suitable for the wDACS system.

In Section II, motivation and related research are described. Existing DACS Scheme are described in Section III. Then, in Section IV, the wDACS system is suggested and experimental results for confirming the possibility of the wDACS system.

2. MOTIVATION AND RELATED RESERACH

In the current Internet systems, problems using anonymity of the network communication, such as personal information leak and crimes using the Internet systems occur. Because the TCP/IP protocol used in Internet systems does not have the user identification information on the communication data, it is difficult to supervise the user performing the above acts immediately.

Many solutions and technologies for Internet systems management using TCP/IP [1][2] have been proposed and are in use:

1) DNS [3]
2) Routing protocols:
    a. Interior Gateway Protocols (IGP), such as Routing Information Protocol (RIP) [4] and Open Shortest Path First (OSPF) [5]
    b. Exterior Gateway Protocols (EGP), such as Border Gateway Protocol (BGP) [6]
3) F/W [7]
4) NAT [8] / NAPT [9]
5) Load balancing [10][11]
6) Virtual Private Network (VPN) [12][13]
7) Public Key Infrastructure （PKI) [14]
8) Server virtualization [15]

However, they are for managing the specific aspect of the Internet systems, but have no purpose of solving our target problems.

In the following, we are focusing on policy-based thinking, to study the policy-based Internet system management.

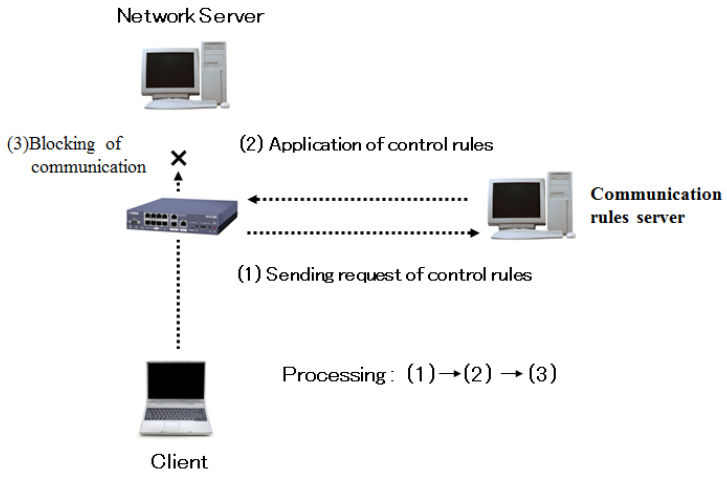


Figure 1. Principle in the first scheme.

In PBNM, there are two types of schemes. The first scheme is described in Figure 1. This scheme is standardized in various organizations. In IETF, a framework of PBNM [16] was established. Standards about each element constituting this framework are as follows. As a model of control information stored in the server called Policy Repository, Policy Core Information model (PCIM) [17] was established. After it, PCMIe [18] was established by extending the PCIM. To describe them in the form of Lightweight Directory Access Protocol (LDAP), Policy Core LDAP Schema (PCLS) [19] was established. As a protocol to distribute the control information stored in Policy Repository or decision result from the Policy Decision Point (PDP) to the Policy Enforcement Point (PEP), Common Open Policy Service (COPS) [20] was established. PDP is the point which performs the judgment about the communication control, and PEP is the point which performs the communication control based on the judgment. Based on the difference in distribution method, COPS usage for RSVP (COPS-RSVP) [21] and COPS usage for Provisioning (COPS-PR) [22] were established. RSVP is an abbreviation for Resource Reservation Protocol. The COPS-RSVP is the method as follows. After the PEP detected the communication from a user or a client application, the PDP makes a judgmental decision for it. The decision is sent and applied to the PEP, and the PEP adds the control to it. The COPS-PR is the method of distributing the control information or decision result to the PEP before accepting the communication.

Next, in the Distributed Management Task Force (DMTF), a framework of PBNM called Directory-enabled Network (DEN) was established. Like the IETF framework, control information is stored in the server called Policy Server which is built by using the directory service, such as LDAP [23], and is distributed to network servers and networking equipment such as switch and router. As the result, the whole LAN is managed. The model of control information used in DEN is called Common Information Model (CIM); the schema of CIM (CIM Schema Version 2.30.0) [25] was published. CIM was extended to support DEN [24], and was incorporated in the framework of DEN.

In addition, Resource and Admission Control Subsystem (RACS) [26] was established in Telecoms and Internet converged Services and protocols for Advanced Network (TISPAN) of European Telecommunications Standards Institute (ETSI), and Resource and Admission Control Functions (RACF) [27] was established in International Telecommunication Union Telecommunication Standardization Sector (ITU-T).

However, all the frameworks explained above are based on the principle shown in Figure 1. As problems of these frameworks, two points are presented as follows. Essential principle is described in Figure 2. To be concrete, in the PDP, judgment such as permission and non-permission for communication pass is performed based on policy information. The judgment is notified and transmitted to the point called the PEP. Based on that judgment, the control is added for the communication that is going to pass by.

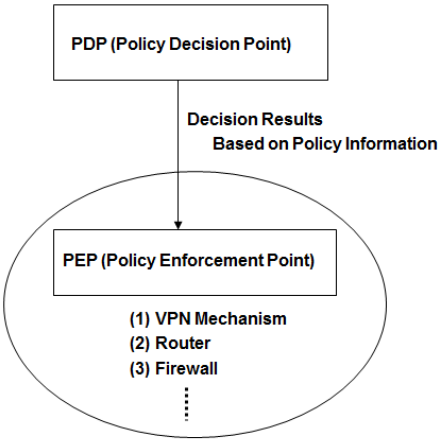


Figure 2. Essential Principle.

The principle of the second scheme is described in Figure 3 [28][29][30][31]. By locating the communication control mechanisms on the clients, the whole LAN is managed. Because this scheme controls the network communications on each client, the processing load is low. However, because the communication control mechanisms need to be located on each client, the workload becomes heavy.

We aim at realizing the PBNM management of an Internet system by applying these two schemes. However, it was difficult to apply the first scheme to Internet system management practically. In the first scheme, the communication control mechanism needs to be located on the course between network servers and clients, necessarily. As the result, the mechanism is operated from outside. It is more likely to violate the network and security policy of each organization.

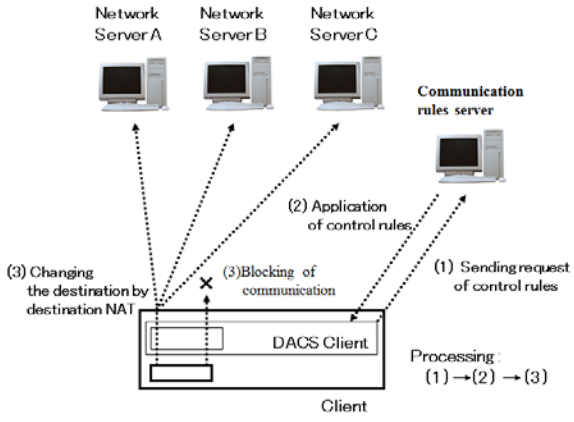


Figure 3. Principle in second scheme.

On the other hand, the second scheme locates the communication controls mechanisms on each client. The software for communication control is installed on each client. Therefore, by devising the installing mechanism letting users install software to the client easily, it becomes possible to apply the second scheme to Internet system management.

3. EXISTING DACS SCHEME

3.1 Basic Principle of the DACS Scheme

Figure 4 shows the basic principle of the network services by the DACS Scheme. At the processing of the (a) or (b), as shown in the following, the DACS rules (rules defined by the user unit) are distributed from the DACS Server to the DACS Client.

a) Processing of a user logging in the client.
b) Processing of a delivery indication from the system administrator.

According to the distributed DACS rules, the DACS Client performs (1) or (2) operation. Then, communication control of the client is performed for every user used for login.

1) Destination information on IP Packet, which is sent from application program, is changed.

2) IP Packet from the client, which is sent from the application program to the outside of the client, is blocked.

An example of the case (1) is shown in Figure 4. In Figure 4, the system administrator can distribute a communication of the user used for login to the specified server among servers A, B or C. Moreover, the case (2) is described. For example, when the system administrator wants to forbid an user to use Mail User Agent (MUA), it is performed by blocking IP Packet with the specific destination information.

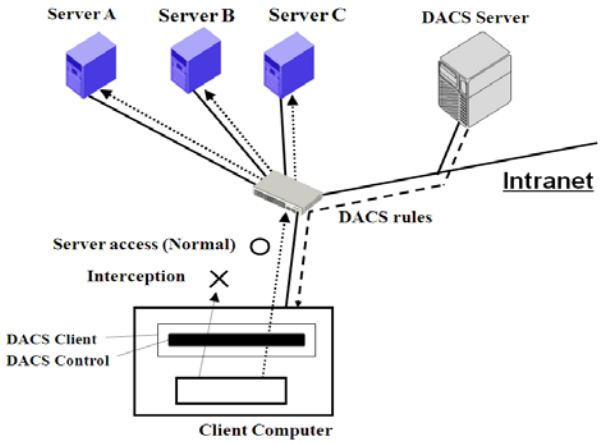


Figure 4. Basic Principle of the DACS Scheme.

In order to realize the DACS Scheme, the operation is done by a DACS Protocol, as shown in Figure 5. As shown by (1) in Figure 5, the distribution of the DACS rules is performed on communication between the DACS Server and the DACS Client, which is arranged at the application layer. The application of the DACS rules to the DACS Control is shown by (2) in Figure 5.

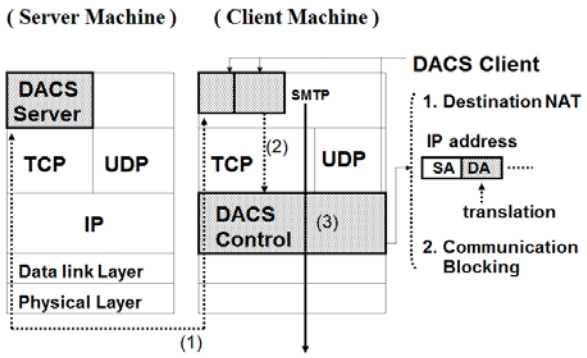


Figure 5. Operation of the DACS Protocol.

The steady communication control, such as a modification of the destination information or the communication blocking is performed at the network layer, as shown by (3) in Figure 5.

3.2 Communication Control on Client

The communication control of every user was given. However, it may be better to perform communication control every client instead of every user. For example, it is the case where many and unspecified users use a computer room, which is controlled. In this section, the method of communication control every client is described, and the coexistence method with the communication control of every user is considered.

When a user logs in to a client, the IP address of the client is transmitted to the DACS Server from the DACS Client. Then, if the DACS rules corresponding to IP address, is registered into the DACS Server side, it is transmitted to the DACS Client. Then, communication control for every client can be realized by applying to the DACS Control. In this case, it is a premise that a client uses a fixed IP address. However, when using DHCP service, it is possible to carry out the same control to all the clients linked to the whole network or its sub network. i.e.

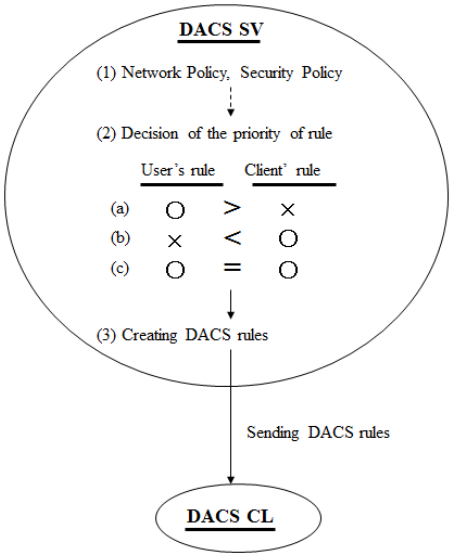


Figure 6. Creating the DACS rules on the DACS Server.

When using the communication control of every user and every client, communication control may conflict. In that case, a priority needs to be given. The judgment is performed in the DACS Server side as shown in Figure 6. Although not necessarily stipulated, the network policy or security policy exists in the organization, such as a university (1). The priority is decided according to the policy (2). In (a), priority is given for the user's rule to control communication by the user unit. In (b), priority is given for the client's rule to control communication by the client unit. In (c), the user's rule is the same as the client's rule. As the result of comparing the conflict rules, one rule is determined, respectively. Those rules and other rules not overlapping are gathered, and the DACS rules are created (3). The DACS rules are transmitted to the DACS Client. In the DACS Client side, the DACS rules are applied to the DACS Control. The difference between the user's rule and the client's rule is not distinguished.

3.3 Security Mechanism of the DACS Scheme

In this section, the security function of the DACS Scheme is described. The communication is tunneled and encrypted by use of Secure Shell (SSH) [31]. By using the function of port forwarding of SSH, it is realized to tunnel and encrypt the communication between the network server and the DACS Client, which the DACS Client is installed in. Normally, to communicate from a client application to a network server by using the function of port forwarding of SSH, the local host (127.0.0.1) needs to be indicated on that client application as a communicating server. The transparent use of a client as the virtue of the DACS Scheme is lost. The transparent use of a client means that a client can be used continuously without changing setups when the network system is updated. The function that does not fail the transparent use of a client is needed. The mechanism of that function is shown in Figure 7.

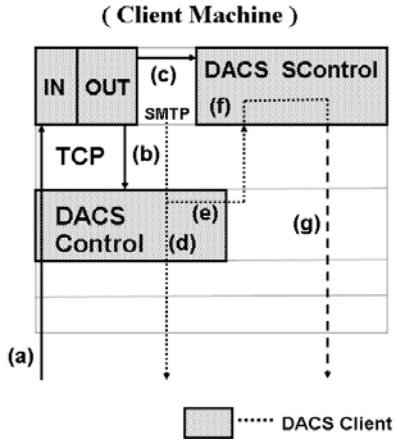


Figure 7. Extend Security Function.

The changed point on network server side is shown as follows, in comparison with the existing DACS Scheme. SSH Server is located and activated, and communication, except, SSH is blocked.

In Figure 7, the DACS rules are sent from the DACS Server to the DACS Client (a). On the DACS Client that accepts the DACS rules, the DACS rules are applied to the DACS Control in the DACS Client (b). These processes are same as the existing DACS Scheme. After functional extension, as shown in (c) of Figure 7, the DACS rules are applied to the DACS SControl. Communication control is performed in the DACS SControl with the function of SSH. By adding the extended function, selecting the tunneled and encrypted or not tunneled and encrypted communication is done for each network service. When communication is not tunneled and encrypted, communication control is performed by the DACS Control, as shown in (d) of Figure 7. When communication is tunneled and encrypted, destination of the communication is changed by the DACS Control to localhost, as shown in Figure 7. In Figure 7, the communication to localhost is shown with the arrows from (e) to the direction of (f). After that, by the DACS SControl which is used for the VPN communication, the communicating server is changed to the network server and tunneled and encrypted communication is sent as, shown in (g) of Figure 7, which are realized by the function of port forwarding of SSH. In the DACS rules applied to the DACS Control, localhost is indicated as the destination of communication. As the functional extension explained in the above, the function of tunneling and encrypting communication is realized in the state of being suitable for the DACS Scheme, that is, with the transparent use of a client. Distinguishing the control in the case of tunneling and encrypting or not tunneling and encrypting by a user unit is realized by changing the content of the DACS rules applied to the DACS Control and the DACS SControl. By tunneling and encrypting the communication for one network service from all users, and blocking the not tunneled and decrypted communication for that network service, the function of preventing the communication for one network service from the client, which DACS Client is not installed in, is realized. Moreover, the communication to the network server from the client on which DACS Client is not installed in is permitted; each user can select whether the communication is tunneled and encrypted or not.

3.4 Technical Points in Implementation of DACS System

(a) Communications between the DACS Server and the DACS Client

The Communications between the DACS Server and the DACS Client such as sending and accepting the DACS rules were realized by the communications through a socket in TCP/IP.

(b) Communication control on the client computer

In this study, the DACS Client working on windows XP was implemented. The functions of the destination NAT and packet filtering required as a part of the DACS Control were implemented by using Winsock2 SPI of Microsoft. As it is described in Figure 8, Winsock2 SPI is a new layer which is created between the existing Winsock API and the layer under it.

To be concrete, though connect() is performed when the client application accesses the server, the processes of destination NAT for the communication from the client application are built in WSP connect() which is called in connect(). In addition, though accept() is performed on

the client when the communication to the client is accepted, the function of packet filtering is implemented in WSPaccept() which is called in accept().

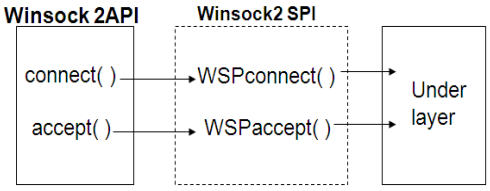


Figure 8. Winsock2 SPI.

(c) VPN communication

The client software for the VPN communication, that is, the DACS SControl was realized by using the port forward function of the Putty. When the communication from the client is supported by the VPN communication, first, the destination of this communication is changed to the localhost. After that, the putty accepts the communication, and sends the VPN communication by using the port forward function.

4. WDACS SYSTEM

In this section, the content of wDACS system is explained.

4.1 System Configuration of wDACS system

The system configuration of the wDACS system is described in Figure 9.

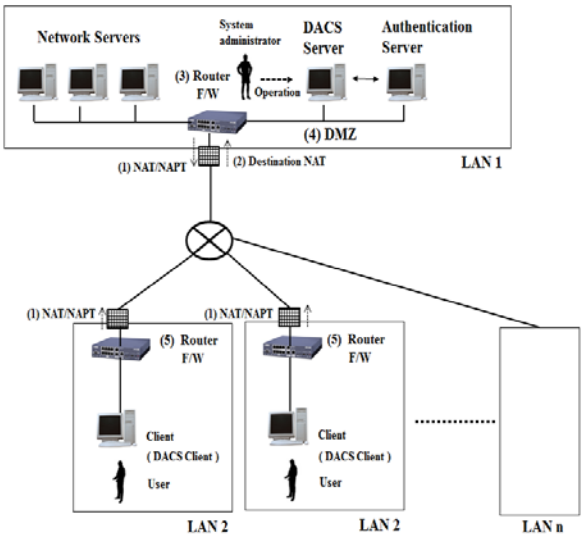


Figure 9. Basic System Configuration of wDACS system

First, as preconditions, because private IP addresses are assigned to all servers and clients existing in from LAN1 to LAN n, mechanisms of NAT/NAPT are necessary for the communication from each LAN to the outside. In this case, NAT/NAPT is located on the entrance of the LAN such as (1), and the private IP address is converted to the global IP address towards the direction of the arrow.

Next, because the private IP addresses are set on the servers and clients in the LAN, other communications except those converted by Destination NAT cannot enter into the LAN. But, responses for the communications sent form the inside of the LAN can enter into the inside of the LAN because of the reverse conversion process by the NAT/NAPT.

In addition, communications from the outside of the LAN1 to the inside are performed through the conversion of the destination IP address by Destination NAT. To be concrete, the global IP address at the same of the outside interface of the router is changed to the private IP address of each server.

From here, system configuration of each LAN is described. First, the DACS Server and the authentication server are located on the DMZ on the LAN1 such as (4). On the entrance of the LAN1, NAT/NAPT and destination NAT exists such as (1) and (2). Because only the DACS Server and network servers are set as the target destination, the authentication server cannot be accessed from the outside of the LAN1. In the LANs from LAN 2 to LAN n, clients managed by the wDACS system exist, and NAT/NAPT is located on the entrance of each LAN such as (1). Then, F/W, such as (3) or (5), exists behind or with NAT/NAPT in all LANs.

4.2 Key ExchangeMechanism for wDACS system

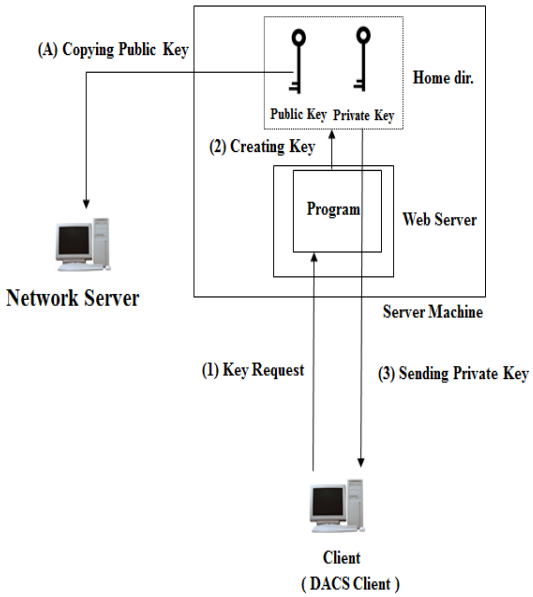


Figure 10. Mechanism of Key Exchange.

This is a periodical key exchange mechanism which is necessary for encrypted communications between the network servers and the client computers. This mechanism is incorporated at the last part of the initialization process of the DACS Client. The preconditions are as follows.

a) The communications between the DACS Client and the Web Server are encrypted by the https.
b) The communications between the Server Machine moving the Web Server and network servers are encrypted by SSH.
c) This mechanism is located on the Server Machine which is separated physically with DACS Sever for the management of a large-scale network with many clients.

Next, the processing of this mechanism is described. First, the key request is performed from the DACS Client (1). The program on the Web Server receives the request, and creates two kinds of keys which are a public key and a private key (2). Then, the program sends the private key to the client (3). The public key stored in the home directory on the Server Machine is copied and stored on the network server by mirroring through SSH. To be concrete, network commands such as rsync and rdiff-backup are used. The mirroring process is performed just before the transmission of the private key.

4.3 Encrypted Communication Mechanism for the wDACS system

In this section, two functions to realize the encrypted communication are described.

(1) Function of encrypted communications in user authentication processes

In this section, the function of the encrypted communications in user authentication processes, which is suitable for the wDACS system, is described. The content of the function is shown in Figure 11.

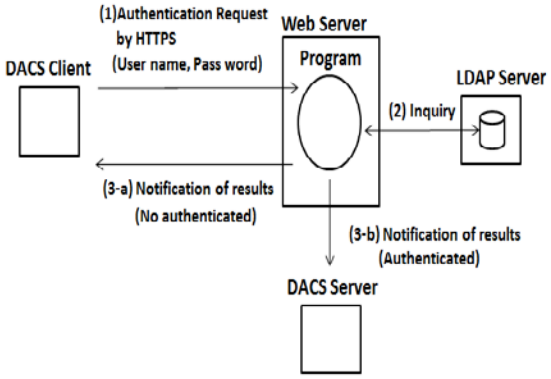


Figure 11. Function of user authentication processes.

First, authentication request is performed form the DACS Client to the program on the Web Server (1). The Program performs inquiry to the LDAP Server which stores user accounts (user

name, pass word) (2). As the result, if authentication is not permitted, the results are notified to the DACS Client (3-a). The DACS Client stops performing subsequent processing. If authentication is permitted, the results are notified to the DACS Server (3-b). The DACS Server performs the processing described in next section.

(2) Function of encrypted communications

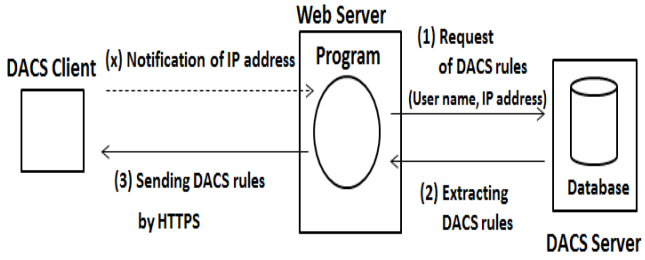


Figure 12. Function of transmission and reception processes for control information.

In this section, the function of the encrypted communications in the DACS rule's transmission and reception processes, which is suitable for the wDACS system, is described.

First, as a part of process (1) in Figure 11, the IP address of the client where the DACS Client is installed is notified with user name and password to the program on the Web Server. This process is described as process (x) in Figure 12, which is shown by a dotted arrow.

Next, based on them, the program performs a request of the DACS rules to the DACS Server (1). The DACS rules are extracted from the database of the DACS Server, and sent to the program on the Web Server (2). The program receives them, and sends to the DACS Client.

Specific to these two functions is the use of HTTPS. Because this wDACS system needs to be extended for Internet management, we chose HTTPS used widely in the world of the Internet.

4.4 Experiments for confirming the possibility of the wDACS system

To confirm the possibility of the wDACS Scheme, we performed a functional experiment. By this experiment, we confirmed that the existing DACS Scheme could be operated in cloud environment.

4.4.1 Constitution of the experiment system

In Figure 12, the experiment system used in this research was described. Two virtual servers which placed VMWare ESXi 5.1 were prepared. Each virtual server was constructed as follows.

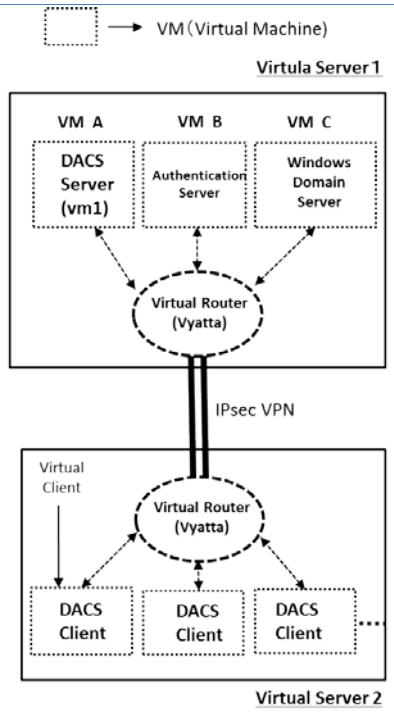


Figure.13 Experiment system

(1) Virtual Server 1 (CPU ： 2.8GHz 4Core×1 Memory:16GB)

Virtualization software ： VMWareESXi5.1

Virtual machine A ：

Operating System (CentOS6.5)

Software for DACS Server

Virtual machine B ：

Operating System (CentOS6.5)

Authentication server (OpenLDAP2.4)

Virtual machine C ：

Operating System (CentOS6.5)

Windows domain server (Samba3.6)

Virtual router for a gateway (Vyatta6.6 ： 64bit)

(2) Virtual Server 2 (CPU ： 2.6GHz 4Core×1 Memory:16GB)

Virtualization software ： VMWareESXi5.1

Each virtual machine (5 virtual machine) ：

Operating System (Windows XP Pro)

Software for DACS Client

Virtual router for a gateway (Vyatta6.6 ： 64bit)

Because we assumed that a service based on this scheme would be offered in the cloud environment, we prepared the experimental environment which each virtual router on each virtual server is connected by IPsec VPN each other.

The DACS Server was located on the virtual machine in the virtual server 1. The DACS Client was located on each virtual client in the virtual server 2, and the DACS Client was located on the CentOS in each virtual machine. The policy information was sent and received through the VPN connected by two virtual routers on each virtual server.

4.4.2 Content of the functional experiment

By using the experiment system in Figure 13, we performed the function experiments about two functions as follows.

(a) User authentication function

In this experimental system, the Windows OS (XP Pro) is used as an operating system on each virtual machine in the virtual server 2. In addition, because we intend to release the software developed to realize this scheme, we adopt the user authentication mechanism by free software. To be concrete, the user authentication is realized by the cooperation of two kinds of servers as follows. To be concrete, user authentication processes are performed between the clients on the virtual server 2 and the DACS Server on the virtual server1. About this point, we could confirm the movement normally.

(Server1) OpenLDAP server for managing user accounts

(Server2) Samba server for building a windows domain

(b) Delivery function of policy information

In this scheme, after the process (a), the policy information is sent and received through the VPN connected by two virtual routers on each virtual server. About this process, I performed two cases of movement experiments as follows.

(Case1) One virtual machine was operated on the virtual server 2.

(Case2) Some virtual machines (Five virtual machines) were operated on the virtual server 2.

4.4.3 Result of functional experiment

In the above both cases, the DACS system was operated with no problem. Then, the communication log was shown in Figure 14.

```
DATETIME:2014/04/10 01:11:18  MESSAGE:ANSWER_DATA received --- STATUS=50 (DACS_GET_OK) FUNCTION:main
DATETIME:2014/04/10 01:11:18  MESSAGE:disconnected by win-service!! FUNCTION:main
DATETIME:2014/04/10 01:11:18  MESSAGE:END!! FUNCTION:main
```

Figure. 13 Communication log on the DACS Client

As the result, we could confirm that the DACS Scheme to premise a physical client conventionally was operated in cloud environment. However, when we prepared the experimental system, it was burden to make many virtual machines in the virtual server 2. At

this point, the mechanism for managing many virtual machines is necessary in the form that adapted to the DACS Scheme. After this research, we will study as another research.

5. CONCLUSION

In this paper, we showed the policy-based wide area network management system called wDACS system. This system is realized by the extension of the policy-based network management system called the DACS system, which is the management scheme of the LAN one organization hold. As a future study, the wDACS system will be implemented by incorporating three functions suggested in this paper, and evaluations will be performed.

REFERENCES

[1]. V. Cerf and E. Kahn, "A Protocol for Packet Network Interconnection," IEEE Trans. on Commn, vol. COM-22, pp. 637-648, May 1974.

[2]. B. M. Leiner, R. Core, J. Postel, and D. Milis, "The DARPA Internet Protocol Suite," IEEE Commun.Magazine, vol. 23 pp. 29-34 March 1985.

[3]. P. Mockapetris and K. J. Dunlap. "Development of the domain name system," SIGCOMM'88, 1988.

[4]. http://tools.ietf.org/html/rfc2453 [retrieved: 2, 2014]

[5]. http://www.ietf.org/rfc/rfc2328.txt [retrieved: 2, 2014]

[6]. http://tools.ietf.org/html/rfc4271 [retrieved: 2, 2014]

[7]. A. X. Liu and M. G. Gouda, "Diverse Firewall Design," IEEE Trans. on Parallel and Distributed Systems, vol. 19, Issue. 9, pp. 1237-1251, Sept. 2008.

[8]. http://tools.ietf.org/html/rfc1631 [retrieved: 2, 2014]

[9]. M. S. Ferdous, F. Chowdhury, and J. C. Acharjee, "An Extended Algorithm to Enhance the Performance of the Current NAPT," Int. Conf. on Information and Communication Technology (ICICT '07), pp. 315-318, March 2007.

[10]. S. K. Das, D. J. Harvey, and R. Biswas, “Parallel processing of adaptive meshes with load balancing,” IEEE Tran.on Parallel and Distributed Systems, vol. 12, no. 12, pp. 1269-1280, Dec 2002.

[11]. J. Aweya, M. Ouellette, D. Y. Montuno, B. Doray, and K. Felske, “An adaptive load balancing scheme for web servers,” Int.,J.of Network Management., vol. 12, no. 1, pp. 3-39, Jan/Feb 2002.

[12]. C. Metz, “The latest in virtual private networks: part I,” IEEE Internet Computing, vol. 7, no. 1, pp. 87-91, 2003.

[13]. C. Metz, “The latest in VPNs: part II,” IEEE Internet Computing, vol. 8, no. 3, pp. 60-65, 2004.

[14]. R. Perlman, "An overview of PKI trust models," IEEE Network, vol. 13, issue 6, pp. 38-43, Nov/Dec 1999.

[15]. A. Singh, M. Korupolu, and D. Mohapatra, "Server-storage virtualization: Integration and load balancing in data centers," Int. Conf. for High Performance Computing, Networking, Storage and Analysis, pp. 1-12, Nov. 2008.

[16]. Yavatkar et al., "A Framework for Policy-based Admission Control," IETF RFC 2753, 2000.

[17]. B. Moore et al., "Policy Core Information Model -- Version 1 Specification," IETF RFC 3060, 2001.

[18]. B. Moore, "Policy Core Information Model (PCIM) Extensions," IETF 3460, 2003.

[19]. J. Strassner et al., " Policy Core Lightweight Directory Access Protocol (LDAP) Schema," IETF RFC 3703, 2004.

[20]. D. Durham et al., "The COPS (Common Open Policy Service) Protocol, " IETF RFC 2748, 2000.

[21]. S. Herzog et al., "COPS usage for RSVP", IETF RFC 2749, 2000.

[22]. K. Chan et al., "COPS Usage for Policy Provisioning (COPS-PR), " IETF RFC 3084, 2001.

[23]. CIM Core Model V2.5 LDAP Mapping Specification, 2002.

[24]. M. Wahl et al., "Lightweight Directory Access Protocol (v3)," IETF RFC 2251, 1997.

[25]. CIM Schema: Version 2.30.0, 2011.

[26]. ETSI ES 282 003: Telecoms and Internet converged Services and protocols for Advanced Network (TISPAN); Resource and Admission Control Subsystem (RACS); Functional Architecture, June 2006.

[27]. ETSI ES 283 026: Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN); Resource and Admission Control; Protocol for QoS reservation information exchange between the Service Policy Decision Function (SPDF) and the Access-Resource and Admission Control Function (A-RACF) in the Resource and Protocol specifica-tion", April 2006.

[28]. K. Odagiri，R. Yaegashi，M. Tadauchi, and N.Ishii, "Efficient Network Management System with DACS Scheme : Management with communication control, " Int. J. of Computer Science and Network Security, vol. 6, no. 1, pp. 30-36, January, 2006.

[29]. K. Odagiri，R. Yaegashi，M. Tadauchi, and N.Ishii, "Secure DACS Scheme," Journal of Network and Computer Applications," Elsevier, vol. 31, Issue 4, pp. 851-861, November 2008.

[30]. K. Odagiri, S. Shimizu, R. Yaegashi, M. Takizawa, and N. Ishii, "DACS System Implementation Method to Realize the Next Generation Policy-based Network Management Scheme," Proc. of Int. Conf. on Advanced Information Networking and Applications (AINA 2010), Perth, Australia, Japan, IEEE Computer Society, pp. 348-354, May 2010.

[31]. http://tools.ietf.org/html/rfc4251 [retrieved: 2, 2014]

TNC Transactions on Networks and Communications
SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
VOLUME 2, ISSUE 3
ISSN: 2054 - 7420

Cloud Computing and Offloading Framework for enhancing Android Smart Device Battery Life

Ahmed H. Najim, Shawkat K. Guirguis, Magda M. Madbouly
Department of Information Technology
Institute of Graduate Studies & Research, Alexandria University
Alexandria, Egypt
Ahmed_al_adhami@yahoo.com, Shawkat_g@yahoo.com, mmadbouly@hotmail.com

ABSTRACT

Information Technology (IT) field before cloud computing is not the same after. It has recently accelerated as new criteria for presenting and delivering services over the Internet. Cloud Computing is reacting positively as a concept that may greatly improve the smart devices (Smartphone, Tablet and Phablet) reliability. This paper mainly concentrates to enhance the smart devices dependability through and applications (app) during offloading of services among cloud and device to save energy, many challenges had faced that idea ex: unstable network, unstable Internet services, Synchronization and the limitations of open source smart devices operating system (OS). In order to cure the earlier challenges, An Android OS used with a new layer to offload the non-critical app's jobs. That should occur in One's of two clouds developed specially to do that, the first cloud is a private, connected with smart device through the same data network on-line and the second cloud is a public, related to the previous cloud and the smart device off-line when Internet exists. Using this system, Smart device users may choose to run their mobile application jobs either locally or offloading it to the cloud, this shall save energy spent from the smart device battery when using Third generation of mobile network (3G) or WI-FI. This system has achieved in the provision of high-energy reached 98.67% and will be seen in the context of this research paper.

Keywords: Offloading, Energy, Power consumption, Smart Device, Android, Mobile cloud computing.

1. INTRODUCTION

Second millennium has seen much technological progress, which was clearly highlighted in the mobile phone sector. Handheld mobile technology is reaching first responders, disaster-relief workers, and soldiers in the field to aid in various tasks, such as speech and image recognition, natural-language processing, decision making, and mission planning. [1] Mobile

DOI: 10.14738/tnc.23.236
Publication Date: 12th June 2014
URL: http://dx.doi.org/10.14738/tnc. 23.236

phones have been undergoing a breathtaking evolution over the last two decades, starting from simple devices with only voice services towards smart device offering novel services such as mobile Internet, high data rate connectivity and many more. Smart devices are battery driven to allow the highest degree of freedom for the user and the battery has to empower all the nice new features of a smartphone. Computation-Intensive tasks consume a large amount of power. At the same time, new expression appeared on the world of information technology called Cloud computing [2] a topic that received a great deal of attention from individuals and organizations from different disciplines in the last decade. This new environment implies a high flexibility and availability of computing resources at different levels of abstraction at a lower cost. The concept of Mobile Cloud Computing (MCC) intends to make the advantages of Cloud Computing available for mobile users, but will provide additional functionality to the cloud, as well. MCC will help to overcome limitations of mobile devices in particular of the processing power and data storage. It might also help to extend the battery life by moving the execution of commutation-intensive application "to the cloud." In this proposed system, developed mobile cloud computing model, provided the environment specifically designed for smart device users. This system allows users to create virtual smartphone images in the public cloud and remotely run their jobs in these images as they would locally By offloading only non-critical jobs when sustainable network existed to synchronization files. Both smart device and the private cloud hosting computer are connected to the same wireless local area network). Private cloud used as a bridge between the public cloud and the smart device. The main objectives of that paper are prolonging the battery life of smart devices by offloading non critical jobs of it with the cloud. Clarify how cloud computing services can increase the effectiveness, dependability, mobility and reliability of the android smart devices in the future. Figure 1 shows the basic proposed idea outline.

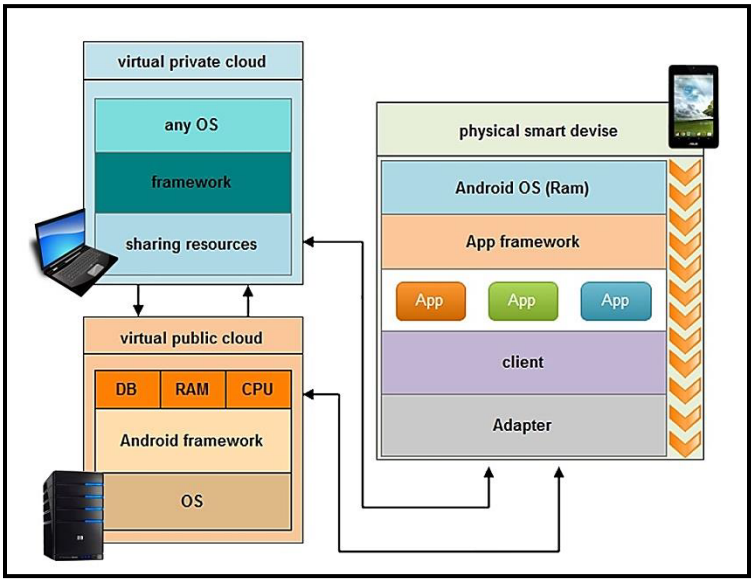


Figure 1: The basic idea Outline

2. RELATED WORK

Reducing power consumption in smart devices always had been a significant target for researchers and companies. After the huge revolution in the mobile sector and the release of a new category called smart phones characterized by the availability of using the internet, applications and many other features. Thus, all of those features are drying batteries faster than ever. Many papers, articles and theses written about that topic, each group of these research papers are trying to solve this problem in a different direction. Some of these papers are comparing smartphones hardware components power consumption through measuring and comparing an energy consuming entities such as wireless air interfaces, display, music player and others who trying to develop and modified the android java code. as mentioned in [1, 4, 10, 11, 12, 13, 14]. Others studies are tried to solve that paper through developing or designing an offloading framework and using other offloading techniques which are a promising way to improve and to reduce the battery power consumption of a smartphone application. That can be achieved by executing some parts of the application on a remote server decide at runtime [4, 5, 6, 7, 8, 9, 15]. The proposed system in this paper most closely related to Chun et al. [3], because of sharing some of the objectives and focusing on mobile applications. However focusing more on offloads the jobs Instead of the whole app.

3. PROPOSED METHODOLOGY

The proposed system comes with better power saving performance. it composed of one external smartphone client and two front-end server(private cloud and public cloud).The client application modified job list and Synchronizing that list with each cloud list in order to determine the non-critical files (jobs) which is want to be downloaded through that system. The following subsections discuss each step in details.

3.1 Proposed System Software Components

3.1.1 Client (Android App)

- The main function for that part of proposed system is to link the local smart device with clouds. That will to update (add or remove) downloading jobs and to choose either to download them using proposed offloading framework or by direct download through any available data network And receiving downloaded file through WI-FI connection when it connected at the same data network with private cloud.

- Technologies used to design that app are Java android programming language, Eclipse Android emulator and editor, Version: 3.7 Indigo Service Release two with Java runtime environment (JRE). Also, AVD Manager and Android SDK that provides Application programming interface (API) libraries and developer tools necessary to build, test, and debug apps for Android.

- As shown in Figure 2 this framework includes of several parts and layers. The first part is the App power consumption adapter and this part would determine the amount of activity applications and energy consumption as well as it works as a decision maker to determine the method of loading. The second part is smart device sharing resources which include (RAM, CPU and storage). The third part is synchronization layer which is obviously responsible about all synchronization operations at that framework. The fourth part is private cloud handler which is responsible for executing and managing connection form with private cloud. The fifth part is public cloud handler which is doing Formulation of the updated version of the job's list and governs the relationship between the public cloud and local smart device. The final part of that framework is the power consumption manager which working as an interface layer for proposed application framework to related private cloud and public cloud.

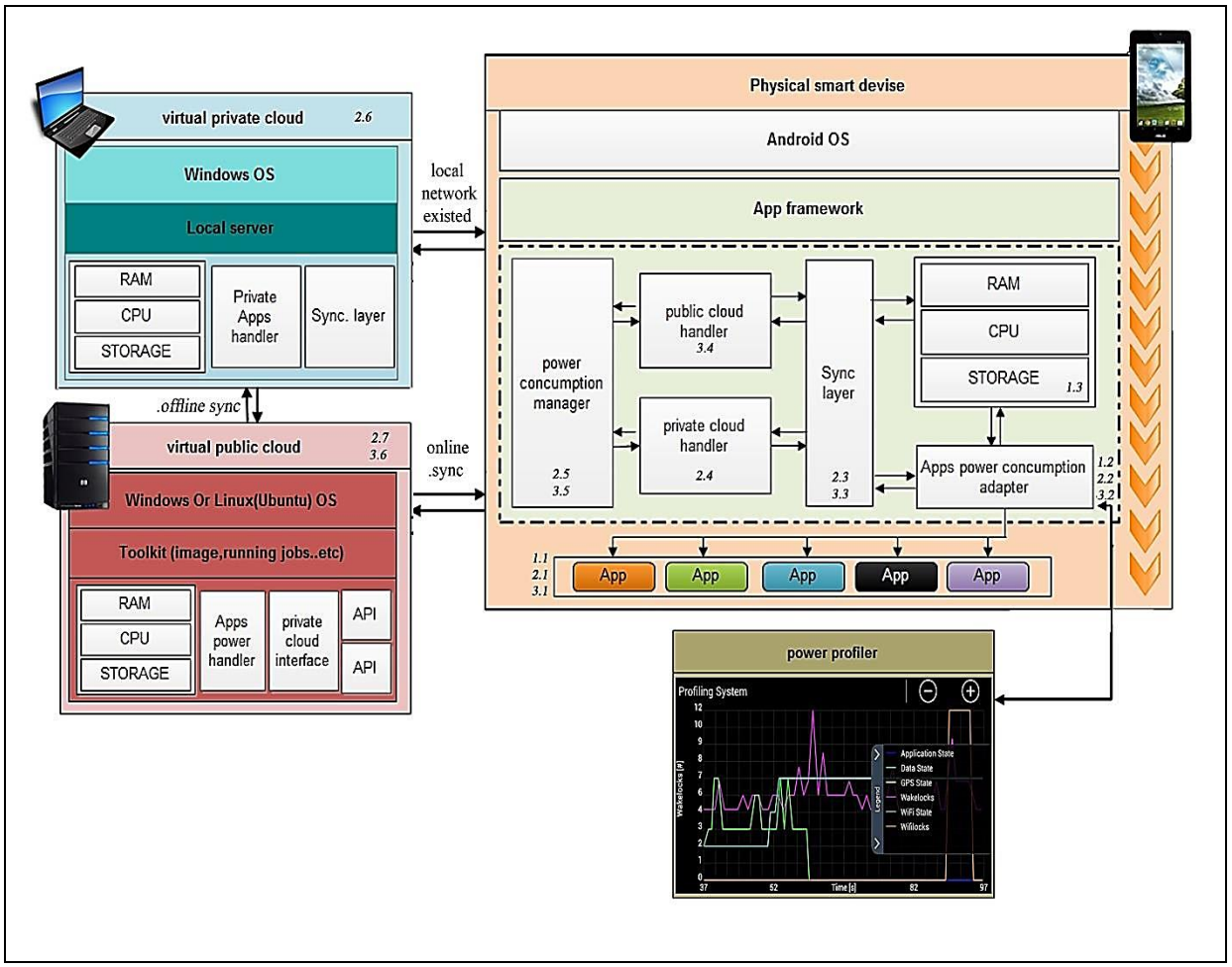


Figure 2: Proposed system prototype

3.1.2 Public cloud

- The main function for that part of proposed system is to creating, Operating and managing virtual image of the local smart device jobs. It is also responsible for updating and Synchronizing download jobs list and download selected jobs waiting to be sent to a private

cloud when internet exists. Then send it back to the client device as soon as it connected with private cloud at the same sustainable data network.

- Technologies used to design that cloud are PHP (server-side scripting language), MySQL (open-source relational database management system (RDBMS)), NetBeans (integrated development environment (IDE)), and XAMPP, Apache HTTP Server, Navicat, Symfony PHP web application framework and Doctrine (PHP) library.

- As shown in Figure 2 public cloud includes of several parts and layers. The first part is windows OS layer which operating server. The second part is a toolkit which is responsible for managing virtual image. The third part is server sharing resources which include (RAM, CPU and storage). The fourth part is Apps power handler which is responsible for managing apps downloading jobs and supervising its progress. The fifth part is private cloud interface which is responsible for managing the connection with private cloud and updating the job's list. The final part of that framework is APIs that governs the form that public cloud components should interact with each other.

3.1.3 Private cloud

- The main function for that part of proposed system is updating and Synchronizing download jobs list and received download files from public cloud and saving it as temporary files. Then send it back to the client device as soon as it connected with it at the same sustainable data network.

- Technologies used to design that cloud are JavaScript programing language, CSS, NetBeans (integrated development environment (IDE)), and Notepad++ source code editor.

- As shown in Figure 2 private cloud includes of several parts and layers. The first part is windows OS layer which operating server on the local server on laptop or desktop computer. The second part is a local server. The third part is computer sharing resources which include (RAM, CPU and storage). The fourth part is a private Apps handler which is responsible for managing power consume for android app image over downloading. The fifth part is private cloud interface which is responsible for managing the connection with private cloud and updating the job's list. The final part of that framework is APIs (Application programming interface) that governs the form that public cloud components should interact with each other.

3.2 Proposed System Hardware Components

The framework, presented in this paper implemented with three devices. The first one is a laptop computer with the following specifications: Intel® Pentium® processor B950 (2.1 GHz, 2 MB) with 6GB RAM DDR3, 1333 MHz, Video Graphics Intel® HD Graphics3000, Video Memory up to 1695 MB and Integrated WLAN Intel® Centrino® Wireless N1030 (IEEE 802.11b/g/n). This machine equipped with operating system Windows® 7 Ultimate 64-bit service back 1. The second device which been used is an Android smartphone with the following specifications: is

equipped with operating system Android OS, v2.3.6 (Gingerbread), CPU: Dual-core 1 GHz Cortex-A9, internal memory 8/16 GB storage, 768 MB RAM, 2 GB ROM, Li-Ion 1500 Milli Amp Hour (mAh) battery, compatible with 3G cellular network: HSDPA 850 / 900 / 1900 / 2100,Wi-Fi 802.11 a/b/g/n, dual-band, DLNA, Wi-Fi hotspot. The third device which been used is a 150Mbps Wireless N ADSL2+ Modem Router with the following specifications:4 10/100Mbps RJ45 Ports, 1 RJ11 Port, This router is equipped with Wireless Standards IEEE 802.11g, 802.11b, with some 802.11n features and Frequency run to 2.400-2.4835GHz.

3.3 proposed system working scenarios

This system exploiting one of the functions that are performed daily through smart devices namely download function. This function existed in any desktop computer, laptop computer and Smart device. Figure 2: shows the prototype of the proposed system, describing the system main components (client device, private cloud and public cloud). It determines three possible scenarios for downloading file as in below.

3.3.1 The first scenario

This scenario assumes that there is available sustainable wireless network (WI-FI) connecting both client device and private cloud and internet service. The first scenario starts with apps that need a file to be downloaded. The Apps send a request to App power consumption adapter which is connected to power profiler .this adapter is working as decision layer decide which file is downloaded locally and which one is downloaded through proposed framework and clouds automatically or manually as done in proposed prototype. Then moving to the synchronization layer, which is responsible for all synchronization operations (job list, job status). Then moving to private cloud handler, which is responsible for executing and managing connection form with private cloud. It's also formulation of the updated version of the list of jobs. Finally moves to the power consumption manager which working as an interface layer for proposed application framework to related private cloud which is connecting offline with public cloud.

3.3.2 The second scenario

This scenario assumes that there is no sustainable wireless network (WI-FI) connecting both client device and private cloud. However, there is an internet service. This scenario starts with the same previous steps until reaching the synchronization layer, which is responsible for all synchronization operations (job list, job status). Then moving to public cloud handler to do Formulation of the updated version of the job's list, in order to post it finally to the power consumption manager and send selected jobs to the public cloud directly. Public cloud should download those jobs to the private cloud and waiting for sustainable connection between the last and client device to finally deliver those files.

3.3.3 The third scenario

This scenario assumes that there is no sustainable wireless network (WI-FI) connecting both client device and private cloud. However, there is an internet service. It starts with apps that need a file to be downloaded. The Apps sends a request to App power consumption adapter which is connected to power profiler .this adapter decide automatically or manually downloading those jobs locally using available data network using device local resource.

3.4 Proposed system implementation steps

Figure 3 shows the general flow diagram of the implementation steps for proposed system, which comes with better classification performance. The first step in the process is running all system associated parts and then moving to the second step which should be include the installation process of the proposed Android application in the smart device and make sure it is ready for action. The second step should create an account on a public cloud plus create and install a virtual image of a smart local running Android. The fourth step process of installing a private Cloud on Desktop or laptop computer which is available as an infrastructure for this system. The fifth step in the process is creating a job list and synchronized between the client application in smart device, private cloud and public cloud. In the sixth step is the addition of a new job by adding job name and address (Uniform resource locator URL). In the seventh step, it highlights the fact that abstracts three main types of files that need to be downloaded frequently in smart devices as follows (operating system update file for android device client, android application update file and other files as (video files, audio files, other files). In the eighth step, a list of jobs will show the location of each file through the expression will be appeared in front of each file describing file (job) status. If the status is (**pending**), it means that the file is still in the phase of downloading between from public to private clouds. If the status is (**ready**), then it means that the file is already downloaded and exists temporarily in a private cloud waiting for a stable connection with the (client smart device) within the same data network in order to be sent. If the status is (**done**) then file downloaded successfully to its final destination (smart device), In step nine, an initialization the power profiler procedure starts to begin calculating the energy spent in the state of start downloading process and this depends on the question in step ten which Relation to the new files who show whether file is critical or not. The eleventh step will choose manually the way of downloading the file. After choosing the download process type flow will move to step twelve and that is where the stopping power profiler and followed step thirteen, which will report the energy consumption and there is the end of the process in step fourteen.

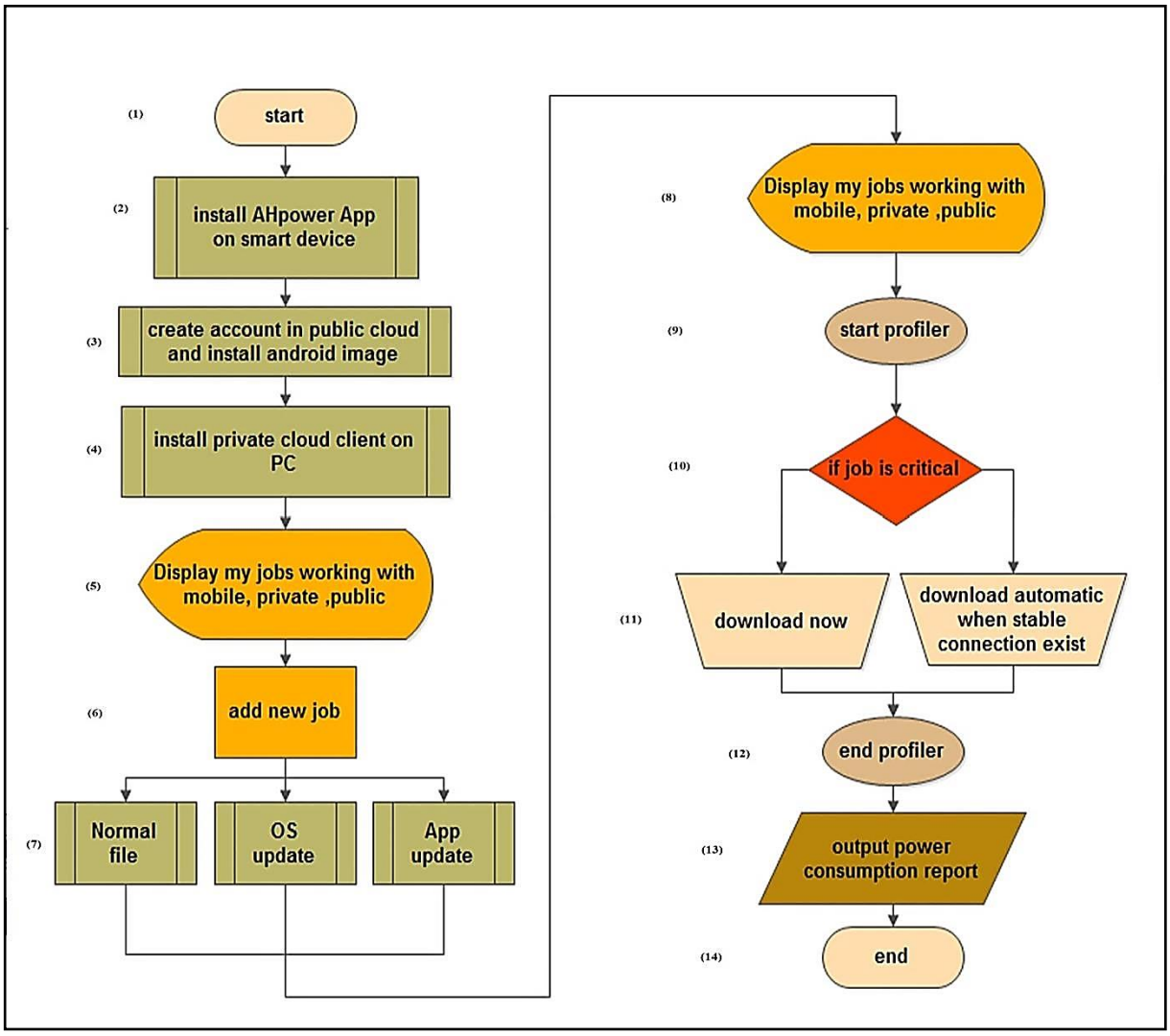


Figure 3: Proposed system implementation steps flow

4. EXPERIMENTAL RESULTS

4.1 Research Material

4.1.1 Used files

The proposed offloading framework (AHpower) described in chapter 3 has been tested using compressed Download Test Files from websites like http://www.thinkbroadband.com/download.html and other different types of files ex: (Audio, Video, RAR, Zip, pdf…etc.) also downloaded from web to compute downloading time and rate. The final database contains 40 downloaded files, each having different size between 1.25MB and 402MB. Files are downloaded in three downloading experiments,' each downloading experiment tested all 40 files with different experimental conditions as shown below in Table 1.

Table 1: Experimental used files

Sizes of downloaded Files used in each experiment in MB			
1.25	28.34	66.00	112.21
4.22	30.30	70.32	120.30
8.12	37.20	75.12	122.15
11.30	39.21	76.59	133.55
13.55	40.00	80.12	139.59
17.40	43.41	85.50	140.00
19.33	46.11	89.13	200.00
22.00	55.20	92.51	275.50
23.22	59.50	95.05	300.00
25.11	64.32	100.21	402.00

4.1.2 Used Devices

The framework, presented in this paper implemented with the same hardware components mentioned previously in 3.2**.**

4.1.3 Used Software

The proposed AHpower framework consist of three important components (client application (App), virtual private cloud and virtual public cloud) client App has been developed using several programming editors and emulators (Eclipse Android emulator and editor, software development kit (SDK)...etc.). Private cloud has been developed using (JavaScript language, CSS, NetBeans...etc.). Public cloud has been developed using (PHP, MySQL, Netbeans, Apache).All those software's Integrated together because of the high performance of its toolboxes And produce the AHpower prototype application compatible with android smart devices which are tested in this chapter.

4.1.4 Evaluation metrics

- **Objective:** the main objective of this work is to find a new idea to reduce battery power consumption in smart devices that are using Android as operating system.
- **Performance Measure:** the main scale Adopted to measure success or Failure of this work is Time spent in the job of downloading a file Thus the Spent energy during that time by PowerTutor profiler Which we will talk about in Comparison methods 4.1.5
- **Target:** creating offloading framework that or developing existed one to reach our objectives.

4.1.5 Comparison methods

In this chapter, an Android power profiler App measuring the power consumption for the smart device in the same time when downloading job occurred and gave the average of the consumed power for five minutes in Milli Watt (mW). Then demonstrate the effectiveness of the employed framework. PowerTutor profiler developed by the University of Michigan Ph.D. students. It allows seeing the impact of design changes on power efficiency, determining how actions are impacting battery life and monitoring the power consumption of any application. PowerTutor uses a power consumption model built by direct measurements during careful control of device power management states. This model provides power consumption estimates within 5% of actual values.

4.2 Experimental Results

In this section, an application to the power profiling is investigated to demonstrate the effectiveness of the employed system. In our experiments, we tested the whole files as a download jobs into three experiments. First experiment is using the proposed framework (AHpower) to execute downloading jobs. Second experiment is using the WI-FI to execute the same downloading jobs. Third experiment is using 3G cellular network to execute downloading jobs In order to make full use of the experiments and to evaluate the amount of power consumed in each experiment more accurately, then made a final comparison to prove our search point by results.

4.2.1 First experiment: download jobs using AHpower framework.

The first experiment was performed using different files to be downloaded for testing. By connecting the smartphone and running a private cloud on the laptop on the same local area network (LAN), a Synchronization start coordination of events to operate the system in unison, URLs of the files as downloading jobs, the results show a high performance. In contrast, the proposed system can be valuable when a huge amount of download jobs existed, which is important for classification make the suggested system outperforms other methods. Figure 4 shows the first experiment results using AHpower framework with WI-FI connection clarified that the average of total energy consumption in the smart device for five minutes = 1375 mW. That result should divide by 300 seconds as shown in equation (1) to get the energy consumption of every second, which is very critical to get accurate final results shown Table (2).

$$\text{Energy via WI-FI}=\frac{\text{Energy consumption value for N of minutes}}{\text{N of minutes}*60\ seconds} \quad (1)$$

Energy via WI-FI= 1375/5=275mW/minute, 1375mW/300sec =4.8 mW/sec.

Table 2: sample of downloaded files via proposed framework

Downloaded Job (file)	Via AHpower framework (APP)		
Size in (MB)	Time H,M,S,MS	Speed average Mb/s	Energy mW
1.25	00.00.00.11	100	0.52
4.22	00.00.00.36	100	1.72
8.12	00.00.00.67	100	3.21
11.30	00.00.00.95	100	4.56
13.55	00.00.01.13	100	5.42
17.40	00.00.01.46	100	7.01
19.33	00.00.01.62	100	7.77
22.00	00.00.01.85	100	8.88
23.22	00.00.01.96	100	9.41
25.11	00.00.02.10	100	10.08

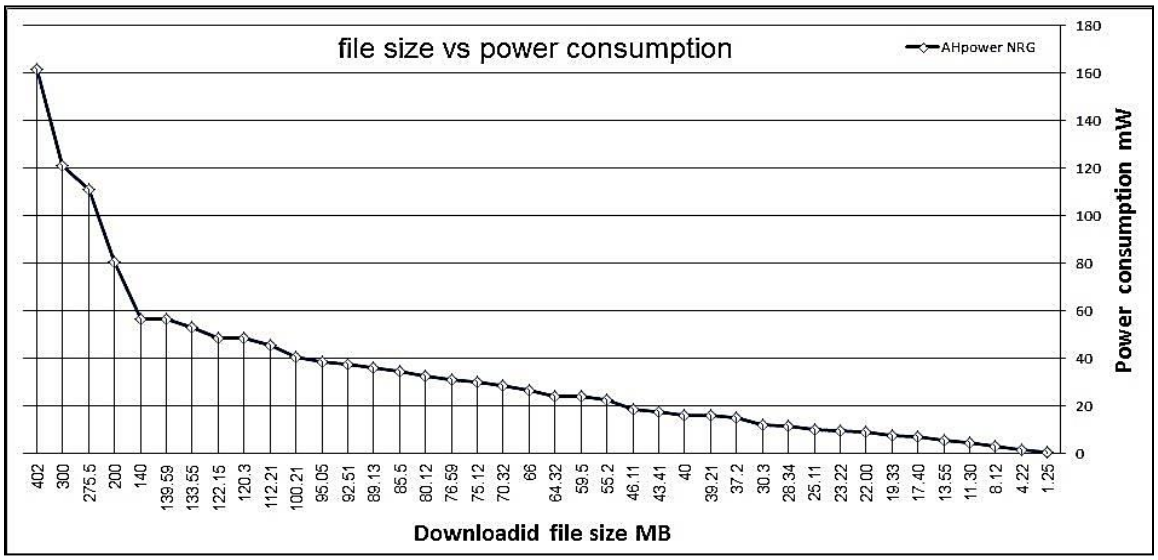


Figure 4: Experimental result using AHpower framework

4.2.2 Second experiment downloads jobs Wi-Fi network without AHpower

The second experiment was performed using the same files had been used in the first experiment to be downloaded for testing. Downloading the jobs take the exact first experiment steps but when putting the URLs but using the bottom (download without AHpower) when still connected to the internet through Wi-Fi. The results show the exact average of power but with a huge different in the downloading time (DLT). Figure 5 shows the second experiment results without using AHpower app with WI-FI connection clarified that the average of total energy

consumption in the smart device for five minutes matching the first result. As shown in equation (1), which is very critical to get accurate final results shown Table (3).

Table 3: sample of downloaded files via WI-FI

Downloaded Job (file)	Via (WI-FI)		
Size in (MB)	Time H,M,S,MS	Speed average Mb/s	Energy mW
1.25	00.00.05.11	2	24.48
4.22	00.00.17.27	2	82.89
8.12	00.00.33.26	2	156.32
11.30	00.00.46.28	2	222.14
13.55	00.00.55.48	2	266.30
17.40	00.01.11.27	2	342.10
19.33	00.01.19.18	2	380.06
22.00	00.01.30.14	2	432.67
23.22	00.01.35.11	2	456.53
25.11	00.01.42.85	2	493.68

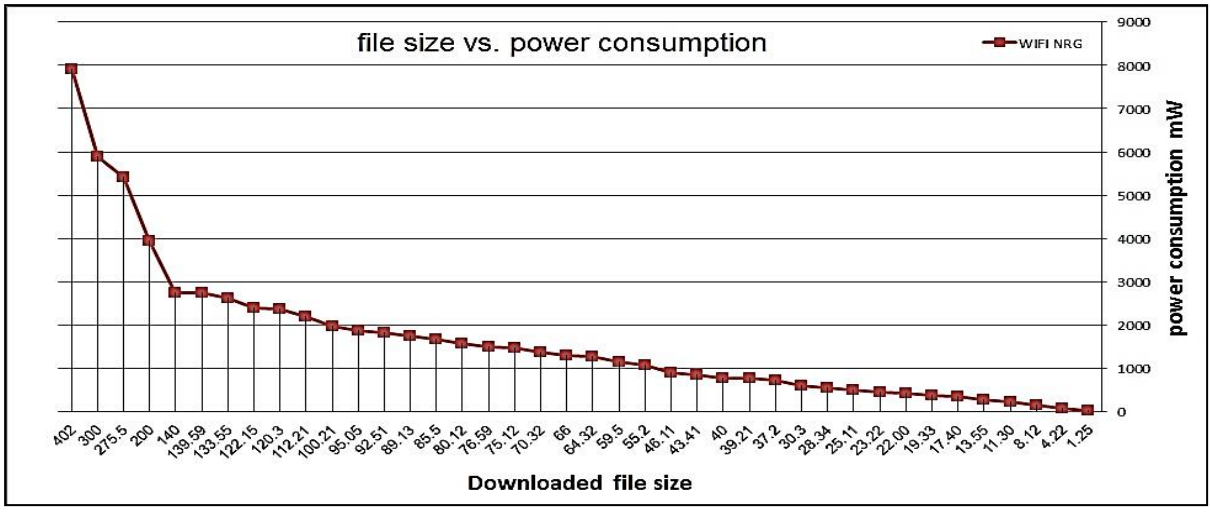


Figure 5: Experimental result using WI-FI connection

4.2.3 Third experiment: Trade-off between No. of training and test sets.

The third experiment was performed using the same files had been used in the previous experiments to be downloaded for testing. Downloading the jobs take the exact first experiment steps but when putting the URLs but using the bottom (download without AHpower) when smart device connected to the internet through 3G cellular network. The results showed anew average of power consumption for five minutes when 2203mW profiled

with nearly the same downloading time (DLT) in the second experiment. Figure 6 shows the third experiment results without using AHpower app with 3G connection clarified that the average of total energy consumption in the smart device for five minutes needed to calculate by new equation. As shown in equation (2), which is very critical to get accurate final results shown Table (4).

$$\text{Energy via 3G} = \frac{\text{Energy consumption value for N of minutes}}{\text{N of minutes} * 60\ seconds} \quad (2)$$

Energy via 3G = 2203/300=7.344mW/sec.

Table 4: sample of downloaded files via 3G

Downloaded Job (file)	Via 3G		
Size in (MB)	Time H,M,S,MS	Speed average Mb/s	Energy mW
1.25	00.00.05.12	2	37.60
4.22	00.00.17.29	2	126.83
8.12	00.00.33.29	2	244.48
11.30	00.00.46.30	2	340.02
13.55	00.00.55.50	2	407.59
17.40	00.01.11.31	2	523.70
19.33	00.01.19.20	2	581.64
22.00	00.01.30.11	2	661.77
23.22	00.01.35.13	2	698.64
25.11	00.01.42.89	2	755.62

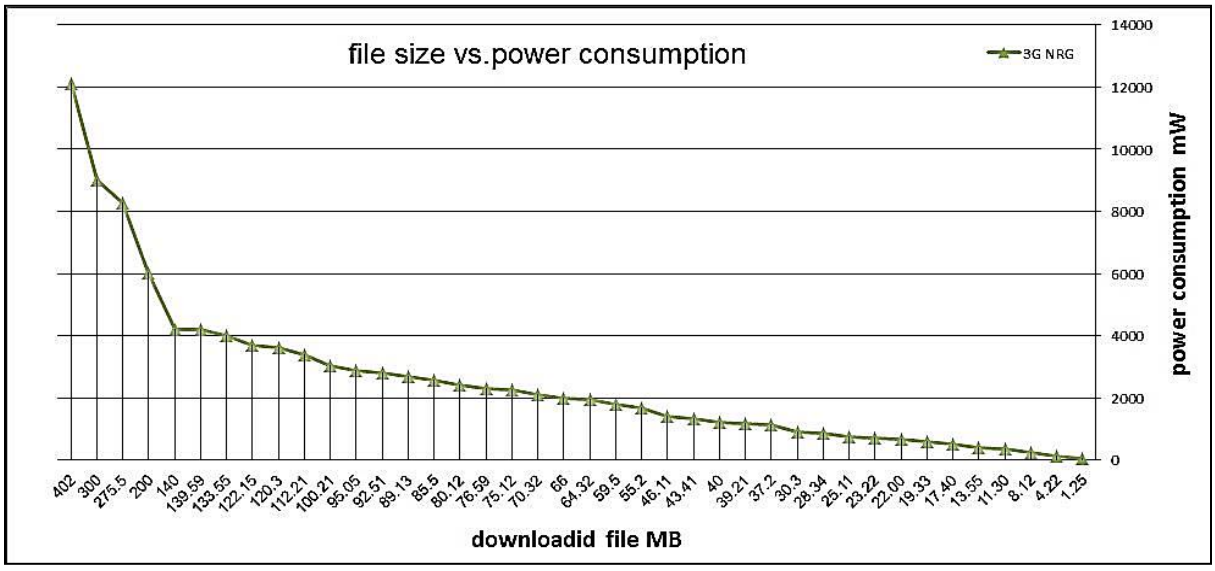


Figure 6: Experimental result using 3G connection

4.2.4 Comparison previous experiments

In AHpower framework … the key of success is the saving in downloading time. When that occurs a power consumption must be decreasing. The first and second experiments are using the same data network to connect internet (WI-FI), but a simple comparison operation declares the different time of downloading jobs results for the same files when the power consumption of WI-FI connection is almost stable with 4.8 mW/sec. Certainly the same could be said about the third experiment even with the used of different data network (3G) which consume 7.344mW/sec when downloading. As explained previously in the details of each experiment. Figure 7 show the linear comparison chart.

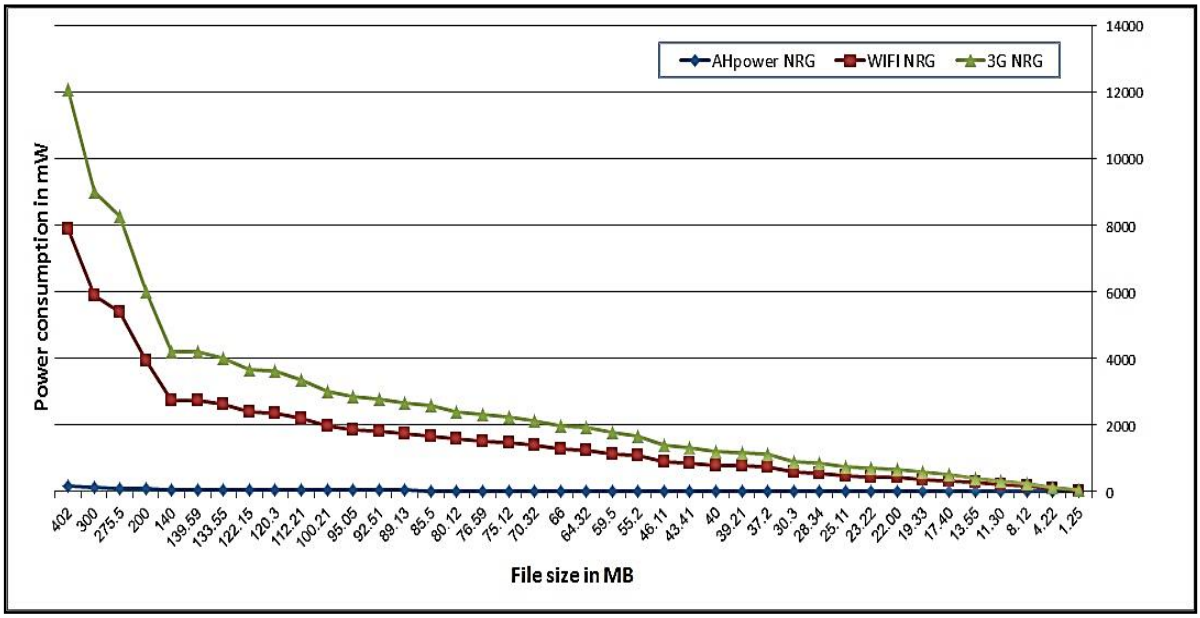


Figure 7: Final experimental comparison chart

Table 5: final power percentage results

Percentage of energy	Mathematical formula	consumption rate	Saving rate
AHpower/WI-FI	(X AHpower - X WI-FI)/ABS (X WI-FI)	2.01%	97.99%
AHpower/3G	(X AHpower - X 3G)/ABS (X 3G)	1.33%	98.67%
WI-FI /3G	(X WI-FI - X 3G)/ABS (X 3G)	65.1%	35.9%

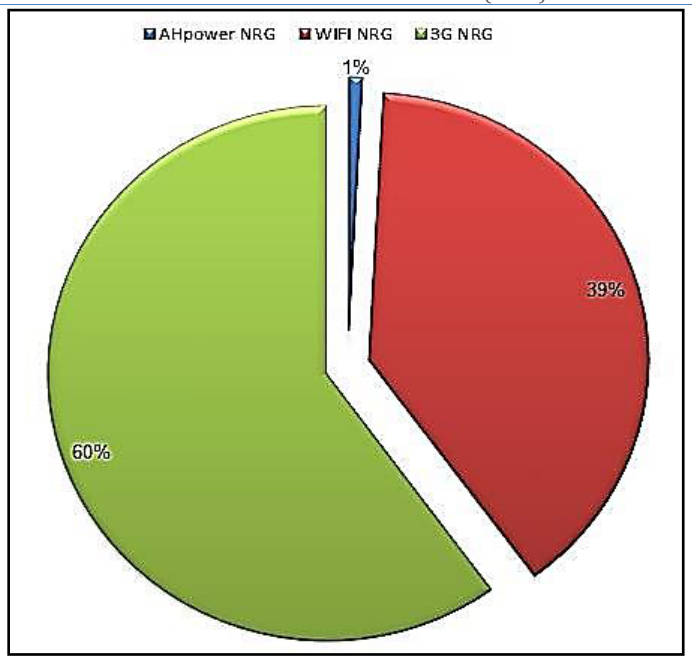


Figure 8: Experiments comparison for power consumption ratio.

5. CONCLUSION AND FUTURE WORK

The final result became clear through all the experiments that are described in detail in the previous section. It confirms the success of the proposed system. Where the energy consumption percentage calculated through the process of downloading the same files (download tasks). When using the proposed system AHpower vs. the same job using WI-FI it reached 2.01% of the original amount of energy, this means that the energy saved is equal to 97.99%. However when the connection occur through the third generation data network (3G), proportion of consumption reached only 1.33% and thus increased energy that was saved to the percentage of 98.67 % as shown below in Table 5. As shown these results characterized the success and the possibility of applying that idea. Many big manufacturers companies that produce smart devices with android operating system can benefit from it as mentioned before in Prospective audience section-chapter one. Figure 8 shows the final consumption percentage ratio for each experiment.

REFERENCES

[1]. G.P. Perrucci, F.H.P Fitzek, J. Widmer, ", Survey on Energy Consumption Entities on the Smartphone Platform", IEEE Computer Society 2011.

[2]. Ahmed E. Youssef ," Exploring Cloud Computing Services and Applications", Journal of Emerging Trends in Computing and Information Sciences,VOL. 3, NO. 6, July 2012.

[3]. Eric Y. Chen, and Mistutaka Itoh," Virtual Smartphone over IP", IEEE Computer Society 2010.

[4]. Aki Saarinen, Matti Siekkinen, Yu Xiao, Jukka K. Nurminen, Matti Kemppainen, and Pan Hui," Can Offloading Save Energy for Popular Apps?", MobiArch'12, , Istanbul, Turkey, August 22, 2012.

[5]. R. Kemp, N. Palmer, T. Kielmann, and H. Balm, "Cuckoo: a computation offloading framework for Smartphones", In Proceedings of MobiCASE, Oct. 2010.

[6]. Dejan Kovachev and Ralf Klamma, "Framework for Computation Offloading in Mobile Cloud Computing", International Journal of Artificial Intelligence and Interactive Multimedia, Vol. 1, № 7, 17.1.2012.

[7]. Eduardo Cuervoy, Aruna Balasubramanianz, Dae-ki Cho, Alec Wolmanx, Stefan Saroiux, Ranveer Chandrax, and Paramvir Bahlx, " MAUI: Making Smartphones Last Longer with Code Offload", MobiSys'10, San Francisco, California, USA, June 15–18, 2010.

[8]. Ying Zhang, Gang Huang, Xuanzhe Liu, Wei Zhang, Hong Mei, Shunxiang Yang," Refactoring Android Java Code for On-Demand Computation Offloading", OOPSLA'12, Tucson, Arizona, USA, October 19–26, 2012.

[9]. Narendran Thiagarajany, Gaurav Aggarwal, Angela Nicoara" Who Killed My Battery: Analyzing Mobile Browser Energy Consumption", WWW 2012, Lyon, France, April 16–20, 2012.

[10]. L.F. Pau" Energy Consumption Effects of WI-FI Off-Loading Access in 3G or LTE Public Wireless Networks", International Journal of Business Data Communications and Networking, 9(2), 1-10, April-June 2013.

[11]. Niranjan Balasubramanian Aruna Balasubramanian Arun Venkataramani," Energy Consumption in Mobile Phones: A Measurement Study and Implications for Network Applications", IMC'09, November 4–6, 2009, Chicago, Illinois, USA.

[12]. Goran Kalic, Iva Bojic and Mario Kusek," Energy Consumption in Android Phones when using Wireless Communication Technologies", MIPRO, 2012 Proceedings of the 35th International Convention, 21-25 May 2012.

[13]. Rahul Murmuria, Jeffrey Medsger, Angelos Stavrou, Jeffrey M. Voas," Mobile Application and Device Power Usage Measurements", Software Security and Reliability (SERE), 2012 IEEE Sixth International Conference - USA, 20-22 June 2012.

[14]. Ahmed Abdelmotalib, Zhibo Wu," Power Consumption in Smartphones (Hardware Behaviorism) ", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 3, No 3, May 2012.

[15]. Sokol Kosta, Andrius Aucinas , Pan Hui, Richard Mortier, and Xinwen Zhang," ThinkAir: Dynamic resource allocation and parallel execution in cloud for mobile code offloading", 1Google's 2011 Q2 earnings call.

TNC Transactions on Networks and Communications
SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
VOLUME 2, ISSUE 3
ISSN: 2054 - 7420

Feature selection using closeness to centers for network intrusion detection

[1]S. Sethuramalingam,[2]Dr. E.R. Naganathan
[1] Department of Computer Science, Aditanar College, Tiruchendur , India
[2]Department of Computer Science, Hindustan University, Chennai , India
[1]seesay@rediffmail.com ; [2]ern_jo@yahoo.com

ABSTRACT

Classification in intrusion detection data set becomes complex due to its high dimensionality. To reduce the complexity, significant attributes for classification called as features in the data set needs to be identified. Numbers of methods are available in the literature for feature selection. In this paper, a new algorithm based on closeness of points to its center is proposed. It is tested with NSL-KDD data set. The algorithm shows better result.

1. INTRODUCTION

Internet provides a lot of services to human being at the same times unauthorized persons can try to access it. It questions the security of the internet. Although many static defense mechanisms are available such as firewalls even the better dynamic security system is needed[1]. The intrusion detection is played a major role. Intrusion detection is based on the principle that intruder features are different from the normal features [2]. It can be divided into two general types known as anomaly detection and misuse detection [3]. Anomaly detection detects threats by judging whether the activity deviate significantly from the known normal behavior. Misuse detection detects threats based on whether the signature of the behavior matching a known threat

pattern or not[4]. In the network based intrusion detection system, a huge amount of data is collected. The data are summarized as connection records. It has number of attributes to describe the record. Therefore the data set is high dimension and volumes of records also large.If the data set directly is used by any machine learning algorithm, it becomes difficult to get expected results since it contains many redundant records.

DOI: 10.14738/tnc.23.286
Publication Date: 12th June 2014
URL: http://dx.doi.org/10.14738/tnc. 23.286

2. RELATED WORK

In the network data set there are features can directly collected from the packets. Such features are referred as basic features example number of bytes, flag , service types etc., . some of the features are created by combing basic features. They are called derived features. The goal of using Derived Features is to find similarities that exist between different TCP connections in the

Network [5]. In order to compute those features, two types of sliding window intervals are used. Time Based Features: are all the derived features computed with respect to the past x seconds, where x is the size of the time window interval [6,7,8,9]. Connection Based Features: are all the derived features computed with respect to the past k TCP connections that were encountered in the network. In [10] authors proposed a feature selection algorithm based on gain ration and correlation. The C4.5 tree uses gain ratio to determine the break and to select the important features. Genetic algorithm is applied as search method with correlation as fitting function.

Li et al. [11] utilized Kmeans clustering to assign the data of each class to k clusters, and then used the new dataset consisting of only the centers of clusters to train SVM, in which k is the upper bound of the number of support vectors in each class. In [12] authors proposed a hybrid based algorithm for future selection which is based on information gain and genetic algorithm. In the proposed algorithm features are selected based on how many values or points closer to its centers. After computing this values, features are selected.

3. PROPOSED ALGORITHM

In this algorithm center of each feature for anomaly class and normal class are computed. Number of values of anomaly closer to anomaly center and similarly number values of normal closer to normal center are computed. Now for each feature number of points closer to its center is known. These values are used to form feature set. The following algorithm compute number of points closer to it center for each feature.

Algorithm closer_points(x,ano)

```
Let x be the given data set and ano is the number of anomaly records
Let s1 and s2 be the number of records and number of columns of x
fp←0;fn←0;tp←0tn←0;
for j=1:s2
   aavg(j)←mean(x(1:ano,j));
   navg(j)←mean(x(ano+1:s1,j));
end
for j=1:s2
   adavg←0;
   ndavg←0;
   for i=1:s1
       adavg←abs(aavg(j)-x(i,j));
       ndavg←abs(navg(j)-x(i,j));
    if i<=ano
      if (adavg<ndavg)
          tp←tp+1;
```

```
      else
         fn←fn+1;
      end
     elseif (i>ano)
      if(ndavg<adavg)
         tn←tn+1;
      else
         fp←fp+1;
      end
     end
    end
   end
```

4. EXPERIMENT RESULTS

In order to remove the influence of dimensions, the data set is standardized [13]. The data set is standardized by subtracting a measure of central location i.e. mean and divided by some measure of spread such as standard deviation. The algorithm closer point is executed with 1000 training data set from NSL-KDD[14] . The results are given below in the table 1.

Table 1. Number of features values closer to its center

Feature no.	Number of normal values closer to anomaly center (Fp)	Number of anomaly values closer to normal center (fn)	Number of anomaly values closer to anomaly center (tp)	Number of normal values closer to normal center (tn)
f3	126	131	219	524
f4	40	63	287	610
f5	11	344	6	639
f6	470	7	343	180
f23	63	116	234	587
f24	524	26	324	126
f25	19	140	210	631
f26	15	140	210	635
f27	28	270	80	622
f28	26	271	79	624
f29	38	83	267	612
f30	43	247	103	607
f31	491	23	327	159
f32	303	50	300	347
f33	161	22	328	489
f34	131	48	302	519
f35	57	299	51	593
f36	125	274	76	525
f37	132	316	34	518
f38	10	140	210	640
f39	4	141	209	646
f40	36	265	85	614
f41	34	271	79	616

From the table features f3,f4,f5,f6 and f38 i.e. service, flags, src_bytes, dst_bytes and dst_host_serror_rate are selected for classification.

The classification algorithm uses 8990 record as training data set and 999 records as testing data set. The testing data set has 470 anomaly records and 529 normal records. The classification algorithm described below is used for classification. The proposed feature selection algorithm is compared with [12] developed by the authors. The results of three different algorithms are tabulated in table 2.

Algorithm fuzzy_compos(trn_amean, trn_astd, trn_nmean, trn_nstd, tstdataset)

Tstdataset: testing data set has m records and n attributes

Trn_amean: mean of anomaly class records in the training data set

Trn_astd: standard deviation of anomaly class records for the training data set

Trn_nmean: mean of normal class records in the testing data set

Trn_nstd: standard deviation of normal class records in the testing data set

```
for each connection record in the testing data set
 py1←1;  py2←2
   for each attribute in the connection record
       y(i,j)←gausmf(x(i,j),[trn_amean,trn_astd])
       y1(i,j)←gausmf(x(i,j),[trn_nmean,trn_nstd])
       py1←py1*y(i,j)
       py2←py2*y1(I,j)
  end
       by1(i)←py1;
       by2(i)←py2;
 end
  for each connection record in the testing data set
  f1(i) ←(by1(i)*trn_astd)+trn_amean;
     f2(i) ←(by2(i)*trn_nstd)+trn_nmean;
    if  (f1(i)>f2(i))
        if   i <= anolimit
          tp←tp+1;
       else
         fn←fn+1;
     end
          if  i> anolimit
            tn←tn+1;
          else
           fp←fp+1;
          end
      end
   end
```

Table 2: comparison of different featuring algorithm

S.No.	Feature selection Algorithm	Features selected	False positive (FP)	False negative (FN)	True positive (TP)	True Negative (TN)
1	Information gain	f1,f2,f3,f4,f5,f6,f23,f24,f25,f26,f27, f28,f29,f30,f31,f32,f33,f34,f35,f36, f37,f38,f39,f40,f41	112	62	408	417
2	Information gain and genetic algorithm	f3,f5,f6,f23,f24,f27,f28,f29,f30,f32, f33,f34,f35,f38,f37,f40	106	49	421	423
3	Number of closer point algorithm	f3,f4,f5,f6,f38	60	85*	385	469

From the table value for false positive (normal as anomaly) and false negative (anomaly as normal) are decreasing. Detection of anomaly (true positive) and Detection of Normal (true negative) are increasing. Therefore third one is performs better than the other two.

5. CONCLUSION

In this paper, a new feature selection algorithm is proposed and testing with testing data set. The results are better than the two algorithms. In the first one the information gain there are 25 features out of 41 features are used to get the result. In the case of information gain and genetic algorithm are used to select features 16 features out of 41 features are selected. In the proposed algorithm uses only five features out of 41 features to get the result. In future fuzzy distances based on closeness of the center can be proposed to improve the results.

REFERENCES

[1]. Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets H. Güneş Kayacık, A. Nur Zincir-Heywood, Malcolm I. Heywood

[2]. Dalhousie University, Faculty of Computer Science, 6050 University Avenue, Halifax, Nova Scotia. B3H 1W5

[3]. Christine Dartigue, Hyun Ik Jang, and and Wenjun Zeng, "A New Data-Mining Based Approach for Network Intrusion Detection," Seventh Annual Conununication Networks and Services Research Conference, May 2009.

[4]. Wenke Lee, Salvatore J. Stolfo, and Kui W. Mok, "A Data Mining Framework for Adaptive Intrusion Detection," Proceedings of the EEE Symposium on Security and Privacy, pp.I20-132, 1999.

[5]. Feature selection and design olintrusion detection system based on k-means and triangle area support vector machine Pingj ie Tang ,Rang-an Jiang, Mingwei Zhao Dept. Computer Science and Engineering Dalian University of Technology Dalian City, China 2010 Second International Conference on Future Networks Iosif-Viorel Onut and Ali A. Ghorbani “A Feature Classification Scheme for

Network Intrusion Detection" Faculty of Computer Science, University of New Brunswickm540 Windsor Street, Fredericton, New Brunswick, PoBox 4400, Postal Code E3B 5A3, CanadaInternational Journal of Network Security, Vol.5, No.1, PP.1–15, July 2007

[6]. P. Dokas, L. Ertoz, V. Kumar, A. Lazarevic, J. Srivastava, and P. Tan, "Data mining for network intrusion detection", in Proceedings of NSF Workshop on Next Generation Data Mining (Baltimore, MD), pp. 21-30, Nov. 2002.

[7]. L. Ertoz, E. Eilertson, A. Lazarevic, P. N. Tan, P.Dokas, V. Kumar, and J. Srivastava, "Detection of novel network attacks using data mining", in ICDM Workshop on Data Mining for Computer Security (DMSEC) (Melbourne, FL), pp. 30-39, Nov. 2003.

[8]. KDD, Kdd-cup-99 task description,The Fifth International Conference on Knowledge Discovery and Data Mining, http://kdd.ics.uci.edu/databases/kddcup99/task.html, last access Oct, 2005.

[9]. S. J. Stolfo, W. Lee and K. W. Mok, "Mining in a data-flow environment: Experience in network intrusion detection", in Proceedings of the 5 International Conference on Knowledge Discovery and Data Mining, pp. 114-124, 1999.

[10]. Asha Gowda Karegowda, A. S. Manjunath & M.A.Jayaram **"C**omparative study of attribute selection using gain ratio and correlation based feature selection International Journal of Information Technology and Knowledge Management July-December 2010, Volume 2, No. 2, pp. 271-277

[11]. Der-Chiang Li, Yao-Hwei Fang, "An algorithm to cluster data for efficient classification of support vector machines," Expert Systems with Applications, Elsevier, vol. 34, issue 3, pp.2013-2018, Apr 200S.

[12]. S.Sethuramalingam and E.R. Naganathan, "Hybrid feature Selection for Network Intrusion", International Journal of Computer Science and Engineering, vol. 3 No. 5 May 2011 pp 1773-1780

[13]. Hai Jin Jianhua Sun, Han Chen, Zongfen Han Cluster and Grid Computing Lab. Huazhong University of Science and Technolory, Wuhan 430074 China. "A Fuzzy Data Mining Based Intrusion Detection Model. Proc. Of the 10th IEEE International Workshop on Feature Trends of Distributed Computing Systems" (FTDCS'04)@2004 IEEE

[14]. http://nsl.cs.unb.ca/NSL-KDD/

TNC Transactions on Networks and Communications
SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
VOLUME 2, ISSUE 3
ISSN: 2054 - 7420

Route Reliability Modelling in Mobile ad hoc Network (R2M2) model using Bayesian algorithm

Pankaj Sharma[1] , Shruti Kohli[2] , Ashok K.Sinha[3]
[1&3]*Department of Information Technology, ABES Engineering College, Ghaziabad, UP, India*
[2]*Department of Computer Science & Engineering, BITEC, Noida, UP, India*
[1]sharma1pk@gmail.com; [2]shruti@bitmesra.ac.in

ABSTRACT

Mobile ad-hoc network technology has gained popularity in recent years by researchers on account of its flexibility, low cost and eases of deployment. The objective of proposed R2M2 model is to evaluate the performance of MANET (Mobile ad hoc Network) operating with DSR(Dynamic Source Routing) as routing protocol. The R2M2 model for MANET is simulated and implemented using network simulator ns 2.34 and validated using Bayesian rules. The R2M2 model is able to find out the probability of having certain behavior (able to decide reliable or not reliable) of routes, in uncertain observations (not able to decide whether the route is reliable or not) in presence of drop and delay. For performance evaluation of MANET using the proposed R2M2 model input variables (like node density, number of active connections, duration of communication, node movement speed, pause time & data transfer rate etc.) and output variables drop rate and delay have been taken as output variables. The R2M2 model helps in deciding the certain and uncertain behavior of routes and consequently the nodes and is found satisfactory.

1. INTRODUCTION

As the network is dynamic, the network topology continuously experiences alterations during deployment. The biggest challenge in MANETs is to find a route between communicating nodes and behavior of nodes forming the route, some of them are cooperating and some are non-cooperating or selfish nodes. A selfish node is a node that wants to save battery life for its own communication can endanger the correct network operation by simply not participating to the routing protocol or by not executing the packet forwarding (this attack is also known as the black hole attack) . Current ad hoc routing protocols cannot cope with the selfishness problem and network performances severely degrade.

In this paper performance of MANET is evaluated by implementing Dynamic Source Routing (DSR) protocol under different scenarios. DSR is an On-demand source routing

DOI: 10.14738/tnc.23.285
Publication Date: 12th June 2014
URL: http://dx.doi.org/10.14738/tnc. 23.285

protocol. In DSR the route routes are discovered after source sends a packet to a destination node in the ad-hoc network. The source node initially does not have a route to the destination when the first packet is sent. The DSR has two functions first is route discovery and the second is route maintenance [1, 2].

2. ROUTING IN MANET

A MANET routing algorithm should not only be capable of finding the shortest route between the source and destination, but it should also be adaptive, in terms of the changing state of the nodes, the changing load conditions of the network and the changing state of the environment. MANET routing algorithms can be classified into three categories as proactive, reactive or hybrid [3]. Proactive algorithms try to maintain up-to-date routes between all pairs of nodes in the network at all times. Examples of proactive algorithms are Destination-Sequence Distance- Vector routing (DSDV) and Optimized Link State Routing (OLSR) [4]. Reactive algorithms only maintain routing information that is strictly necessary: they set up routes on demand when a new communication session is started, or when a running communication session falls without route. Examples of reactive routing algorithms include Dynamic Source Routing (DSR) and Adhoc On-demand Distance-Vector routing (AODV) [5].

"Routing is the process of information exchange from one host to the other host in a network."[6]. Routing is the mechanism of forwarding packet towards its destination using most efficient route. Efficiency of the route is measured in various metrics like, Number of hops, traffic, security, etc. In Ad-hoc network each host node acts as specialized router itself [7].

Routing protocol for ad-hoc network can be categorized in three strategies.

a) Flat Vs Hierarchical architecture.
b) Pro- active Vs Re- active routing protocol.
c) Hybrid protocols.

Hierarchical network architecture topology consists of multiple layers where top layers are more seen as master of their lower layer nodes. There are cluster of nodes and one gateway node among all clusters has a duty to communicate with the gateway node in other cluster. In this schema there is a clear distribution of task. Burden of storage of network topology is on gateway nodes, where communicating different control message is dependent on cluster nodes.

But this architecture breaks down when there is single node failure (Gateway node). Gateway nodes become very critical for successful operation of network. Examples include Zone-based Hierarchical Link State (ZHLS) routing protocol [8]. Where in flat architecture there is no layering of responsibility. Each and every node does follow the same routing algorithm as any other node in the network.

In proactive routing scheme every node continuously maintains complete routing information of the network. This is achieved by flooding network periodically with network status information to find out any possible change in network topology.

Current routing protocol like Link State Routing (LSR) protocol (open shortest route first) and the Distance Vector Routing Protocol (Bellman-Ford algorithm) are not suitable to be used in mobile environment.

Destination Sequenced Distance Vector Routing protocol (DSDV) and Wireless routing protocols were proposed to eliminate counting to infinity and looping problems of the distributed Bellman-Ford Algorithm.

Examples of Proactive Routing Protocols are: [9].

a) Global State Routing (GSR).
b) Hierarchical State Routing (HSR).
c) Destination Sequenced Distance Vector Routing (DSDV).

Every node in this routing protocol maintains information of only active routes to the destination nodes. A route search is needed for every new destination therefore the communication overhead is reduced at the expense of delay to search the route. Rapidly changing wireless network topology may break active route and cause subsequent route search [8].

Examples of reactive protocols are:

a) Ad hoc On-demand Distance Vector Routing (AODV).
b) Dynamic Source Routing (DSR).
c) Location Aided Routing (LAR).
d) Temporally Ordered Routing Algorithm (TORA).

There exist a number of routing protocols of globally reactive and locally proactive states. Hybrid routing algorithm is ideal for Zone Based Routing Protocol (ZRP) [8][9].

3. DSR (DYNAMIC SOURCE ROUTING)

This is an On-demand source routing protocol. In DSR the route routes are discovered after source sends a packet to a destination node in the ad-hoc network. The source node initially does not have a route to the destination when the first packet is sent. The DSR has two functions first is route discovery and the second is route maintenance [10,11].

4. STEPS TO IMPLEMENT THE PROPOSED R2M2 MODEL

The algorithm implemented in this paper follows the steps:

1. Define the MANET with state input variables
2. Determine the scenarios using state variables

3. Simulate the behavior of MANET using scenarios in step (2)
4. Find out all possible routes
5. Compute the performance of route by considering the performance of each nodes participating in the route
6. Filter out behavior of routes based on result computed in step (6)
7. Validate the R2M2 model with bayes 'probability test

5. TOOLS & METHODOLOGY USED IN SIMULATIONS

In this paper we have used various tools such as network simulator version 2.34 (NS2.34) for getting the simulation results by writing and running the TCL script, applying the parameters in Table1, in addition we have taken the help of traffic generation tool such as cbrgen.tcl and mobile movement scenario generation tool such as Bonmotion 1.4, after getting the results.

For implementation DSR protocol in MANET environment, we prepared a scenario as shown in Table 1 and initial model of MANET is shown in fig. 1 which shows some input variables and output variable described as following:

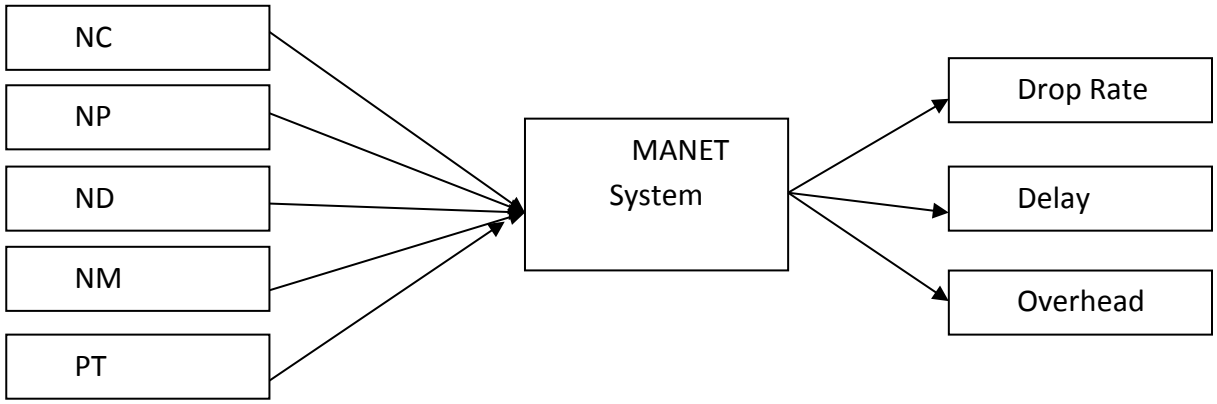


Fig. 1 Block Diagram of MANET system

Where NC: number of connections, PT: Pause time, ND:Node density, NM:Node mobility

PDF: Packet Delivery Fraction, NRL: Normalized Routing Load,AE2ED: Average End 2 End Delay.

Table 1. Scenario parameters

Simulation Parameters	
Routing Protocol	DSR
Mobility Model	RPGM
Simulation Time	10
Number of Nodes	5,10,15,20,25
Simulation Area	x=1000 m, y= 1000 m
Speed	5 m/sec
Pause Time	5
Traffic Type	CBR
Packet Size	512 bytes
Rate	5 packets/sec
Number of Connections	3,7,10,15,18
Seed	1.0

A scenario defined in Table1 is simulated with network simulator 2.34 and output is generated. Having analyzed the results, routes are computed and performance of routes is evaluated as dropped packet rate which is indirectly proportional to packet delivery ratio and delay in packet transmission, sample route performance is shown in Table 2.

Table 2. Occurrences of route1 with drop and delays

Possible occurrences of route1	Drop	Delay
1	0	14.57
2	0	14.85
3	34	9.53
4	36	9.53
5	0	8.57
6	23	13.85
7	29	9.53
8	0	8.53

6. PROPOSED R2M2 MODEL

The model is tested and validated on route 1 shown in table 2. Table 2 is obtained after simulating the scenario defined in table 1 using network simulator. Hence with respect to the results shown in table 2 , the population size of all possible outcomes of routes is 40 and out of which 65% (26) unique outcomes of routes are observed; assuming that unique outcomes are taken as sample space for the proposed model. In this paper we have randomly chosen a route

(route 1) for testing the behavior. The total occurrences of route 1 in sample space 30% (8) times having different behavior i.e in some occurrences it drops packets and in some occurrences it does not drop. The objective of this paper is to predict the behaviors of route1, whether the route's behavior is certain or uncertain. For testing and validating the behavior, some observations of route 1 have been taken with respect to drop and delay. Out of total observations, in 50% cases it drops packets and in remaining 50% it does not drop. If we cannot say that the drop is the only parameter for prediction of behavior of any route, and then based on the drop, it is difficult to determine the actual behavior of route. In presence of delay it may behave differently. For validating the behavior of route we have taken another parameter that is delay. In this paper initially we have chosen the threshold value of delay is 10 ms; although in realistic it may be very low.

Table 3: Behavior prediction table

Route1 (1-0-2)	**Delay>=10**	**Delay<10**	Marginal probability (**Total Probability)**
Case-1 (Packets dropping)	1	3	04
	1/4 = 25%	**03/4= 75%**	**4/8=50%**
Case- 2 (Packets not dropping)	2	2	04
Marginal probability	02/04=50%	2/04=50%	4/8=50%

A route in a network may comprise of two or more nodes and the behavior of a route depends upon the node's behavior and transmission medium. In this paper we primarily will focus only on node's behavior and later on transmission processing (delay). Initially after analyzing the behavior based up on the simulation output) we can categorise the node behavior in Mobile ad hoc network environment in two categories either having certain behavior (the nodes which do not loss the information) or having uncertain behavior (which may loss information).

The performance of MANET under operation of routing protocol depends upon the behavior of routes. In this paper the MANET is simulated under DSR (Dynamic Source Routing) protocol. The performance of MANET depends on delay and packet drop rate which effect the behavior of route, If we assume that the processing time of the packets in network is 10ms. thus after classifying the outcomes of the route including the delay parameter also, one category of outcomes is having delay >=10ms and another category is having delay <10ms . Both category is under consideration of zero drop rate i.e. 100% packet delivery ratio, the categorization is shown in Table 3.

This paper is focusing the behavior of a route (1-0-2) that has both type of behavior In MANET system, based on the scenarios defined in table1, it has been observed that in 50%

cases the route is having certain behavior, in case of either dropping minimum data packets or zero data packets and in rest of the 50% cases the observed route is having uncertain behavior, which means the route is dropping packets.

For validating the issues in deciding the actual behavior of route either certain or uncertain, the outcomes of the route are passed to Bayesian decision theorem; the base equation of bayes theorem is[12][13] :

$$\mathbf{P(A|B)} = \frac{\mathbf{P(A).P(B|A)}}{\mathbf{[P(A).P(B|A)]+[P(A').P(B|A')]}} \quad \text{.....(1)}$$

Where event A is the hypothesis and event B is the evidence.

Let X denotes certain behavior (not dropping packets) and X' denotes the uncertain behavior (dropping packets), D is the case where delay >=10ms and D' is the case where delay is <10ms. In this proposed model we need to test the following observations:

Prior assumption **is** that 50% of the total cases are having certain behavior so prior probability of randomly selecting a route and having certain behavior is P(X)=0.5. The goal is to find the probability that selected route is having certain behavior based on the observations shown in table 3; where 50% of positive cases of a route (positive cases are those having drop rate zero) having delay >=10ms, 50% of positive cases having delay <10ms, 25% of negative cases (negative cases which drop the packets) having delay >=10ms and 75% of negative cases having delay <10ms. We can say that P(X)=.5 (Since 50% of routes are having certain behavior), P(X')=.5 (Since 65% of routes are having uncertain behavior), P(D|X)=.5 and P(D|X') =.25.

If we compare the above scenario with the realistic scenario where delay should be minimum, assuming. For validating that the cases where route does not drop packet and delay in transmission is greater than 10 ms ,then for testing the behavior of route with the following equation (2):

$$\mathbf{P(X|D)} = \frac{\mathbf{P(X).P(D|X)}}{\mathbf{[P(X).P(D|X)]+[P(X').P(D|X')]}} \quad \text{.....(2)}$$

Now it has been observed that 66.6% results indicate for a route to be having certain behavior and 33.4% indicates for a route to be having uncertain behavior. For validating that the cases where route does not drop packet and delay in transmission is less than 10 ms ,then for testing the behavior of route with the following equation (2):

Where P(X)=.5 (Since 50% of routes are certain), P(X')=.5 (Since 65% of routes are uncertain), P(D'|X) =.5 and P(D'|X') =.75. The following computation will help to find out the probability of the positive cases having delay <10ms

$$\mathbf{P(X|D')} = \frac{\mathbf{P(X).P(D'|X)}}{\mathbf{[P(X).P(D'|X)]+[P(X').P(D'|X')]}}. \quad \text{.....(3)}$$

Finally it has been observed that in 40% cases the behavior of route is satisfactory and in 60% the behavior is unsatisfactory.

Table 4 Behavior of route1 observed after validation

Behavior of route	probability
Probability of having certain behavior of positive cases of route when delay is high (>=10ms)	66.6%
Probability of having certain behavior of negative cases of route when delay is low (< 10ms)	40%
Probability of having uncertain behavior of negative cases of route when delay is low(< 10ms)	60%
Probability of having uncertain behavior of positive cases of route when delay is low >=10ms	33.4%

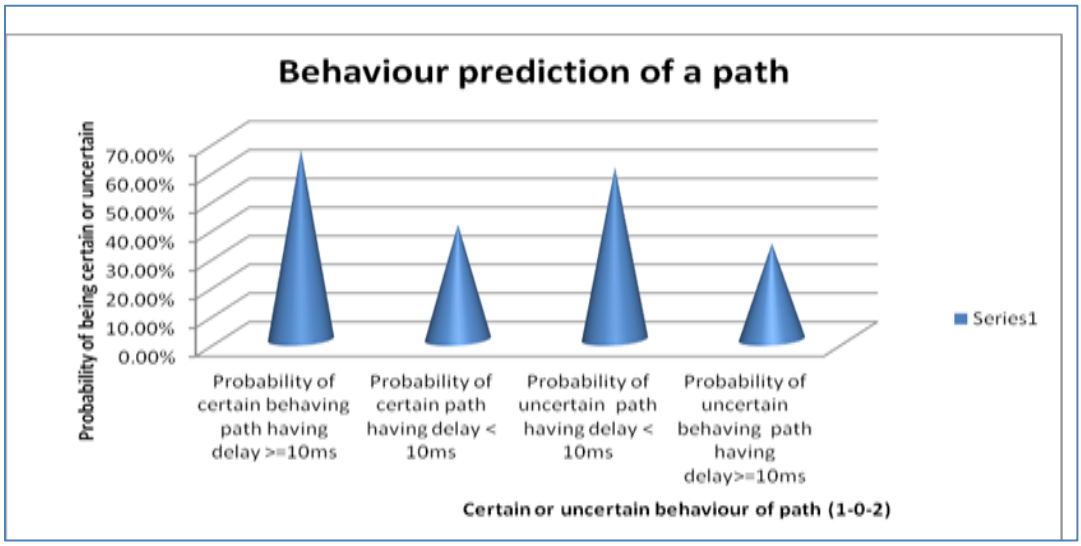


Fig. 2 : Validation Results comparision of occurance of rout 1

7. CONCLUSION

It has been concluded that the model is found satisfactory for reliability testing of a route. Figure 2 shows the graphical view of behavior analysis of route 1 and R2M2 model can be applied to any number of routes for behavior prediction. The model is able to find out that the probability of chosen route is 0.66 which indicates for a route to be having certain behavior and with probability .33 shows a route to be having uncertain behavior, while delay is greater than 10ms.

The probability of the cases, where route does not drop packet and delay in transmission is less than 10 ms, it has been observed that with probability .40 , it shows the behavior of route is reliable and with probability .60 the route is not reliable. Thus according to MAP (Maximum Posterior) the route will belongs to the class of reliability. The outcome of model is shown in table 4.

REFERENCES

[1]. Loutfi, Valerie, Bruno. "Securing mobile adhoc networks", MP71 project, 2003

[2]. Sergio Marti, T. J. Giuli, Kevin Lai, Mary Baker "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks", Proceedings of the 6th annual international conference on Mobile computing and networking, Boston, Massachusetts, 2000,Pages: 255 – 265

[3]. E.M. Royer and C.K. Toh. A review of current routing protocols for ad hoc mobile wireless networks. In IEEE Personal Communications, volume 6, April 1999.

[4]. T. Clausen and P. Jacquet. "Optimized link state routing protocol (olsr)". RFC 3626: Optimized link state routing protocol (OLSR), Oct 2003.

[5]. C. Perkins. "Ad hoc on-demand distance vector routing. Internet-Draft",draft-ietf-manet-aodv-00.txt, November 1997.

[6]. Humayun Bakht, *"Computing Unplugged, Wireless infrastructure, Some Applications of Mobile ad hoc networks",* http://www.computingunplugged.com/issues/ issue200410/00001395001.html, April-2003.

[7]. Charles E.Perkins and Elizabeth M. Royer, "*Ad hoc on demand distance vector (AODV) routing (Internet-Draft)*", Aug-1998.

[8]. Mario Joa-Ng, "*A Peer-to-Peer Zone-Based Two-Level Link State Routing for Mobile Ad Hoc Networks*", IEEE Journal on selected areas in communications, Vol. 17, No. 8, Aug-1999.

[9]. Padmini Misra, "Routing Protocols for ad hoc mobile wireless Networks", http://www.cse.ohio-state.edu /~jain/cis788-99/ftp/adhoc_routing/#TDRP, Nov-1999.

[10]. Loutfi, Valerie, Bruno. *"Securing mobile adhoc networks",* MP71 project, 2003

[11]. Sergio Marti, T. J. Giuli, Kevin Lai, Mary Baker *"Mitigating Routing Misbehavior in Mobile Ad Hoc Networks"***,** Proceedings of the 6th annual international conference on Mobile computing and networking, Boston, Massachusetts, 2000,Pages: 255 – 265.

[12]. Popper, K. (1959). The Logic of Scientific Discovery. Routledge, London, New York. St.Andrews (2003). Bayes. School of Mathematics and Statistics, University of St Andrews Scotland, http://www-gap.dcs.st-and.ac.uk/ history/Mathematicians/Bayes.html. Edited by John O'Connor and Edmund Robertson.

[13]. UCalgary (2003). Bayes Theorem. University of Calgary, Department of Mathematics and Statistics, Division of Statistics and Actuarial Science, http://balducci.math.ucalgary.ca/.

TNC Transactions on Networks and Communications
SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
VOLUME 2, ISSUE 3
ISSN: 2054 - 7420

Four Parallel Decoding Schemas of Product BlockCodes

Abdeslam Ahmadi[(1)], Faissal El Bouanani[(2)], Hussain Ben-Azza[(1)]
[(1)]Ecole Nationale Supérieure d'Arts et Métiers-Meknes, Morocco;
[(2)]Ecole Nationale Supérieure d'Informatique et d'Analyse des Systàmes-Rabat, Morocco;
ab_ahmadi@hotmail.com,elbouanani@ensias.ma,hbenazza@yahoo.com

ABSTRACT

This paper presents four new iterative decoders of two dimensional product block codes (2D-PBC) based on Genetic Algorithms. Each one runs in parallel on a number of processors connected by a network. As for the conventional iterative decoder, each elementary decoder of these new schemas uses as input, the received word and the extrinsic information computed by the previous elementary decoder. They have polynomial complexities in parameters of the code and those of the genetic algorithm. These are almost the same of the conventional iterative decoder complexity, but the performances are improved. Indeed, at each iteration, the new parallel decoders preserve the better of extrinsic information computed by elementary decoders running simultaneously on all processors.

Keywords: Error Correcting Codes, Product Block Codes, Genetic Algorithms, Parallel Decoding, Iterative Decoding, Time Complexity.

1. INTRODUCTION

In digital transmission, the information encoded in a binary sequence may be translated (modulated) into an analog signal to cross the communication channel, which is always noisy. The Noise due to channel parasites and modulation /demodulation processes can alter the useful signal. Likewise, the data stored in the storage media can be corrupted because of several factors (scratches, wear, etc). The encoder receives the information symbols provided by the source and adds redundancy symbols, carefully chosen, so that the maximum of infiltrated errors can be corrected. Upon arrival, the decoder attempts to restore the original sequence, using the redundancy symbols.

The analytical decoding techniques prove to be limited. Either they do not give satisfactory performance, or they are efficient, but require a high execution time and/or very large memory space like Maximum Likelihood decoding. This is why research in coding theory is oriented towards probabilistic, iterative, or meta-heuristic decoding techniques, where

DOI: 10.14738/tnc.23.229
Publication Date: 12th June 2014
URL: http://dx.doi.org/10.14738/tnc. 23.229

performances of some decoders approaching the Shannon limit [1]. This is the case for example for Turbo codes [2] and LDPC codes [3-4].

The challenge is to find methods which ensure a compromise between their correction capabilities and their complexities(acceptable execution time and memory space). Thus, in 1975, Holland introduced Genetic Algorithms(GAs) inspired by biological laws and natural selection [5]. They were then developed and popularized by Goldberg [6].

In 1994, Maini proposes a decoder based on GAs (GAD) giving good performances [8]. The works we have proposed in [9-12], have as objectives to improve performances and/or complexities of decoders based on GAs. In this paperwe propose four new parallelization schemas of two dimensional product block codes iterative decoding, where an elementarydecoder based on GAs is used [8].

This paper is organized as follows. Section 2 reminds some fundamental theoretical concepts. Section 3 presents elementary,iterative and parallel decoders that we use in the proposed schemas. Section 4 describes our parallelization schemas of aniterative decoding. In section 5 we discuss and study their time complexities. Finally, we give in section 6 our conclusions andperspectives of this work.

2. BACKGROUND

In In this section, we first make a quick reminder of product block codes. We then define the main classes of complexity, and we finish by presentation of genetic algorithms.

2.1 Product blockcodes

2.1.1 Linear block codes

Let $F_2 = \{0, 1\}$ be the binary alphabet and $(F_2)^n$be the set of vectors of length *n*. i.e. :

$$(\mathbb{F}_2)^n = \{x_1|...x_n / x_1, ..., x_n \in \mathbb{F}_2\} \tag{1}$$

A linear code C of length *n* on F_2 is a vector subspace of $(F_2)^n$. Such that the hamming distance between two different code-words is greater than the minimum distance of the code. i.e. :

$$\begin{cases} \forall\, x, y \in C \;/\; x \neq y,\; d_H(x,y) \geq d_{\min} \\ \forall\, x, y \in C,\; x + y \in C\;; \\ \forall x \in C, \forall \lambda \in \mathbb{F}_2,\; \lambda x \in C\;. \end{cases} \tag{2}$$

Note that in F_2, the second condition is equivalent to $0 \in C$. Let*k = dim(C)*,be the dimension of *C*, and $B = (g_i)_{1 \leq i \leq k}$a base of *C*.Since $g_i \in C$, then *length*(g_i) = *n*. The matrix *G* whose rows are the vectors of the base, is called the generator matrix of the code *C*. Note that the matrix *G*is not unique, since the base *B* is not. Thus:

$$G = \begin{pmatrix} g_1 \\ \dots \\ g_k \end{pmatrix} = \begin{pmatrix} g_{11} & g_{12} & \dots & g_{1n} \\ \dots & \dots & \dots & \dots \\ g_{k1} & g_{k2} & \dots & g_{kn} \end{pmatrix}$$

So we have $C=\{\alpha G/\alpha \in (F_2)^k\}$.i.e :

$$\forall x = (x_1, ..., x_n) \in C, \exists! \alpha = (\alpha_1, ..., \alpha_k) \in (\mathbb{F}_2)^k / \; x = \alpha G$$

$$\iff \quad (x_1, ..., x_n) = (\alpha_1, ..., \alpha_k) \begin{pmatrix} g_{11} & g_{12} & \dots & g_{1n} \\ \dots & \dots & \dots & \dots \\ g_{k1} & g_{k2} & \dots & g_{kn} \end{pmatrix}$$

$$\iff \quad \begin{cases} x_1 = \alpha_1 g_{11} + ... + \alpha_k g_{k1} \\ x_2 = \alpha_1 g_{12} + ... + \alpha_k g_{k2} \\ \dots \\ x_n = \alpha_1 g_{1n} + ... + \alpha_k g_{kn} \end{cases}$$

The parity check matrix H is a marix such that $GH^T = HG^T = 0$, where H^T is the transpose of matrix H. We have then $v = uG \Leftrightarrow vH^T = Hv^T = 0$. H has a crucial role in decoding. Indeed, the received word $v$is a codeword (without errors) if and only if $vH^T = 0$. It is said, in this case, that the word $v$satisfies the parity constraints of the code C.

The dual code of C, denoted by $C^T(n,n-k)$, is a linear block code whose generator matrix is H(parity matrix of C). Then, we can write:

$$C^T = \{y = (\mathbb{F}_2)^n \; / \; \forall x \in C, \quad < x, y >= 0\} \tag{3}$$

where$<x,y>$is the scalar product of $x$and y.

2.1.2 **Product block codes**

A product code is built from two or more elementary block codes, generally linears. Let $C_1(n_1, k_1, d_1)$and $C_2(n_2,k_2, d_2)$be two linear block codes. The product code $C= C_1 \otimes C_2$ is constructed as follows:

• The information symbols are arranged in a matrix of k_1 rows and k_2 columns ($k_1 \times k_2$ symbols) ;

• Eachone of the k_1 rows is coded by the code C_2;

• Eachone of the n_2 columns is coded by the code C_1.

Thus, a codeword of the product code $C$is a block of n_2 rows and n_1 columns ($n_2 \times n_1$ symbols). We show [13] that allrows are codewords of C_1 and all columns are codewords of C_2. Furthermore, the parameters of the product code $C(n, k, d)$are :

• $k= k_1 \times k_2$;

• $n= n_1 \times n_2$;

• $d= d_1 \times d_2$;

• $R = R_1 \times R_2$;

Where R_1,R_2, and $R$are respectively the rates of C_1,C_2, and C.

Product block codes represent a particular case of serial concatenated codes. Their highlight is that they allow the construction of codes of large lengths and large minimum distances, by concatenating two or more codes of small lengths and small minimum distances. Product code built with a large minimum distance will have then a large capacity for detection and correction of errors.

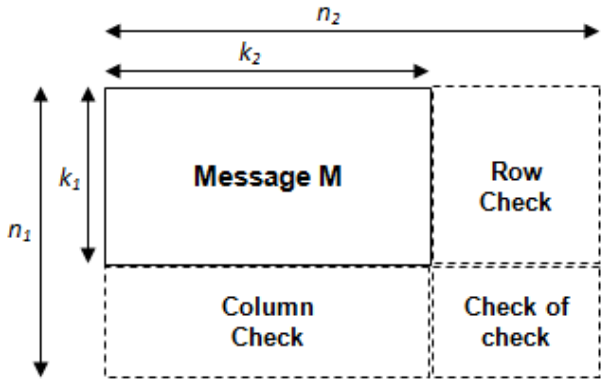


Figure 1: Product block code

2.2 Decoding Complexity

2.2.1 Complexity Classes

The complexity theory is a recent discipline that aims to classifying problems, according to their degree of difficulty of resolution. Several classes of complexity as well temporal as spatial have been defined. The best known are the classes P, NP and NP-Complete. Class P includes problems for which there exists an algorithm with a polynomial running time in the size of data, allowing solving them. The problems of this class are called "easy". The class NP contains all optimization problems where the number of potential solutions, is at worst exponential such that we can verify in a polynomial time whether a potential solution satisfies the question. A problem X of class NP is said NP-complete (or NP-hard), if each problem Y of class NP is polynomially reducible to X. i.e., there is a polynomial algorithm used to brought back the search of solution of Y to the search of solution of X. NP-Complete class contains all the problems of class NP such that if one of them is proven to be easy (solvable in polynomial time) then all NP problems are easy. If one of them is hard, then P≠ NP.

2.2.2 Complexity of linear codes decoding

In 1978, Berlkamp, McEliece and Van Tilborg [14] have stated two conjectures:

i) the problem of decoding linear codes is NP-complete;

ii) computing the minimum weight of a code is an NP-complete problem.

For a linear code, the second conjecture is equivalent to the conjecture of calculating the minimum distance. If this is calculated, the error correction capacity is then determined. This conjecture was open until 1997, when it has been proved by Vardy [15].

The fact that a problem is NP-complete causes its computational difficulty, i.e., the inefficiency of deterministic algorithms to solve it. Thus, mathematicians and computer scientist's recourse to meta-heuristic methods to find good solutions (not necessarily optimal) to NP-complete problems in polynomial time.

2.3 Genetic Algorithms

Genetic Algorithms (GAs) have been inspired in genetics and the theory of evolution of species, presented by Darwin in1860. It refutes the idea that the natural system is fixed forever. For him, the species are gradually adapting to their natural environment that could change depending on external parameters and constraints to which it is exposed. The AGs are designed to simulate processes of the natural systems required for evolution, especially those who respect the principle of "survival of stronger". The Strongest individuals will reproduced and their offspring are improved over generations. The lowest ones will disappear.

Researchers have tried to program and simulate natural phenomena since the 50s. However, these attempts were not very fruitful, because of the limitation of computer performance at this period. The use of GAs, made a big boom in the last three decades, when Holland has posed their theoretical foundations in 1975 in his book "Adaption in Natural and Artificial Systems"[5], and when Goldberg wrote in 1989 his famous book "Genetic algorithms in search, optimization, and machine learning"[6]. Thus we moved from natural Darwinism to artificial evolution, which began to be used increasingly in the solving of problems with very high complexity, in several fields.

2.3.1 Principle of GAs

GA generates randomly n individuals to form the initial population. For Every individual, which is a potential solution to the problem to be optimized, we associate a fitness (or cost) that measures its quality as solution. Then we select, with a probability that depends on the fitness, the best individuals that may be crossed to give birth of new individuals (children). These undergo, with a certain probability, to mutations in their genes. This forms the new population of the next generation. We repeat the same treatment until the stop condition is satisfied.

Here is the basic genetic algorithm, as shown in figure 2:

Algorithm: Basic GA

1. [Initialization]: generate a random population of n individuals :$(I_i)_{1\leq i\leq n}$;

2. [Fitness]: compute the fitness $f(I_i)$ for each individual I_i of the current population ;

3. [Reproduction]: create a new population by repeating the following steps:

- [Selection]: select two parents, taking into account their fitness;
- [Crossover]: cross, with a probability p_c, the parents to create new individuals ;
- [Mutation]: mutate, with probability p_m, the new individuals ;
- [Insert]: add new individuals to the new population;

4. [Replace]: use new generated population for the next iteration;

5. [Test]: if the stop condition is satisfied, then return the best individual of the current population;

6. [Loop]: else go to the step 2.

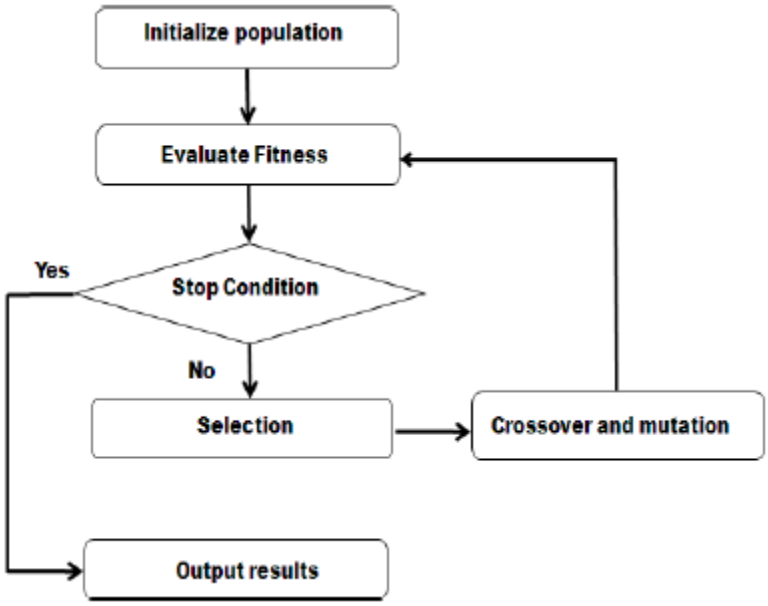


Figure 2: Genetic algorithm flowchart

2.3.2 **Convergence of GAs**

The researches consisting of laying the theoretical foundations of GAs are very rare. The reference works in this sense are those of Eiben et al. [16], Fogel [17] and Rudolph [18].They have, first, given a mathematical formulation of the basic genetic algorithm, modeled its evolution as a Markov chain, and finally defined sufficient conditions for its convergence toward an optimum. They have shown that the convergence to the global optimum is not an inherent property of the basic genetic algorithm, but a consequence of the idea of keeping track of the best solution found over time (from one generation to another). In other words, the basic GA can be considered an optimization algorithm for static optimization problems, because it is provable that it does not converge toward any subset of the set of states containing at least one global solution, even in infinite time.

2.4 Parallel Systems

The choice of distributed systems or parallel machines is strongly imposed for applications requiring very important treatment and/or memory space. There are many types of parallel machines which are classified by Flynn (1972) according to two independent concepts: the instruction stream and data stream used by these instructions. Thus, there are four possible combinations [7] :

- Single Instruction Single Data: the machine executes one instruction on a data each clock cycle. It's not really a parallel machine but rather a classical Von Newman computer;
- Single Instruction Multiple Data: a processor, with a single control unit (CU) and multiple arithmetic logic units (ALUs),executes the same instruction on different data each clock cycle ;
- Multiple Instruction Single Data: systems executing multiple instructions on the same data at each clock cycle
- Multiple Instructions Multiple Data (MIMD): these systems have multiple independent processors. i.e., each processor has its own CU and its own ALU. In the same clock cycle, each processor executes a different instruction on a different data.

These instructions can be synchronous or asynchronous. Most parallel systems are MIMD. A MIMD system can be either a multiprocessor, or a multi-computer or a hybrid of both.

A multiprocessor contains several autonomous processors and a single shared memory, figure 3. This last can also provide communication between different processors. A multi-computer contains a given number of autonomous computers, having a control unit, an ALU and a private memory each one, figure 4. The communication between the computers is provided by a network.

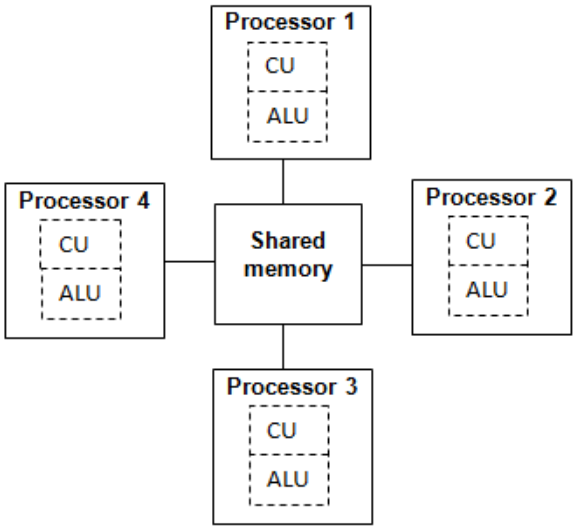


Figure 1:A Parallel system with 4 processors

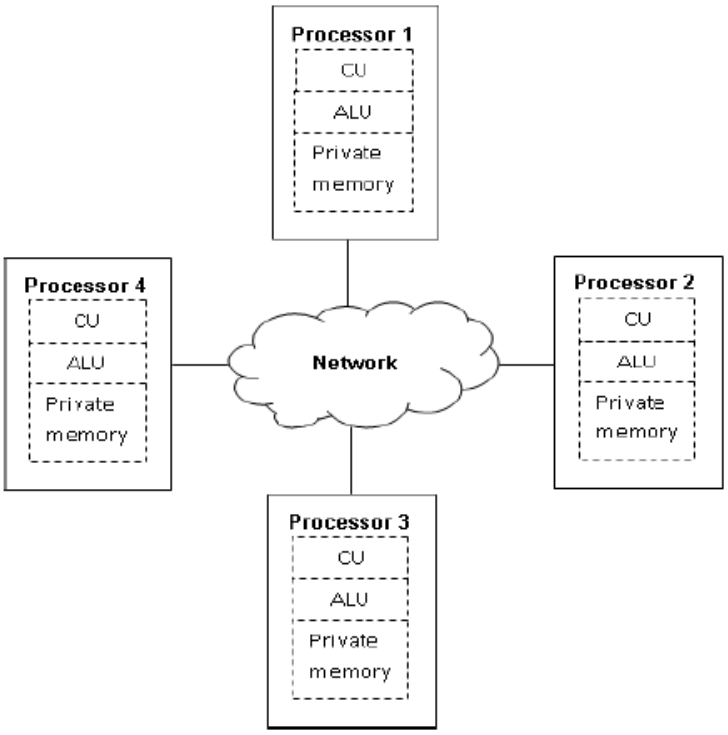


Figure 4:A multi-computer with 4 computers

A multiprocessor contains several autonomous processors and a single shared memory, figure 3. This last can also provide communication between different processors. A multi-computer contains a given number of autonomous computers, having a control unit, an ALU and a private memory each one, as shown in figure 4. The communication between the computers is provided by a network.

3. DECODING ALGORITHMS BASED ON GENETIC ALGORITHMS

Let $C(n,k,d)$ be a linear block code with generator matrix G, and $F = (F_1,...,F_n)$, $r = (r_1,...,r_n)$, respectively the vector of fading and the received sequence (associated with the transmitted sequence). The parameters N_p, N_g, N_e, p_c, $p_m$are respectively the size of the population, the number of generations, the number of elites, the crossover and the mutation rates.

3.1 Elementary decoder

We take GAD the decoder of block codes based on GAs that we have already used in [10-12]. It's a HISO (Hard-In Soft-Out) and uses GAs to decide the code word D, knowing r and F.

Algorithm : GAD

D = GAD (r, F,k,n,N_p, N_g, N_e, p_c, p_m)

- **Step1 :**sort the elements of received vector r in descending order of magnitude to have a new vector $r^{(1)}$. i.e., find a permutation π_1 such that $r^{(1)} = \pi_1(r)$ and $|r_1^{(1)}| \geq |r_2^{(1)}| \geq ... \geq |r_n^{(1)}|$.This will put, for BPSK modulation, reliable elements in the first ranks. Indeed, when the absolute value is large (far from 0), there is no risk to decode the bit 1 to 0 or 0 to 1. Let $F^{(1)}$ be the permutation of F by π_1. i.e., $F^{(1)} = \pi_1(F)$. Likewise, $G^{(1)} = \pi_1(G)$. Then, permute $G^{(1)}$ by π_2 to have G', such that its first k columns are linearly independent, permute the vectors $r^{(1)}$ and $F^{(1)}$ by the same permutation. Let $r' = \pi(r)$ and $F' = \pi(F)$, where $\pi = \pi_2 \circ \pi_1$.
- **Step 2** : quantize the first k bits of r' ($r'_i \in \mathbb{R}$) to obtain a binary vector I_1 and randomly generate ($N_i - 1$) information vectors of k bits each one. This vectors form with vector I_1 the initial population of N_p individuals (I_1, ...,I_{Np}).
- **Step 3** : encode individuals of the current population, using G' to obtain code words : $C_i = I_i G'$ ($1 \leq i \leq N_p$). Then, compute individuals fitness, defined as Euclidian distance between C_i and r', and sort individuals in ascending order of fitness.
- **Step 4** : place the first N_e individuals (N_e: elite number $\leq N_p$) to the next population, which will be completed by offsprings generated using reproduction operators : selection of two best individuals as parents (a,b) using the following linear ranking :

$$W_i = \frac{W_{\max} - 2(i-1)(W_{\max} - 1)}{N_i - 1}, \forall i \in \{1, ..., N_p\} \quad (4)$$

  whereW_i is the ith individual weight and W_{max} weight assigned to the fittest (nearest) individual.

- **Step 5** :Reproduce the (N_p- N_e) remaining individuals of the next population using crossover and mutation operations. Let *Rand* be a uniformly random value between 0 and 1, generated at each time.

  if$Rand < p_c$ then $\forall i \in \{N_e+1, ..., N_p\}$, $\forall j \in \{1, ..., k\}$,

$$I_{ij} = \begin{cases} a_j \ if \ \ Rand < (1 - a_j + a_j b_j \ ) + \frac{a_j - b_j}{1+e^{\frac{-4r'_j F'_j}{N_0}}} \\ b_j \quad else \end{cases} \tag{5}$$

and then,

$$I_{ij} = 1 - I_{ij} \quad if \ \ Rand < p_m \tag{6}$$

else,

$$I_i = \begin{cases} a \ \text{ if } \ Rand < 0.5 \\ b \ \text{else} \end{cases} \tag{7}$$

end if

Repeat steps 3 to 5 for N_g − 1 next generations.

- **Step 6** :The first (fittest) individual *D'* of the last generation is the nearest to *r'*. So, the decided code word is $D = \pi^{(-1)}(D')$

3.2 Iterative decoder

An iterative algorithm receives at its input soft information and produces another one called extrinsic information which depends on the decided code word. This extrinsic information will be combined with the received word and fed back to its input. The processing is repeated Nit times and the decided code word will be the one decided at the last iteration.

For product block codes, an iterative decoder consists of placing in series two or three decoders. The extrinsic information computed by the ith decoder is combined with the received word and the result is fed to the input of the (*i* + 1)th decoder.

The result of the combination of the extrinsic information of the last decoder and the received word is injected at the input of the first decoder. The process is repeated Nit times. The iterative decoder for product block codes with two dimensions is depicted in figure 5.

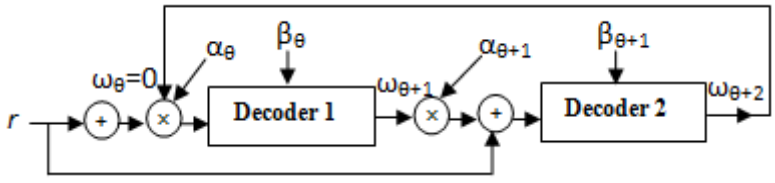


Figure 5:The iterative decoder based on AGs (IGAD)

Our iterative decoder here, is IGAD (Iterative decoder based on Genetic Algorithms), where the elementary decoders are GADs.

Let *D* denote the GAD decision of the input sequence *r*, and ω be the extrinsic information, and $H^{(j)}$ be the competitor codeword of *D* corresponding to the *j*th bit defined by :

$$\left\| H^{(j)} - r \right\| = \min_{2 \le p \le N_t} \left\{ \left\| Q^{(p)} - r \right\|, Q_j^{(p)} \neq D_j \right\},$$

where :

- $Q^{(p)}$ is the *p*th codeword of the last generation

- $Q_j^{(p)}, D_j$ are the *j*th bits of $Q^{(p)}$, *D*

- $\| . \|$ is the Euclidean distance.

The algorithm executed by each of the elementary decoders of the iterative decoder, accepts as input *r, k, n*, N_p, N_g, N_e, N_{it}, p_c, p_m, and coefficients $(\alpha_j)_{1 \le j \le 2Nit}$ and $(\beta_j)_{1 \le j \le 2Nit}$. These coefficients are optimized for each code and SNR to enhance the algorithm performance.

Algorithm : IGAD

(ω,*D*) = IGAD(*r, F,k,n,*N_p, N_g, N_e, N_{it}, p_c, p_m, α,β)

•**Step1** :θ = 0, ω_0 = 0 ;

•**Step2** :*Iterative decoding*

While (θ < 2(N_{it}− 1)) **do**

- Run D(θ) = GAD (r+ α(θ)ω(θ), F, k,n,Np, Ng, Ne, pc, pm) on the first decoder to decide the codeword D(θ) ;
- Compute the extrinsic information ω(θ+1) in terms of D(θ)

For *j* = 1 **to***n***do**

If $H^{(j)}$ *exists* **then**

$$\omega_j^{(\theta+1)} = \widetilde{D}_j^{(\theta)} \left[\frac{\left\| H^{(j)} - r \right\| - \left\| D^{(\theta)} - r \right\|}{4} \right] \quad (8)$$

$$= \widetilde{D}_j^{(\theta)} \sum_{\substack{p=1, p \neq j \\ H_p^{(j)} \neq D_p^{(\theta)}}}^{n} r_p \widetilde{D}_p^{(\theta)}$$

else

$$\omega_j = \beta \widetilde{D}_j^{(\theta)} \quad (9)$$

$$\widetilde{D}_j^{(\theta)} = 2D_j^{(\theta)} - 1$$

End if

End for

- Run $D^{(\theta+1)}$ = GAD ($r+ \alpha^{(\theta+1)}\omega^{(\theta+1)}$, F, k,n,N_p, N_g, N_e, p_c, p_m) on the second decoder to decide the codeword $D^{(\theta+1)}$;
- Compute the extrinsic information $\omega^{(\theta+2)}$ in terms of $D^{(\theta+1)}$
- $\theta = \theta + 2$;

End While

•**Step3** :*Decision*

✓ Select the codeword decided by the second decoder at the last iteration $D^{(2(Nit-1))}$

3.3 Parallel decoder

We give here the sub-algorithm of our parallel decoder PGAD, based on GAs that we proposed in [12]. It runs in parallel on each one of the N_s processors:

Algorithm : PGAD

D = PGAD (r, F, k,n,N_p, N_g, N_e, p_c, p_m,N_s, N_c)

•**Step1**:*Permutations and Initialization*

✓ Run the two first steps of the algorithm GAD

•**Step 2**:*Reproduction*

✓ Encode individuals of the current population to obtain code words $C_i = I_i G'$ ($1 \le i \le N_p$)

✓ Compute individual fitness, defined as Euclidian distance between C_i and r' :

$$f(C_i) = \sum_{j=1}^{n} (C_{ij} - r'_j)^2 , \forall i \in \{1, \dots, N_p\}$$

✓ Sort the current population individuals in descending order of their fitness ;

✓ $gen \leftarrow 0$ (gen is the current generation number).

While ($gen<N_g$) **do**

✓ Copy the $N_e$best individuals (elites) from the current population to the new one ;

✓ Select parents from the N_p–N_e individuals of the current population ;

✓ Quantize the first k bits of each selected parent ;

✓ Cross with probability p_c the selected parents to generate N_c new individuals of k bits

✓ Mutate the new individuals with a probability p_m if their parents are crossed ;

✓ Encode and the N_c new individuals, compute their fitness, and insert them into the new population ;

✓ Receive N_m migrant elites (with their fitness) from the previous population of each N_s– 1 other processors, to complete the new population ;

✓ Send the best N_m= (N_i–N_e–N_c)/(N_s– 1) individuals (with their fitness) to the new populations of N_s–1 other processors ;

✓ Sort the N_p individuals (codewords) in descending order of their fitness ;

✓ Replace the current population with the new one ;*gen*←*gen* + 1 ;

End While

•**Step 3**:*Decision*

✓ Get the best individuals $(D'^{(i)})_{1\le i\le Ns}$ of the last populations of all processors. The best one *D'* of them is the closest to *r'*.i.e. *D'* = arg min { $\| D'^{(i)} - r' \|$, $1 \le i \le N_s$}. The decided codeword is then $D = \pi^{(-1)}(D')$.

The flowcharts of the previous algorithm are illustrated in both figure 6 and figure 7.

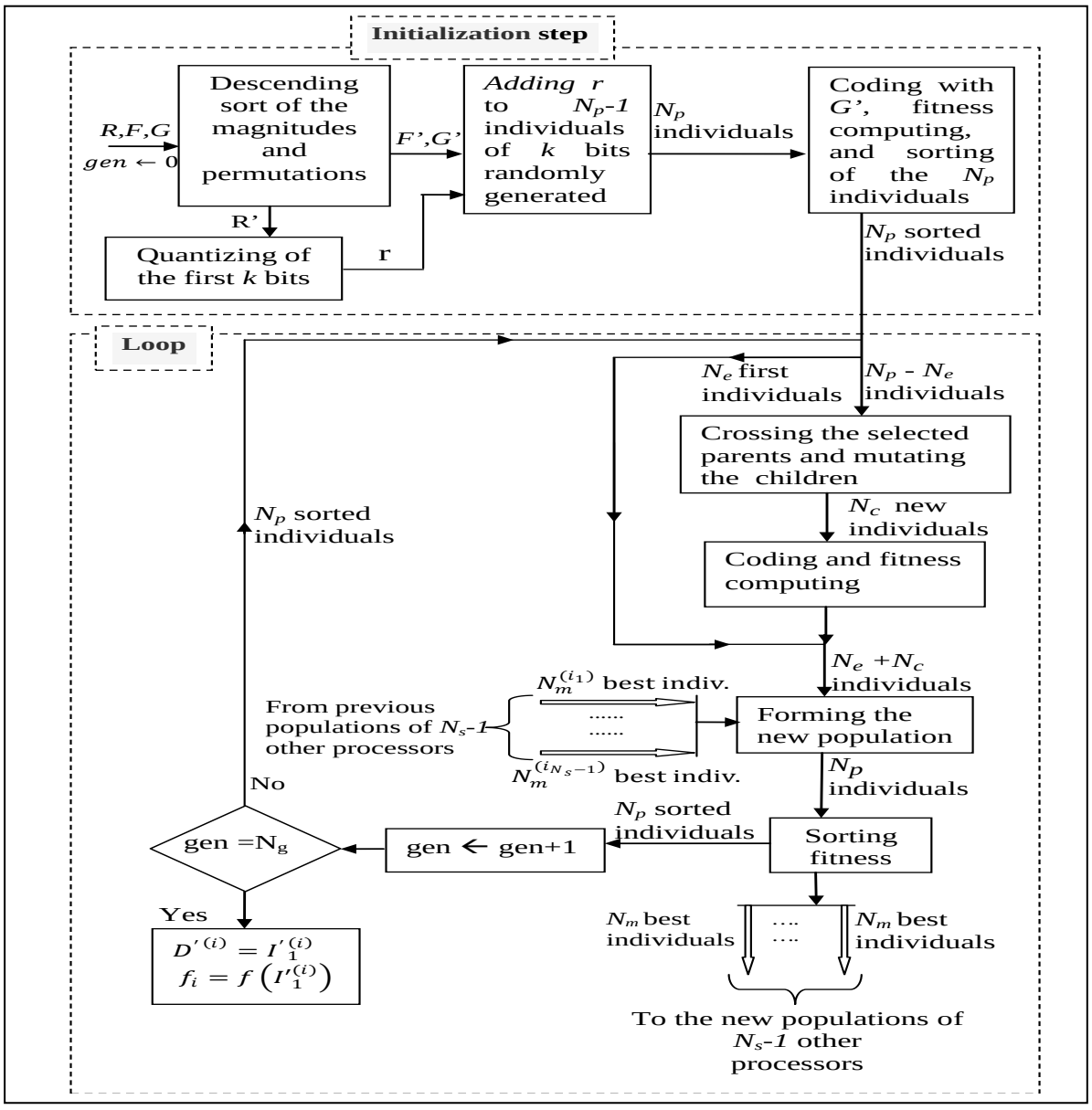


Fig. 6.The flowchart of the proposed PGAD sub-algorithm running on the *i*th processor.

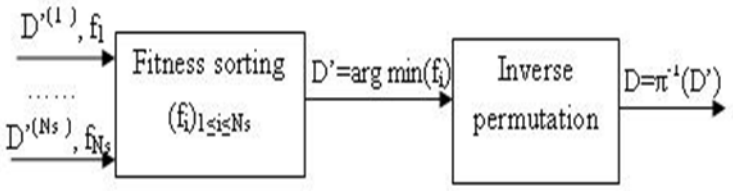


Fig. 7.The codeword decision flowchart of the proposed PGAD algorithm.

4. PARALLELIZATION SCHEMAS OF ITERATIVE 2D-PBC DECODING

The purpose is not only to reduce the execution time and occupied memory space, but also improve the quality of error correction. So, the parallelization schemas can affect the entire iterative decoder or just its elementary decoders.

Their corresponding algorithms can be run on parallel machines containing enough processors (multi-processors, multicomputer)or a network of computers. Communications between processors are performed using global variables stored in a common memory, or via send/receive primitives. To simplify the graphs, we have considered just four processors.

4.1 First schema

The figure 8 depicts the first parallel iterative decoder that we propose. It is simply runs in parallel, a conventional 2Diterative decoder on each one of the N_s processors of a parallel machine or a distributed system, a given number of iterations N_{it}. At the last iteration, the ith processor decides the codeword $D^{(i)}$. The final codeword to decide is the one with the best fitness of all $D^{(i)}$, where $1 \leq i \leq N_s$.

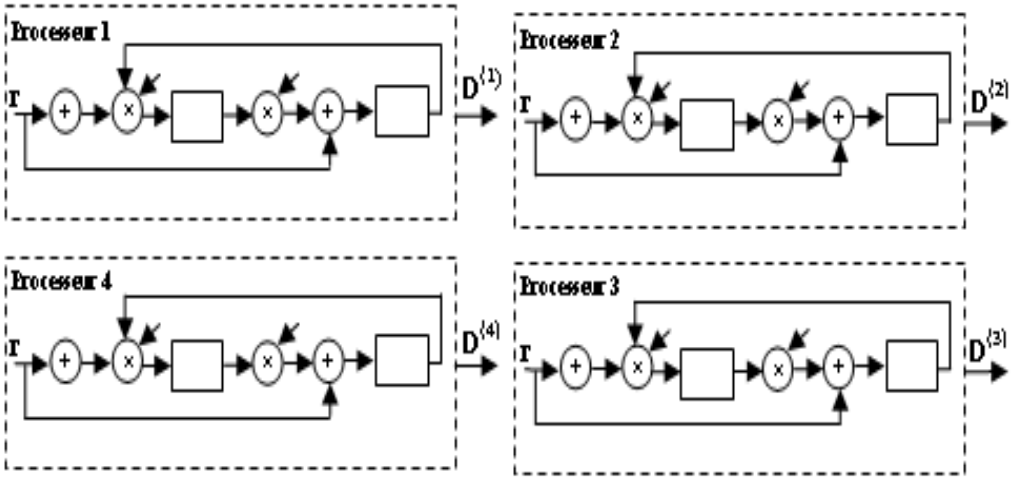


Figure 8.The first iterative parallel decoding schema on 4 processors.

The corresponding algorithm is given below:

Algorithm: PARAL_ITER1

For i=1**to**N_s**do**

Run *IGAD on* the $i$thprocessor *;*

Get the decided codeword$D^{(i)}$*;*

End For

$D=arg\ max\ \{fitness(D^{(i)}),\ 1 \le i \le N_s\}$

4.2 Second schema

As shown in figure 9, the second schema consists in executing, Nit times, a parallel decoder which is composed of N_sIGADs decoders. Each one is running on a dedicated processor. The processors are connected by a network. At each iteration, all decoders run in parallel according to their inputs. At the first iteration, the extrinsic information at the input of all decoders is zero. At tth iteration, the decoder of ith processor provides as output, extrinsic information which will be injected, in combination with the received word r, to the decoder input of processor s+1, at the iteration t+1. The extrinsic information given by the decoder of the last processor is fed to the decoder input of the first one.

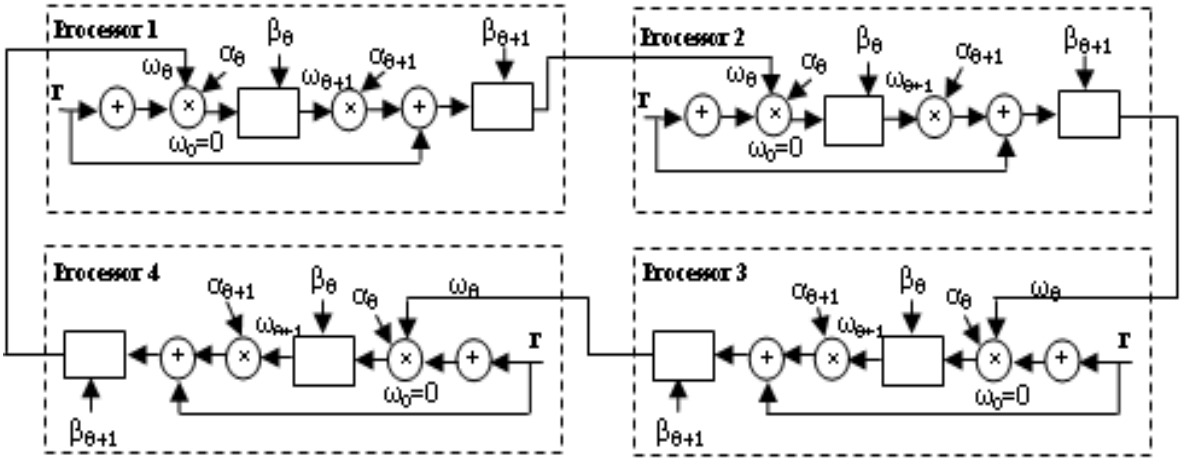


Figure 9. Second Parallel iterative decoding schema on 4 processors

The algorithm of this schema is:

Algorithm: PARAL_ITER2

θ=0 *;* ω_0=0 *;*

While $\theta \le 2(N_{it}-1)$**do**

For i=1**to**N_s**do**

Run$IGAD_1^{(\theta/2)}$ *;*

Get the decided codeword on the$i$th processor$D_i^{(\theta/2)}$*;*

Compute$\omega_i^{(\theta+2)}$based on$D_i^{(\theta/2)}$*;*

End For

Temp=$\omega_{Ns}^{(\theta+2)}$ *;*

For $i = N_s$ **to** 2 **do**

$\omega_i^{(\theta+2)} = \omega_{i-1}^{(\theta+2)}$;

End For

$\omega_1^{(\theta+2)} = temp$;

$\theta = \theta + 2$;

End while

$D = D_{Ns}^{(Nit-1)}$;

4.3 Third schema

The elementary decoder in the conventional iterative decoder which running on a single processor is replaced by the parallel decoder which will be run on N_s interconnected processors. At each iteration, these processors run their basis decoders independently, and each one decides its own code word. The extrinsic information provided by the parallel decoder at this iteration, is computed based on the best decided codeword.

The first decoder accepts an input that depends on the received word r and the extrinsic information $\omega^{(\theta)}$, and provides another information $\omega^{(\theta+1)}$ which will form with r , the input of the second parallel decoder. Similarly, the second decoder makes a processing based on its input, to decide the codeword. It also provides a new extrinsic information $\omega^{(\theta+2)}$ to be injected in combination with r to the input of the first decoder. This processing is performed N_{it} times, before getting the code word decided by the second parallel decoder (figure 10).

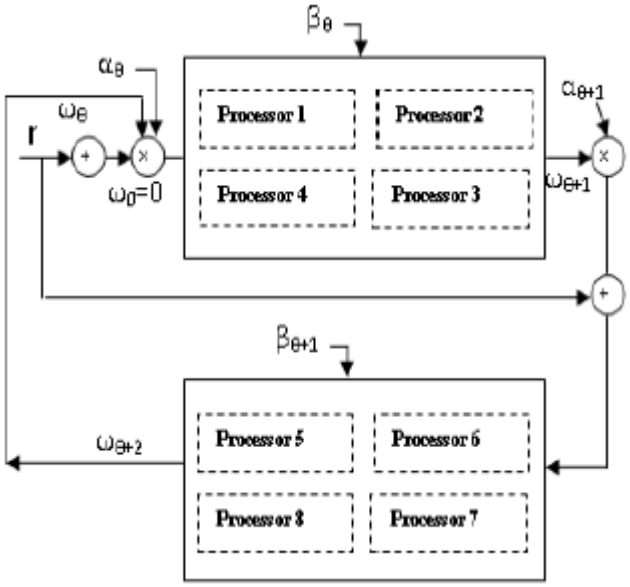


Fig. 10. Third Parallel iterative decoding schema on 4 processors

The following algorithm describes well the conduct of this parallelization scheme of iterative decoding:

Algorithm : PARAL_ ITER3

θ=0 ; ω_0=0 ;
While $\theta \leq 2(N_{it}-1)$ **do**
  For i=1 **to** N_s **do**
    Run $IGAD_1^{(\theta/2+1)}$;
    Get the decided codeword on the ith processor $D_i^{(\theta/2+1)}$;
  End For
  Get $\omega^{(\theta+1)}$ using $D_1^{(\theta/2+1)} = arg\ max\{fitness(D_i^{(\theta/2+1)}, 1 \leq i \leq N_s\}$;
  For $i = N_s+1$ **to** $2N_s$ **do**
    Run $GAD_i^{(\theta/2+1)}$;
    Get $D_i^{(\theta/2+1)}$;
  End For
  Get $\omega^{(\theta+2)}$ using $D_2^{(\theta/2+1)} = arg\ max\ \{fitness\ (D_i^{(\theta/2+1)}),\}$;
  $\theta = \theta+2$; $N_s+1 \leq i \leq 2N_s$
End while
$D = arg\ max\ \{fitness\ (D_i^{(Nit)}),\ N_s+1 \leq i \leq 2N_s\}$

4.4 Fourth schema

The principle of the fourth scheme is the same as the previous one, by replacing the two elementary parallel decoders based on GAD by two elementary parallel decoders PGAD1 and PGAD2. As shown in figure 5, the processors do not runindependent decoders but decoders working together.

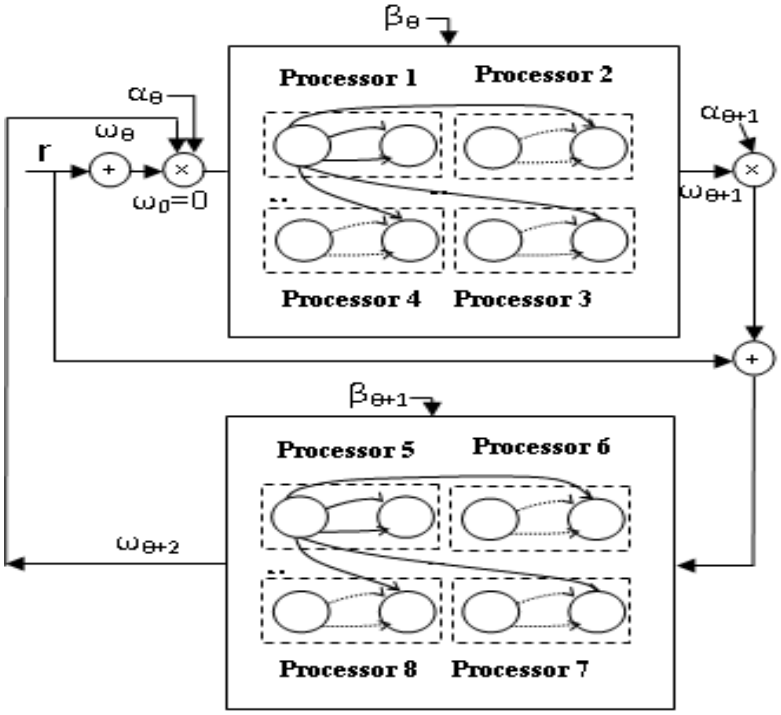


Fig. 11. Fourth Parallel iterative decoding schema on 4 processors

Its algorithm is presented below:

Algorithm: PARAL_ITER4

θ=0 ; ω_0=0 ;
While $\theta \leq 2(N_{it}-1)$ **do**
Run$PGAD_1$;
Get$D^{(\theta+1)}$;
Compute $\omega^{(\theta+1)}$ using $D^{(\theta+1)}$;
Run$PGAD_2$;
Get$D^{(\theta+2)}$;
Compute $\omega^{(\theta+2)}$ using $D^{(\theta+2)}$;
$\theta=\theta+2$;
End while
$D=D^{(Nit)}$;

5. TIME COMPLEXITY OF THE PROPOSED SCHEMES

In this section, we interest to the expression of the time complexity of each proposed algorithm.

The complexity of GAD is [9] :

$$O[k^2n + NpNg(kn + \ln(Np))] \tag{10}$$

For the IGAD decoder, its complexity is [11] :

$$O\left(N_{it}\left[k^2 g\left(k^1, n^1, N_p, N_g\right) + n^1 g\left(k^2, n^2, N_p, N_g\right)\right]\right), \tag{11}$$

where

$$g(k, n, N_p, N_g) = k^2n + N_pN_g(kn + \ln(N_p)) + N_pn + n^2 \tag{12}$$

The complexity of PGAD is derived in [12] as :

$$O[nln(n) + k^2n + kn\left(N_p + N_gN_c\right) + N_pN_g \ln\left(N_p\right) + N_p \ln\left(N_p\right)] \tag{13}$$

5.1 First iterative Parallel decoding complexity

The algorithm PARAL_ITER1 contains a loop of instructions that run in parallel on Ns processors. The last instruction which sorts, in descending order, code words decided by all processors, has a complexity of $N_s\ln(N_s)$. Thus the complexity of PARAL _ITER1 is Complexity(IGAD) + $N_s\ln(N_s)$. i.e. :

$$O(N_{it}[k_2 g(k_1, n_1, N_p, N_g) + n_1 g(k_2, n_2, N_p, N_g)] + N_s \ln(N_s)) \quad (14)$$

Note that the complexity of this new iterative decoder is increased by $N_s\ln(N_s)$ relative to that of the conventional one (IGAD). However, it increases the probability of having decided the correct code word, $N_s$times.

5.2 Second Iterative Parallel decoding complexity

The block instructions of the first loop of the algorithm PARAL_ITER2 are run in parallel on each one of N_s processors. Therefore their complexity is the same as that of IGAD. The complexity of the loop in the second algorithm is $N_{it}N_s$. So, the total complexity of this algorithm is :

$$O(N_{it}[N_s + k_2 g(k_1, n_1, N_p, N_g) + n_1 g(k_2, n_2, N_p, N_g)]) \quad (15)$$

However, the turbo effect with N_{it} iterations in IGAD is equivalent to that of PARAL_ITER2 with N_{it}/N_s iterations. Furthermore, this scheme has the advantage of being able to cascade different types of decoders, where some of them can compensate inefficient other at each iteration.

5.3 Third Iterative Parallel decoding complexity

For this scheme, the extrinsic information computed by the two elementary decoders, depends on the best codeword decided by their N_s processors. The complexity of descending sort of these decided words for N_{it} iterations is $N_{it}N_s\ln(N_s)$. The last instruction has a complexity of $N_s\ln(N_s)$. Thus its complexity is increased by $N_{it}N_s\ln(N_s)$ compared to IGAD. i.e. :

$$O(N_{it}[k_2 g(k_1, n_1, N_p, N_g) + n_1 g(k_2, n_2, N_p, N_g) + N_s ln(N_s)]) \quad (16)$$

5.4 Fourth Iterative Parallel decoding complexity

It is clear that the complexity of PARAL_ITER4 is $N_{it}\times$ complexity(PGAD). i.e. :

$$O[nln(n) + k^2 n + kn\left(N_p + N_g N_c\right) + N_p N_g \ln(N_i) + N_p \ln\left(N_p\right)] \quad (17)$$

The complexity is multiplied by the iteration number, but the turbo effect is significantly augmented. So the correction capacity is improved.

6. CONCLUSION

We have presented four iterative parallel decoders for two dimensional product block codes. They can run on a parallel machine with enough processors or on a network of computers. One of their advantages is that they allow combining elementary decoders with the

same or different genetic parameters. This allows benefiting from the highlights of each code, at each iteration. We have shown that their complexities were slightly increased, but improve the decoding performances.

We intend to implement these schemes and test them on certain codes like BCH codes to validate the theoretical results with simulations.

The proposed schemes can also be applied to three dimensional product block codes. Also, for the first three schemes, their elementary decoders can be any ones, not necessarily based on GAs.

REFERENCES

[1]. C. E. Shannon, ”A mathematical theory of communication,” Bell System Technical Journal, vol. 27, pp. 379-423 and 623-656, July and October, 1948.

[2]. Berrou, G., A. Glavieux, and P. Thitimajshima. ”Near Shannon limit error-correcting coding : Turbo Codes,” in proc. 1993 Int. Conf. Commun. Geneva,Switzerland, May 1993, pp. 1064-1070.

[3]. Gallager, R.G. Low Density parity Check Codes. Th`ese. Cambridge, Mass.1963. IRE Transactions on Information Theory.1962.

[4]. D.J.C Mackay, and R.M. Neal, ”Near Shannon Limit Performance of Low Density Parity Check Codes”, Electronics Letters, Vol. 32, no. 18, pp.1645-1646, August 1996.

[5]. J. H. Holland, ”Adaptation In Natural And Artificial Systems, University of Michigan Press”, 1975.

[6]. D.E. Goldberg, ”Genetic Algorithms in Search, Optimization, and Machine Learning”, Addison-Wesley, 1989.

[7]. A.TANENBAUM, ”Computer architecture”. Dunod 2001.

[8]. H. S. Maini, K. G. Mehrotra,C. Mohan, S. Ranka, ”Genetic Algorithms for Soft Decision Decoding of Linear Block Codes”, Journal of EvolutionaryComputation, Vol.2, No.2, pp.145-164, Nov.1994.

[9]. F. El Bouanani, H. Berbia, M. Belkasmi, H. Ben-azza, ”Comparaison des d´ecodeurs de Chase, l’OSD et ceux basés sur les algorithmes génétiques”,GRETSI 2007, Troyes, France 11-14 Septembre 2007 in french.

[10]. M. Belkasmi, H. Berbia, F. El Bouanani, ”Iterative decoding of product block codes based on the genetic algorithms”, Source and Channel Coding(SCC), 2008 7th International ITG Conference on Source and Channel Coding (SCC’08), 2008.

[11]. A. Ahmadi, F. El Bouanani, H. Ben-Azza, and Y. Benghabrit, ”Reduced Complexity Iterative Decoding of 3D-Product Block Codes Based on GeneticAlgorithms”, Journal of Electrical and Computer Engineering, Volume 2012.

[12]. A. Ahmadi, F. El Bouanani, H. Ben-Azza, and Y. Benghabrit, ”A Novel Decoder Based on Parallel Genetic Algorithms for Linear Block Codes”,International Journal of Communications, Network and System Sciences, 2013, 6, 66-76.

[13]. F.J. Macwilliams and N.J.A. Slone, ”The Theory of error correcting rrror”, North-Holland publishing comapny, 1978, pp. 567-580.

[14]. E.R. Berlkamp, R. J. McEliece, and H. C. A. Van Tilborg, ”On the inherent intractability of certain coding problems”, IEEE Transactions on InformationTheory, IT-24 :384-386, 1978.

[15]. A. Vardy, ”The intractability of computing the minimum distance of a code”, IEEE Transactions on Information Theory, IT-43 :1757-1766, 1997.

[16]. A. E. Eiben, E. H. L. Aarts, and K. M. Van Hee, ”Global convergence of genetic algorithms: A markov chain analysis”, in Parallel Problem Solvingfrom Nature H. -P. Schwefel and R. M¨anner (Eds), Berlin and Heidelberg : Springer, 1991, pp. 4-12.

[17]. D.B. Fogel, ”Evolving Arti cial Intelligence”, PhD dissert., San Diego University of California, 1992.

[18]. G. Rudolph, ”Convergence analysis of canonical genetic algorithms”, IEEE Trans. on Neural Networks, 5(1): 96-101, 1994.

TNC Transactions on Networks and Communications
SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
VOLUME 2, ISSUE 3
ISSN: 2054 - 7420

'Green Wall Rating': A Methodology to Evaluate Sustainable Development by Implementing Green Wall Model

Ankit Kumar Srivastava, Neeraj Kumar Tiwari
Department of Computer Science and Engineering, Shri Ramswaroop Memorial University, Lucknow-Deva Road, Hadauri, Tindola, Uttar Pradesh-225003, India
Assistant Professor Under faculty of Computer Science and Engineering, Shri RamSwaroop Memorial University, Lucknow-Deva Road, Hadauri, Tindola,Uttar Pradesh-225003, India
Ankit.suraj786@gmail.com, neeraj.cs@srmu.ac.in

ABSTRACT

The modern technology has changed the life of human being. Rapidly changing technology develops many kind of development approaches and products which makes the life easier and comfortable but they have also a negative impact on human beings as well as on the environment, In this paper a green wall rating methodology is being introduced. The protocols used in this rating methodology, a developer can rate any product or process well in terms of sustainability. Further green wall rating (GWR) can be measured with the help of sustainable development qualities. About the essential qualities for sustainable development has been discussed in our earlier publication (Srivastava, et. al., 2014). On the basis of conceptual hypothesis the practical value of GWR has to be set between 2.00 to 4.00 units. For being a sustainable process or product it must contains at least 2.00 green wall rating.

Keywords: Sustainable development, GHG, EPA (environmental protection agency).

1. INTRODUCTION

In our recent study we have proposed a Green Wall Model, a model for sustainable development. By this model we can address both the qualities of sustainable software and profiling energy consumption. In that model we have introduced four basic qualities of sustainable development: Entourage effect, curtailment, social impact, performance [1]. If any development process contains these qualities then we can say that it is a sustainable development and now with the help of that model we are introducing a new technique and terminology which is called as green wall rating. It is a grading methodology or rating technique of any development process and this methodology also enlighten about the process compatibility regarding with the environment. This methodology comes under the area of green computing and it ensures about sustainable development. The reasons behind this study

DOI: 10.14738/tnc.23.254
Publication Date: 12th June 2014
URL: http://dx.doi.org/10.14738/tnc. 23.254

are firstly the world of supercomputing and fast growing technology there are rapidly changing of product and various demands call for manufacturing products so for that various methods and approaches are adapted and developed [2] so there is no technique has present that ensures the new adapted technology is compatible to human being and also for environment. Secondly if any one considered the environment, only discuss about greenhouse gas emissions or wastages no one relates to the effect on human being e.g. IT industry is responsible for 2–2.5% of global greenhouse (GHS) gas emissions [3]. This paper contributes about the rating system for various life cycle models or development process by which we can decide that how much it is sustainable.

2. BACKGROUND

Until now, there are many publications available discussing about the sustainable development and energy saving methodology. Shaofei jiang [4] has describes the innovative design of a new laptop by which we can overcome the energy related problems of traditional laptops. Anh Hoang describes the life cycle of a laptop from materials acquisition to manufacturing, use, and end-of-life disposition in terms of contribution to greenhouse gas (GHG) emission [5]. EPA (Environmental Protection Agency) described the life cycle of a cell phone [6]. In this study the requirements for mobile phone manufacturing and as well as various stages of mobile phones have been discussed. Lewis produces its artifacts for the environmentally sustainable infrastructure design [7]. It is a comprehensive understanding of environmental sustainability that needs for IT infrastructure system design.

3. MATERIALS AND METHODS

This paper is based on the qualities of sustainable development that we are discussed in our previous study [1]. According to that qualities of sustainable development can be divided into four major qualities those are: Entourage effect, Curtailment, Social impact and Performance and by the help of these qualities we can calculate the Green wall rating of any development process or any product. Green wall rating is a terminology used for explaining about any product that how much that product is effective regarding with the human being or environment. We can calculate it by following way as mentioned in Table 1.

Table 1: Sustainable Development Qualities

S. No.	Major Qualities	Sub Qualities
1	Entourage effect	Bionomics impact, Pernicious effect, Low vitiation, Extravagance, Use renewable energy.
2	Curtailment	Low elementary cost, Less raw material cost, Low ontogeny, Low breakdown cost.
3	Social impact	Inclinated, Use provincial material, Less use of energy, Safe and secure, Externalize energy.

4	Performance	Ease to build, Long life, Use green energy sources, Less chance to decay, Time saving, Money saving, High performance

Each major quality has value=1 means,

- Value of Entourage effect=1.
- Value of Curtailment =1.
- Value Social impact =1.
- Value of Performance =1.

So from the above values we can calculate the values of sub qualities so,

3.1 For Entourage effect

Total number of sub qualities=5 and value of entourage effect=1, thus

Value of each sub quality = 1/5

=.20

So that we can say that,

- value of Bionomics impact=.20
- Value of Pernicious effect=.20
- Value of Low vitiation=.20
- Value of Extravagance=.20
- Value of Use renewable energy=.20

3.2 3.2 For Curtailment

Total number of sub qualities=4 and value of Curtailment =1, thus

Value of each sub quality = 1/4

=.25

So that we can say that,

- Value of Low elementary cost=.25
- Value of Less raw material cost=.25
- Value of Low ontogeny=.25
- Value of Low breakdown cost=.25

3.3 For Social Impact

Total number of sub qualities=5 and value of Social impact =1, thus

Value of each sub quality = 1/5.

=.20

So that we can say that,

- value of Inclinated =.20
- Value of Use provincial material =.20
- Value of Less use of energy =.20
- Value of Safe and secure =.20
- Value of Externalize energy=.20

3.4 For Performance

Total number of sub qualities=7 and value of Performance=1, thus

Value of each sub quality=1/7

=.14

So that we can say that,

- Value of Ease to build=.14
- Value of Long life=.14
- Value of Use green energy sources=.14
- Value of Less chance to decay=.14
- Value of Time saving=.14
- Value of Money saving=.14
- Value of High performance=.14

If any development process contains all the sustainable qualities then we can say that development will be the sustainable development [1]. So the maximum value of green wall rating will be = 4 and the minimum value of green wall rating = .14, if it contains only one quality.

So the green wall rating that is necessary for the sustainable development = mean of (maximum value of green wall rating, minimum value of green wall rating)

Minimum green wall rating = (4+.14)/2

=2.07

Thus we can say that if any development process has green wall rating = 2 or more than, then that will be called as sustainable development.

Green wall rating evaluation or for organizational implementation

We can calculate the green wall rating of the after construction of the product, if product's development process satisfy any one of the qualities then we will take the value of that quality for e.g. if any development process has low elementary cost then we will consider its value=.20 such as the other are also considered so,

Green wall rating = addition of total values of qualities that are containing by development process.

Representation of green wall rating

The representation of green wall rating in product is can be done by GWR= (the value of green wall rating). Means if value of green wall rating for any product=3.00, then we can show its green wall rating by GWR=3.00

4. CONCLUSION

The initial studies presented in this paper have shown the qualities of sustainable development and these qualities are Entourage effect, Curtailment, Social impact and performance. By visualizing these qualities we evaluated green wall rating (GWR). The calculation of GWR will depends on the number of qualities contained by development process. The maximum value of GWR will be 4.00 units if development process satisfies all the qualities of sustainable development and minimum value of green wall rating for sustainable development is 2.00. If development process has 2.00 or more than 2.00 GWR then we can say that is a sustainable development process. The sustainability of the product depends on green wall rating (GWR). The higher value of GWR (between 3.00 to 4.00) means the product or process is more sustainable as compare to those products which contains the less green wall rating (2.00 to 3.00).

REFERENCES

[1]. Srivastava, A. and N. K. Tiwari, Green *Wall: A Methodology for Sustainable Development using Green Computing*, International Journal of Scientific and Innovative Research, 2014. 2(1).

[2]. Lindskoga, E., Berglunda, J., Vallhagenb, J. and B. Johanssona, *Visualization support for virtual redesign of manufacturing systems*, 2013. 7: p. 419-424.

[3]. Kharchenko, V. and A. Gorbenko, *Green Computing and Communications in Critical Application Domains: Challenges and Solutions*, 2013. p. 191-197.

[4]. Jiang, S., Lianga, W., Lia, J. and Congda. Lua, *Innovative Design of a New Laptop*, International Workshop on Information and Electronics Engineering (IWIEE), 2012. pp. 2932-2937.

[5]. Hoang, A., Tseng, W., Viswanathan, S. and H. Evans, *Life Cycle Assessment of a Laptop Computer and its Contribution to Greenhouse Gas Emissions*. National university, (San Diego), 2009. pp. 130.

[6]. The life cycle of a cell phone by EPA (environmental protection agency), 2004.

[7]. Curtis L*., Environmentally Sustainable Infrastructure Design*, the Architecture Journal #18, 2009.