

PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
UNIVERSITY OF MOHAMED BOUDIAF - M'SILA

FACULTY: Mathematics and Computer Science

DEPARTMENT: Computer Science

N°: 49



DOMAIN: Mathematics and Computer Science

BRANCH: Computer Science

OPTION: Network

**A Dissertation in Fulfillment
For the Requirements of the Degree of Master**

By: LATELLI OUSSAMA

SUBJECT

**Router nodes placement problem in Green Wireless
Mesh Networks(WMNs)**

Defended publicly on: 11/06/2017

Board of Examiners:

MR: Abbache Farid

University of M'sila

Chairman

DR: Lamri Sayad

University of M'sila

Supervisor

MR: Kamel Mohamed

University of M'sila

Examiner

Academic Year: 2016/2017

Table of Contents

Table of Contents.....	i
Figure list.....	iv
Table list.....	v

GENERAL INTRODUCTION

1. Context.....	- 2 -
2. Statement of the Problem.....	- 2 -
3. General Objectives.....	- 3 -
4. Methodology.....	- 3 -
5. Report Outline.....	- 3 -

CHAPTER 01

WIRELESS MESH NETWORK

1. Introduction.....	- 5 -
2. Network architecture.....	- 5 -
2.1 Infrastructure/Backbone WMNs.....	- 6 -
2.2 Client WMNs.....	- 7 -
2.3 Hybrid WMNs.....	- 8 -
3. Characteristics.....	- 9 -
3.1 Multi-hop wireless network.....	- 9 -
3.2 Support for ad hoc networking, and capability of self-forming, self-healing, and self-organization.....	- 9 -
3.3 Mobility dependence on the type of mesh nodes.....	- 9 -
3.4 Multiple types of network access.....	- 9 -
3.5 Dependence of power-consumption constraints on the type of mesh nodes... ..	- 9 -
3.6 Compatibility and interoperability with existing wireless networks.....	- 10 -
3.7 Integration.....	- 10 -

3.8	Dedicated routing and configuration	- 10 -
3.9	Multiple radios.....	- 10 -
3.10	Mobility	- 11 -
4.	Key Benefits of a Wireless Mesh Network	- 11 -
4.1	Less Expensive than Traditional Networks.....	- 11 -
4.2	Wireless Mesh is extremely adaptable and expandable	- 11 -
4.3	Wireless Mesh Networks Support High Demand	- 11 -
5.	WMN deployments and applications.....	- 11 -
5.1	Connectivity services.....	- 11 -
5.2	Use cases scenarios.....	- 12 -
5.3	Use cases examples	- 12 -
6.	Conclusion	- 15 -

CHAPTER 02

ROUTER PLACEMENT PROBLEM IN WIRELESS MESH NETWORKS

1.	Introduction.....	- 17 -
2.	Routers placement problems in WMN	- 17 -
2.1	WMN-dynRNP	- 17 -
2.2	WMN-RNP.....	- 17 -
3.	Problem Description	- 18 -
4.	A PSO approach to the Routers placement problems in WMN	- 21 -
4.1	Solution exemplification	- 21 -
4.2	Fitness function	- 22 -
4.3	Updating position	- 23 -
4.4	PSO Algorithm	- 24 -
5.	Rechargeable router system	- 25 -
5.1	Energy flow model	- 25 -

5.2	Energy charging model.....	- 25 -
5.3	Energy discharging model	- 25 -
6.	Conclusion	- 26 -

CHAPTER 03

DESIGN, IMPLEMENTATION AND RESULTS

1.	Introduction.....	- 28 -
2.	Embarcadero C++ Builder	- 28 -
3.	Conception and implementation	- 29 -
3.1	Conception of the PSO algorithm.....	- 30 -
3.2	conception of the energy flow model algorithm.....	- 36 -
4.	Simulation.....	- 39 -
4.1	Simulation parameters	- 39 -
5.	Results.....	- 40 -
5.1	Case 01 (changing clients number)	- 40 -
5.2	Case 02 (changing routers number).....	- 43 -
5.3	Case 03 (changing PSO iteration number)	- 46 -
5.4	Case 04 (changing the client's positions).....	- 49 -
6.	Conclusion	- 52 -

GENERAL CONCLUSION

1.	GENERAL CONCLUSION	- 54 -
----	--------------------------	--------

Figure List

Figure 1.1: mesh routers	6
Figure 1.2: mesh clients	6
Figure 1.3: Infrastructure/backbone WMNs.....	7
Figure 1.4: Client WMNs.....	8
Figure 1.5: Hybrid WMNs.....	8
Figure 1.6: Transportation system.....	12
Figure 1.7: Broadband Home Networking.....	13
Figure 1.8: Enterprise network.....	13
Figure 1.9: WMNs are replacing DSL based solution.....	14
Figure 1.10: Military networks.....	14
Figure 1.11: Emergency response	15
Figure 2.1: 2 placements for the same WNN with 5 mesh routers and 14 mesh clients.....	20
Figure 2.1: the new topology	20
Figure 2.3: clarification of updating the location of particle k.....	23
Figure 3.1: Embarcadero C++ Builder.....	29
Figure 3.2: Implementation description.	30
Figure 3.3: Implementation interface (RAD studio XE3)	40
Figure 3.4: number of routers variations by contribution of number of clients.....	41
Figure 3.5: fitness variations by contribution of number of clients.....	42
Figure 3.6: connectivity variations by contribution of number of clients.....	42
Figure 3.7: coverage variations by contribution of number of clients.....	43
Figure 3.8: number of routers variations by contribution of number of routers.....	44
Figure 3.9: fitness variations by contribution of number of routers	44
Figure 3.10: connectivity variations by contribution of number of routers.....	45
Figure 3.11: coverage variations by contribution of number of routers.....	46
Figure 3.12: number of routers variations by contribution of PSO iteration number	46
Figure 3.13: fitness variations by contribution of PSO iteration number.....	47
Figure 3.14: connectivity variations by contribution of PSO iteration number.....	48

Figure 3.15: coverage variations by contribution of PSO iteration number.....	48
Figure 3.16: Topology 01	49
Figure 3.17: Topology 02.....	49
Figure 3.18: Topology 03.....	49
Figure 3.19: Topology 04.....	49
Figure 3.20: Topology 05.....	50
Figure 3.21: Topology 06.....	50
Figure 3.22: The average of routers number variation in six topologies.....	50
Figure 3.23: The average of fitness variation in six topologies.....	51
Figure 3.24: The average of connectivity variation in six topologies	51
Figure 3.25: The average of coverage variation in six topologies.....	52

Table List

Table 3.1: Simulation parameters	39
Table 3.2: PSO parameters	39
Table 3.3: Energy Flow Model parameters	39

GENERAL INTRODUCTION

GENERAL INTRODUCTION

1. Context

Over the past few years, wireless communications technologies have become an important part of our daily lives. Thanks to their flexibility, wireless networks have seen real growth and unprecedented popularity among the public. Thus, they motivated the researchers towards the deployment of new wireless solutions, one of the most important of these technologies is the appearance of Wireless Mesh Networks (WMN), which consist in interconnecting a plurality of wireless devices together forming a mesh without recourse to wiring.

Lately, wireless mesh networks have been a concentration of interest for a growing number of analysts and researchers. Despite this work, several problems remain to be solved. The latter revolve around adaptive routing, autonomy mechanisms, self-organization, self-configuration, network design, and so on.

The wireless mesh network is an infrastructure radio network based on distributed collaboration between the access points. The communications between two access points are supported by several intermediate nodes (multi-jumps) whose role is to relay the information from one point to another. Thus, some access points have a specific functionality: they are connected to a transport network providing access to the Internet. The clients are connected via a wireless network to the access points, which are likewise connected by wireless links. Clients can be laptops, desktops, PDAs, IP phones, RFID readers, etc.

2. Statement of the Problem

During these last few years, the problem of router node placement (RNP) in WMNs has been studied so many times. Such a combinatorial optimization problem considers a WMN composed of mesh clients and mesh routers in which mesh routers serve as the access point towards mesh clients and connect to other mesh routers through point-to-point wireless links.

So, in this work we chose the PSO algorithm (Particle Swarm Optimization), we implemented this algorithm and we adopted it to solve the problem of router nodes placement in WMNs, and in our project all the mesh routers had a solar panel for charging the energy, for this purpose we applied an Energy Flow Model, that system model contains two models the first is energy charging

model and the second is energy discharging model, we implemented that and we concluded with a solution for the problem of router node placement (RNP) in WMNs.

3. General Objectives

The subject of this work is to develop an algorithm for solving the problem of router node placement (RNP) in green WMNs. And create a new algorithm from two strong techniques in this domain (PSO algorithm and Energy Flow Model).

4. Methodology

Our work is divided in three phases:

- Problem modulization.
- Implementation of the problem modulization, and the algorithms.
- Study of the performance of the algorithm.

5. Report Outline

This report is organized as follows:

- Chapter 01: Devoted to the presentation of wireless mesh networks.
- Chapter 02: Presents the problem of deploying routers in a wireless mesh network, and prescribes the two algorithms that we combine to solve that problem
- Chapter 03: Presents the conception and the implementation of the proposed algorithms and discusses the results obtained.

**WIRELESS MESH
NETWORKS**

**CHAPTER
01**

CHAPTER 01

WIRELESS MESH NETWORKS

1. Introduction

In this chapter, we will talk about the Wireless Mesh Network (WMN), what is WMNs, the Network architecture of WMNs (mesh routers and mesh clients), the classification of WMNs (Infrastructure/Backbone WMNs, Client WMNs and Hybrid WMNs) and finally we will describe the Characteristics of WMNs and their application areas. A WMN consists of a set of mesh routers, mesh clients and mesh gateways. Forwarding packets on behalf of other nodes that may not be within direct wireless transmission range of their destinations[17]. A WMN is dynamically self-organized and self-configured, with the nodes in the network automatically establishing and maintaining mesh connectivity among themselves (creating, in effect, an ad hoc network). This feature brings many advantages to WMNs such as low up-front cost, easy network maintenance, robustness, and reliable service coverage[15].

2. Network architecture

Wireless mesh networks consist of 3 types of nodes: mesh routers, mesh clients and mesh gateways. Other than the routing capability for gateway/repeater functions as in a conventional wireless router, a wireless mesh router contains additional routing functions to funding mesh networking. To further expand the flexibility of mesh networking, a mesh router is typically equipped with multiple wireless interfaces built on either the same or different wireless access technologies. Compared with a conventional wireless router, a wireless mesh router can achieve the same coverage with much lower transmission power through multi-hop communications. Optionally, the medium access control (MAC) protocol in a mesh router is boosted with better scalability in a multi-hop mesh environment. In spite of all these differences, mesh and conventional wireless routers are usually built based on a similar hardware platform. Mesh routers can be built based on dedicated computer systems (e.g., embedded systems) and look compact, as shown in (Figure1.1). They can also be built based on general-purpose computer systems (for example, laptop/desktop PC)[15].

Mesh clients also have essential functions for mesh networking, and thus, can also work as a router. However, gateway or bridge functions do not exist in these nodes. In addition, mesh clients frequently have only one wireless interface. Therefore, the hardware platform and the software for mesh clients can be much simpler than those for mesh routers. Mesh clients have a higher variety of devices compared to mesh routers. They can be a laptop/desktop PC, pocket PC, PDA, IP phone, RFID reader, BACnet (building automation and control networks) controller, and many other devices, as shown in (Figure1.2)[15].



Figure 1.1 mesh routers [15]



Figure 1.2 mesh clients [15]

The architecture of WMNs can be classified into three main groups based on the functionality of the nodes:

2.1 Infrastructure/Backbone WMNs

The first architecture is shown in Figure1.3, where dash and solid lines indicate wireless and wired links, respectively. This type of WMNs includes mesh routers forming an infrastructure for clients that connect to them. The WMN infrastructure/ backbone can be built using various types of radio technologies, in addition to the mostly used IEEE 802.11 technologies. The mesh routers form a mesh of self-configuring, self-healing links among themselves. With gateway functionality, mesh routers can be connected to the Internet. This approach, also referred to as infrastructure meshing, provides backbone for conventional clients and enables integration of WMNs with existing wireless networks, through gateway/bridge functionalities in mesh routers. Conventional clients with Ethernet interface can be linked to mesh routers via Ethernet links. For conventional clients with the same radio technologies as mesh routers, they can straight communicate with mesh

routers. If different radio technologies are used, clients must communicate with the base stations that have Ethernet connections to mesh routers. Infrastructure/Backbone WMNs are the most commonly used type. For example, community and neighborhood networks can be built using infrastructure meshing. The mesh routers are placed on the roof of houses in a neighborhood, which serve as access points for users inside the homes and along the roads. Typically, two types of radios are used in the routers, i.e., for backbone communication and for user communication, respectively. The mesh backbone communication can be established using long-range communication techniques including directional antennas[15].

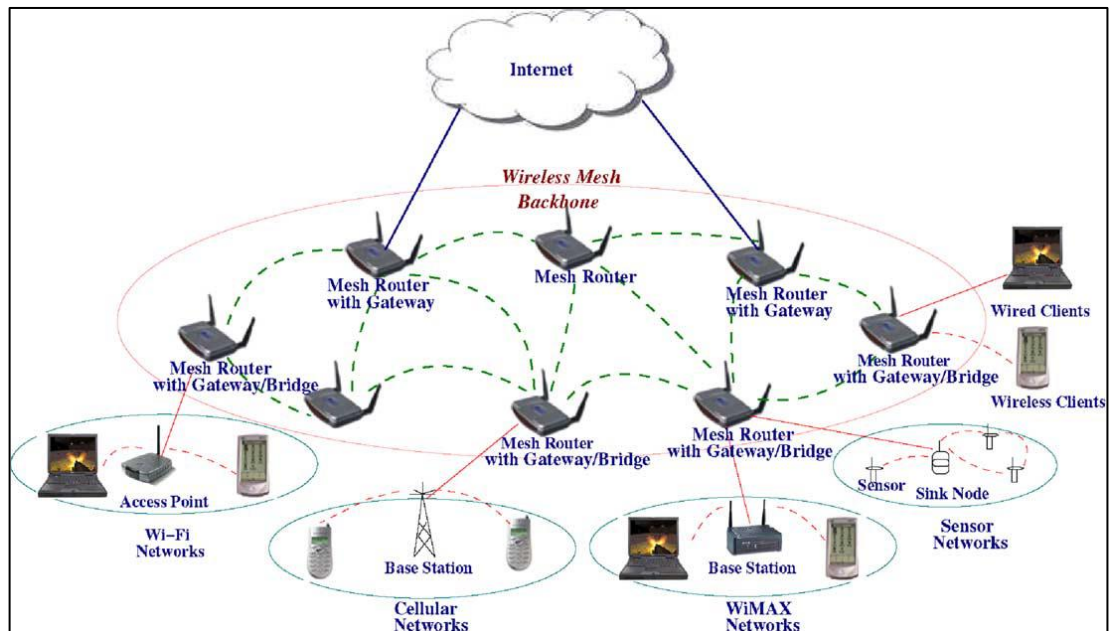


Figure 1.3 Infrastructure/backbone WMNs.[15]

2.2 Client WMNs

The second architecture (Client meshing) provides peer-to-peer networks among client devices[20]. In this type of architecture, client nodes constitute the actual network to perform routing and configuration functionalities as well as providing end-user applications to customers. Hence, a mesh router is not required for these types of networks. The basic architecture is shown in Figure 1.4. In Client WMNs, a packet destined to a node in the network hops through multiple nodes to reach the destination. Client WMNs are usually formed using one type of radios on devices. Moreover, the requirements on end-user devices is increased when compared to

infrastructure meshing, since, in Client WMNs, the end-users must perform additional functions such as routing and self-configuration[15].

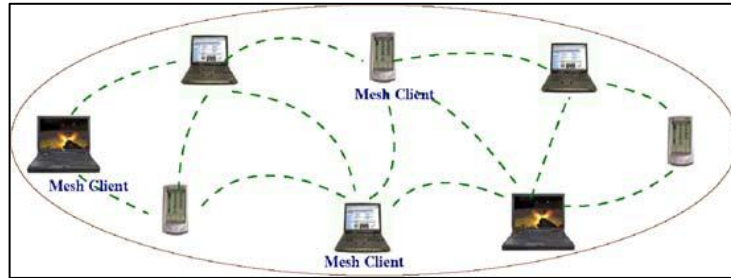


Figure 1.4 Client WMNs.[15]

2.3 Hybrid WMNs

This architecture is the first and the second architectures as shown in Figure 1.5. Mesh clients can access the network through mesh routers as well as directly meshing with other mesh clients. While the infrastructure provides connectivity to other networks such as the Internet, Wi-Fi, WiMAX, cellular, and sensor networks, the routing capabilities of clients provide improved connectivity and coverage inside the WMN. The hybrid architecture will be the most applicable case in our opinion[15].

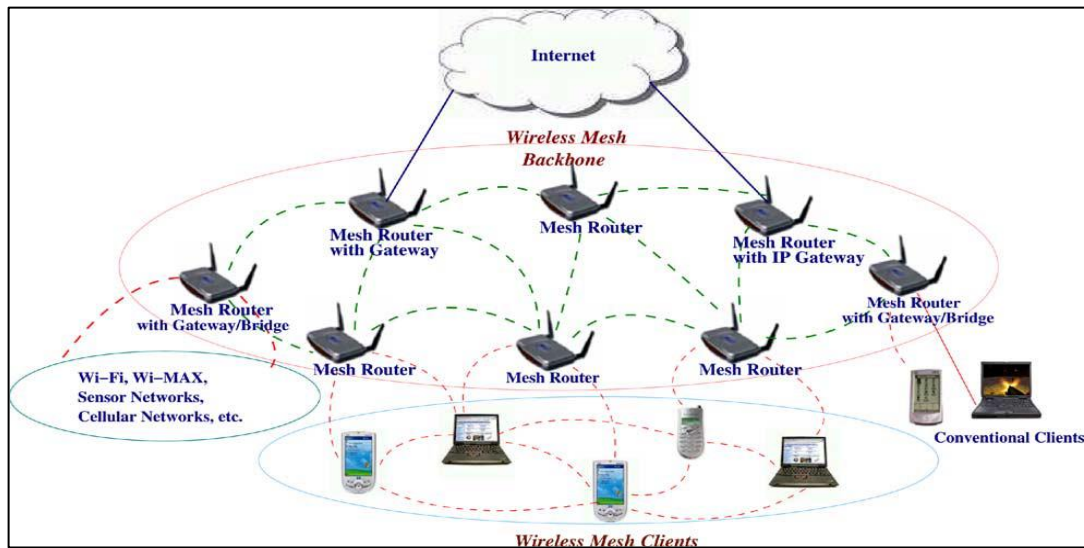


Figure 1.5 Hybrid WMNs.[15]

3. Characteristics

The characteristics of WMNs are shown as follows:

3.1 Multi-hop wireless network

An objective to develop WMNs is to extend the coverage range of current wireless networks without sacrificing the channel capacity. Another objective is to provide non-line-of-sight (NLOS) connectivity among the users without direct line-of-sight (LOS) links. To meet these requirements, the mesh-style multi-hopping is indispensable[12], which achieves higher throughput without sacrificing effective radio range via shorter link distances, less interference between the nodes, and more efficient frequency re-use.

3.2 Support for ad hoc networking, and capability of self-forming, self-healing, and self-organization

WMNs enhance network performance, because of flexible network architecture, easy deployment and configuration, fault tolerance, and mesh connectivity, i.e., multipoint-to-multipoint communications[13]. Due to these features, WMNs have low upfront investment requirement, and the network can grow gradually as needed.

3.3 Mobility dependence on the type of mesh nodes

Mesh routers usually have minimal mobility, while mesh clients can be stationary or mobile nodes[15].

3.4 Multiple types of network access

In WMNs, both backhaul access to the Internet and peer to peer (P2P) communications are supported[14]. In addition, the integration of WMNs with other wireless networks and providing services to end-users of these networks can be accomplished through WMNs.

3.5 Dependence of power-consumption constraints on the type of mesh nodes

Mesh routers usually do not have strict constraints on power consumption. However, mesh clients may require power efficient protocols. As an example, a mesh-capable sensor requires its communication protocols to be power efficient. Thus, the MAC or routing protocols optimized for mesh routers may not be appropriate for mesh clients such as sensors, because power efficiency is the primary concern for wireless sensor networks [15].

3.6 Compatibility and interoperability with existing wireless networks

For example, WMNs built based on IEEE 802.11 technologies must be compatible with IEEE 802.11 standards in the sense of supporting both mesh capable and conventional Wi-Fi clients. Such WMNs also need to be inter-operable with other wireless networks such as WiMAX, and cellular networks[15].

3.7 Integration

WMNs support conventional clients that use the same radio technologies as a mesh router. This is accomplished through a host-routing function available in mesh routers. WMNs also enable integration of various existing networks such as Wi-Fi, the Internet, and cellular and sensor networks through gateway/bridge functionalities in the mesh routers. Consequently, users in one network are provided with services in other networks, through the use of the wireless infrastructure. The integrated wireless networks through WMNs resembles the Internet backbone, since the physical location of network nodes becomes less important than the capacity and network topology[15].

3.8 Dedicated routing and configuration

In ad hoc networks, end-user devices also perform routing and configuration functionalities for all other nodes. However, WMNs contain mesh routers for these functionalities. Hence, the load on end-user devices is significantly decreased, which provides lower energy consumption and high-end application capabilities to possibly mobile and energy constrained end-users. Moreover, the end-user requirements are limited which decreases the cost of devices that can be used in WMNs[15].

3.9 Multiple radios

As discussed before, mesh routers can be equipped with multiple radios to perform routing and access functionalities. This enables separation of two main types of traffic in the wireless domain. While routing and configuration are performed between mesh routers, the access to the network by end users can be carried out on a different radio. This significantly improves the capacity of the network. On the other hand, in ad hoc networks, these functionalities are performed in the same channel, and as a result, the performance decreases[15].

3.10 Mobility

For WMNs capable of supporting user mobility, it is necessary for the physical layer to support the shift in frequency and adapt to the fast fading conditions commonly associated with mobile users[14, 8].

4. Key Benefits of a Wireless Mesh Network

4.1 Less Expensive than Traditional Networks

Using fewer wires means it costs less to set up the wireless mesh network. The wireless mesh network is used particularly for large areas of coverage. Using wireless mesh networks eliminate the cost and complexity of installing fiber (wires) between buildings, on campus grounds and business parks[21].

4.2 Wireless Mesh is extremely adaptable and expandable

As more or less coverage is needed, wireless mesh nodes can be added or removed. Wireless Mesh is very useful for areas where there is lack of sight or where network configurations are intermittently blocked. An example of an area as this would be an amusement park where a Ferris wheel may occasionally block the signal from the wireless access point. With wireless mesh, adding more wireless mesh nodes will adjust to find a clear signal. Wireless Mesh is also extremely convenient where wall connections may be lacking, such as in outdoor environments, warehouses or transportation settings[21].

4.3 Wireless Mesh Networks Support High Demand

Public Safety and emergency response demand wireless connectivity that supports coverage of large geographic areas, high speed mobility and high-quality video surveillance. Wireless Mesh Networks are ideal to deliver high throughput and highly reliable wireless connectivity[21].

5. WMN deployments and applications

5.1 Connectivity services

- Broadband Internet access (fixed or mobile)
- L2 connectivity between clients
- Extended WLAN coverage
- Metropolitan networks (usually 802.11 + 802.16)

5.2 Use cases scenarios

- General support for A/V, data and multimedia applications.
- Remote monitoring and control, Video surveillance.
- Emergency response.
- Military and field applications.
- Community networks.
- Enterprise networks.
- Public transportation Internet access.
- Multimedia home networking.

5.3 Use cases examples

- Transportation system:

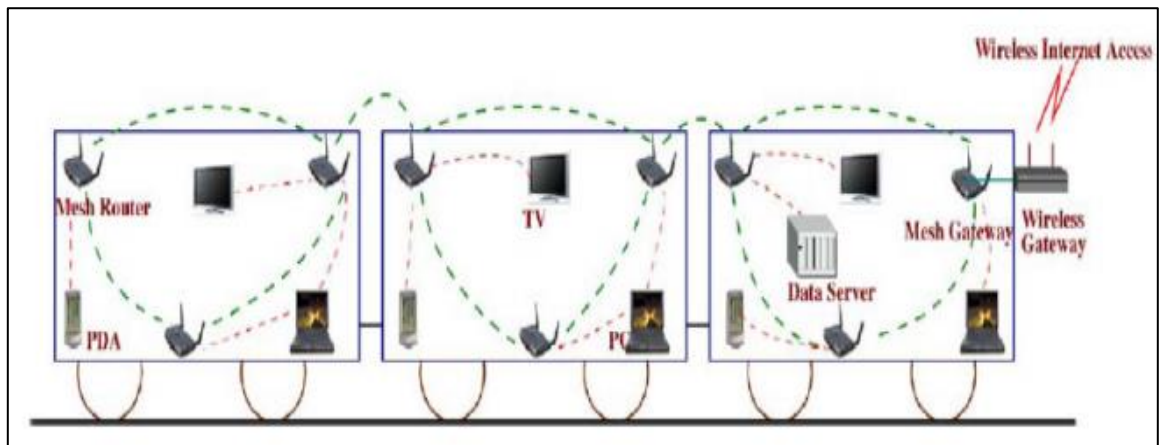


Figure 1.6 Transportation system

- Broadband Home Networking [15]:

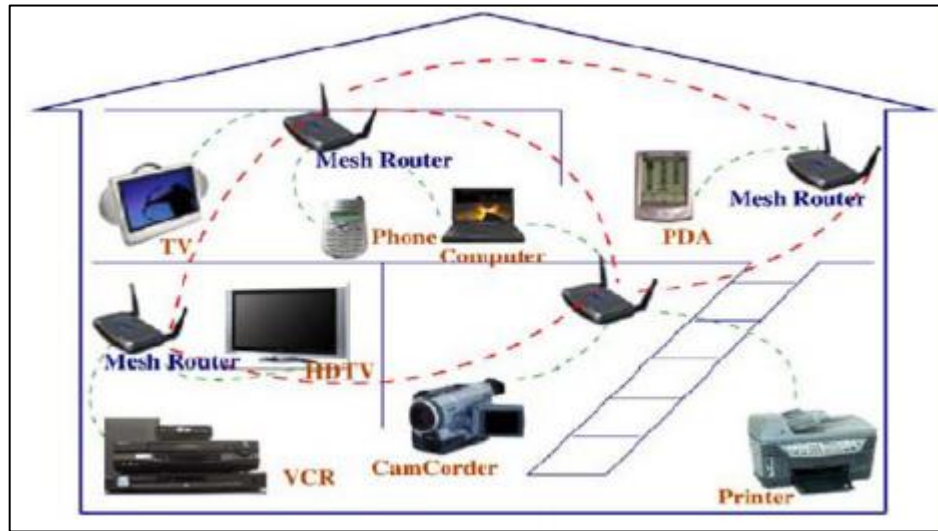


Figure 1.7 Broadband Home Networking

- Enterprise network:

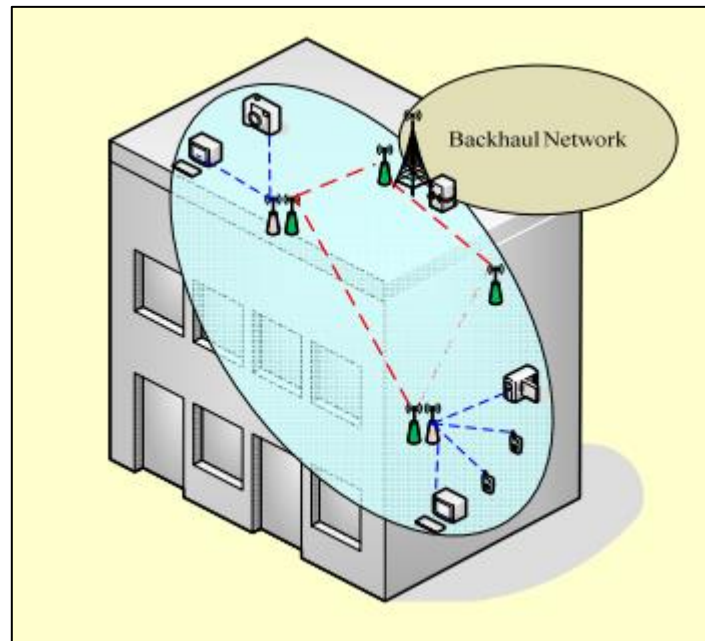


Figure 1.8 Enterprise network

- Community network:

WMNs are replacing DSL based solution [18]

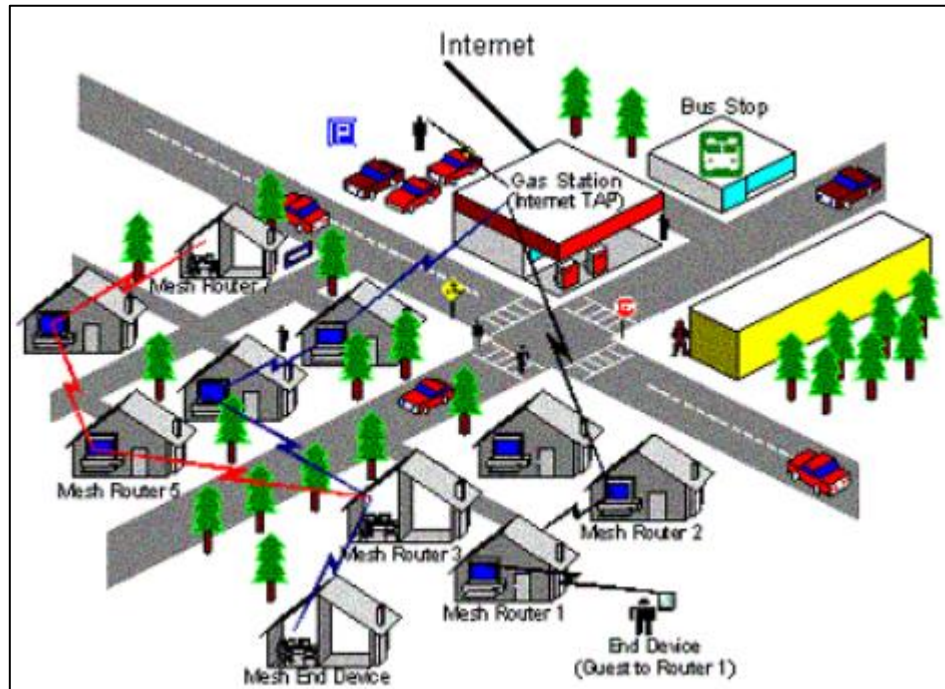


Figure 1.9 WMNs are replacing DSL based solution

- Military networks [19]:

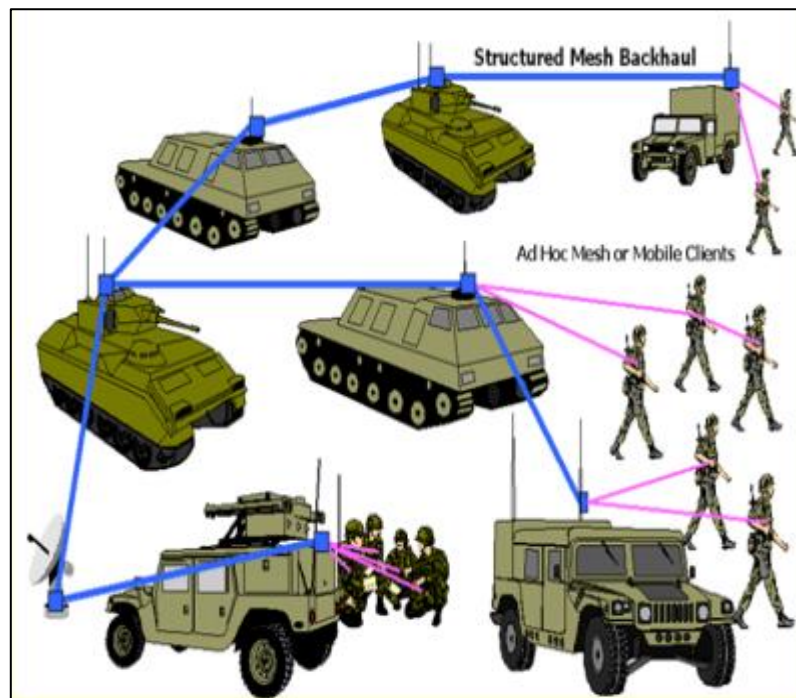


Figure 1.10 Military networks

- Emergency response [19]:

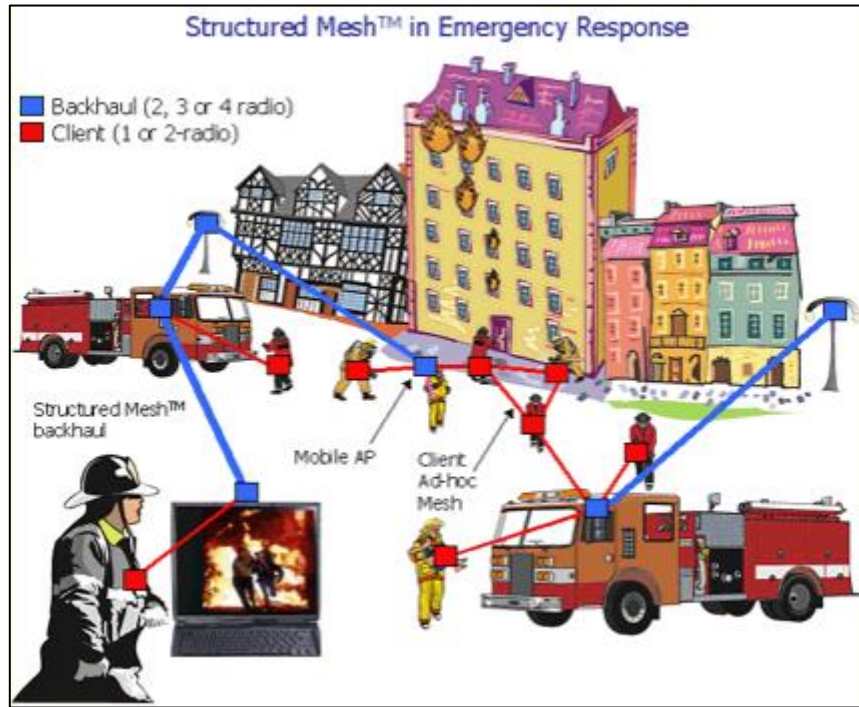


Figure 1.11 Emergency response

6. Conclusion

Wireless Mesh Networks deliver both indoor and outdoor connectivity. Wireless Mesh is an instant, complete solution for covering large areas without sacrificing quality of the wireless network. Wireless Mesh Networks are a reliable source of wireless connectivity for a variety of public safety applications, parking garages, campus grounds, schools, business parks, and other large outdoor facilities.

**ROUTER
PLACEMENT
PROBLEM IN
WIRELESS MESH
NETWORKS**

CHAPTER

02

CHAPTER 02

ROUTER PLACEMENT PROBLEM IN WIRELESS MESH NETWORKS

1. Introduction

In this chapter we will explain the Router Placement Problem in Wireless Mesh Network, A Wireless Mesh Network (WMN) is a multi-hop wireless network, in which stationary mesh routers wirelessly relay traffic on behalf of other mesh routers or client stations and thereby form a wireless backbone[10]. An overview of WMNs can be found in Chapter 01, Among the most important problems of WMN networks: the problem of placement of wireless routers which consists of choosing the best positions of mesh routers so that the coverage of the clients and connectivity of the routers are both optimized. This chapter deals with the problem of wireless mesh networks and describes its various variants.

2. Routers placement problems in WMN

This problem can be divided into several types, depending on whether the nodes of the network are static or dynamic: dynamic placement (WMN-dynRNP) or static placement (WMN-RNP)

2.1 WMN-dynRNP

The problem of dynamic router node placement (dynRNP) in wireless mesh networks (WMNs) is concerned with determining a dynamic geographical placement of mesh routers to serve mobile mesh clients at different times, so that both network connectivity (i.e., the greatest topology sub graph component size) and client coverage (i.e., the number of the served mesh clients) are maximized. Mesh clients are wireless devices associated with users, and in real world, the users with same interests or some social relationship have higher chance to gather and move together geographically, i.e., they form a community, and the WMN with multiple communities can be regarded as a social network.

2.2 WMN-RNP

The problem of router node placement (RNP) in WMNs has been studied, e.g., see [1, 3, 4] Such a combinatorial optimization problem considers a WMN composed of mesh clients and mesh

routers in which mesh routers serve as the access point towards mesh clients and connect to other mesh routers through point-to-point wireless links[6].

3. Problem Description

WMN consists of 3 types of nodes: mesh routers, mesh clients and mesh gateways. Each mesh client can only communicate with the node within the same radio coverage or any node that can be accessed via multi-hop router communications [6]. That is, any mesh client cannot communicate with other nodes in the network if it is not located in the radio coverage of some mesh routers. Consider a WMN with n mesh routers and m mesh clients deployed in a two-dimensional geographical area. Let the mesh nodes in the WMN be denoted by $U = R \cup C$ in which:

- $R = \{r_1, r_2 \dots r_n\}$ where each r_i represents a mesh router and has a radio coverage range of size γ_i .
- $C = \{c_1, c_2 \dots c_m\}$ where each c_i represents a mesh client.

We assume that the client locations are fixed in the deployment area by a random distribution, each mesh client $c_i \in C$ is located at $D_t(c_i) \in \mathbb{R}^2$ in the deployment area, According to the deployment of mesh clients at each time point, we determine the placements of mesh routers, denoted by: $D(R) = \{D(r_1), D(r_2), \dots, D(r_n)\}$, Let the circle centered at the location $D(r_i)$ of nodes r_i with radius γ_i be denoted by Y_i . For a determined placement of mesh routers, we can establish a topology graph $G = (U, E)$.

- $U = R \cup (C \setminus S)$, where S is the set of mesh clients that switch off network access or do not access a network.
- For any two mesh routers $r_i, r_j \in R$ $edge(r_i, r_j) \iff Y_i \cap Y_j \neq \emptyset$.
- For any mesh client $c_i \in C \setminus S$ and any mesh router $r_j \in R$ $edge(c_i, r_j) \iff D(c_i) \in Y_j$.

Note, for simplicity, we assume that two mesh routers can communicate only if their radio coverage is overlapped, the assumption is reasonable because 2 mesh routers with overlapping radio coverage imply that they are so close geographically that they can communicate with each other [6].

It should be noticed that graph G may not be connected, i.e., G may consist of several sub-graph components. In order to increase the network connectivity of WMN, we would like to make

the size of the greatest sub-graph component as large as possible. However, a large size of the greatest component does not imply a wide radio coverage of mesh clients, hence, we consider client coverage as the other concerned design on placement of mesh routers.

Assume that there is h sub-graph components $G_1, G_2 \dots G_h$ in G , i.e.

- $G = G_1 \cup G_2 \cup G_h$, and $G_i \cap G_j = \emptyset$ for $i, j \in \{1 \dots h\}$.

The size of the greatest sub-graph component in G can be expressed as follows:

$$\phi(G) = \max \{|G_i|\}, \text{ for } i \in \{1 \dots h\}. \quad (1)$$

The client coverage can be expressed as follows:

$$\psi(G) = |\{i, d(ci) > 0, \text{ for } i \in \{1 \dots m\}\}|. \quad (2)$$

Where $d(ci)$ is the degree of node ci in topology graph G .

In order to explain the above notation, we demonstrate two examples for a WMN with 5 mesh routers $r_1 \dots r_5$ and 14 mesh clients $c_1 \dots c_{14}$ in Figure 2.1, in which each mesh router has a different-size radio coverage range. If two mesh routers have a radio coverage overlap, they are connected by a green dashed link (e.g., see the green dashed link between r_1 and r_2 in Figure 2.1(a)). If a mesh client is located within the radio coverage range of a mesh router, they are connected by a red dotted link (e.g., c_1 is connected to r_1 and r_2 by two red dotted links in Figure 2.1(a) because it is located within both of the radio coverage ranges of r_1 and r_2 simultaneously). Hence, the topology graph in Figure 2.1(a) has three sub-graph components among which the size of the greatest component is 8 (i.e., $\phi=8$), and 10 mesh clients $c_1-c_7; c_{11}-c_{13}$ are covered (i.e., $\psi=10$). If we move mesh router r_2 to the center location of the deployment area as shown in Figure 2.1(b), then all the components are merged into one with size 16 (i.e., $\phi=16$), and 11 mesh clients are covered (i.e., $\psi=11$). That is, both two network performance measures are improved by moving only the mesh router r_2 . As a result, it is of interest to investigate how to place the mesh routers so that both two network performance measures are optimal.

Furthermore, the problem becomes more complicated if mesh clients have mobility and can switch on or off their network access as time goes by. For example, consider a scenario in

Figure 2.2(a) transformed from Figure 2.1(b). In this scenario, mesh routers c6 and c11 switch of their network access.

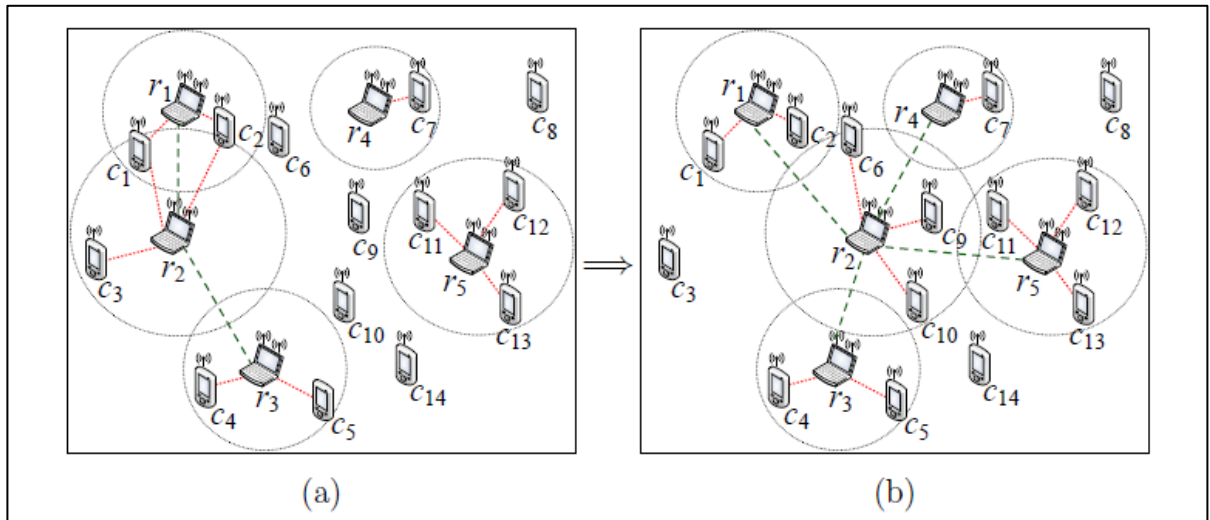


Figure 2.1 2 placements for the same WNN with 5 mesh routers and 14 mesh clients[6].

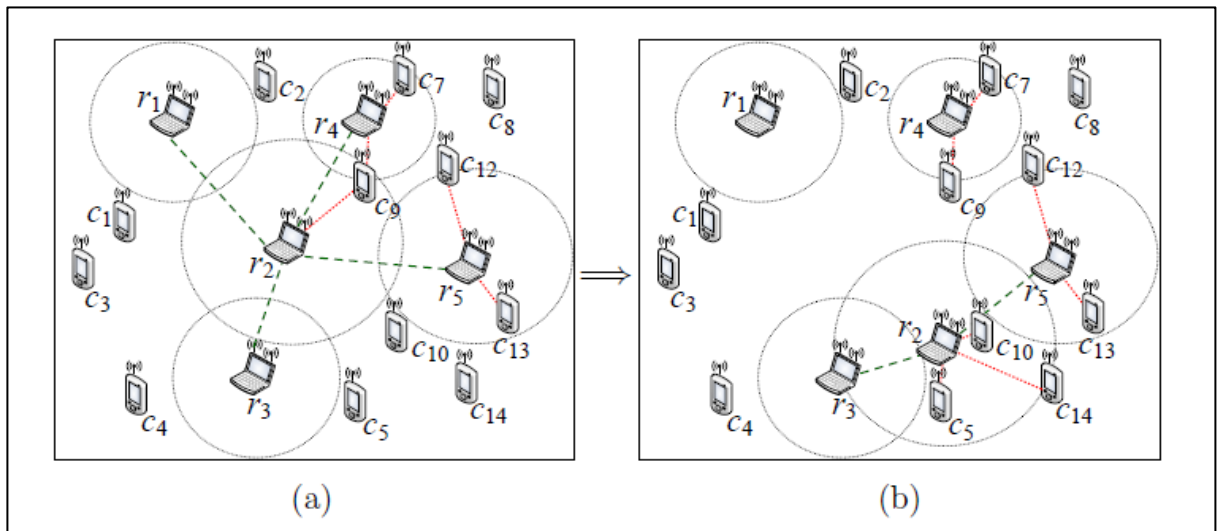


Figure 2.2 the new topology[6].

Note:

Red line: means connection between client and router.

Dashed line: means connection between routers.

4. A PSO approach to the Routers placement problems in WMN

The beauty of PSO lies in its simplicity and ease of applicability, PSO is an algorithm used for solving optimization problems based on the social intelligence of a swarm of particles, proposed by [11, 5]. The basic idea of PSO is to simulate a swarm of particles' social behavior of searching for a food source in a multi-dimensional search space. It is assumed that each particle has a position and a velocity, which are updated by iteration. Each particle moves toward the food source by the guidance of its local best-known position (which has been found so far by itself) and also the global best-known position (which has been found so far by any particle of the swarm). The food source is associated with the global optimal solution of the concerned optimization problem, while the location of each particle in the search space is associated with a candidate solution. Hence, the global optimal solution can be found if almost all the candidate solutions move to the same position in the search space.

This part gives in detail our PSO approach to the WMN-dynRNP problem. We first give the solution representation of each particle, then the fitness function used in the PSO approach, then the scheme of updating positions of each particle, and finally our PSO algorithm for the WMN-dynRNP problem.

4.1 Solution exemplification

The solution of the concerned WMN-MNP problem is a placement of n mesh routers in a two-dimensional $W \times H$ area, whose lower-left corner is placed at the origin of an $x \times y$ plane. That is, the (x, y) coordinates of the n mesh routers should be determined for each candidate solution [2].

In PSO, each particle k represents a candidate solution, which is determined by the following three vectors and two fitness values:

- The X-vector: $X_k = (X_{k1}, X_{k2} \dots X_{k(2n)})$, records the current position of the particle in the search space in which $(X_{k(2i-1)}, X_{k(2i)})$, denotes the (x, y) coordinate of mesh router r_i for $i = 1, 2 \dots n$.
- The P-vector: $P_k = (P_{k1}, P_{k2} \dots P_{k(2n)})$, register the location of the best solution established so far by particle k .

- The V-vector: $V_k = (V_{k1}, V_{k2} \dots V_{k(2n)})$, register the velocity along which particle k will move.
- $f(X_k)$: register the fitness of X_k .
- $f(P_k)$: register the fitness of P_k .

Since all of the mesh routers are placed within a W*H area, we have the following constraints:
 $\forall i \in \{1, 2 \dots n\}$.

$$0 \leq X_{k(2i-1)} \leq W \quad (3)$$

$$0 \leq X_{k(2i)} \leq H \quad (4)$$

$$-W \leq V_{k(2i-1)} \leq W$$

$$-H \leq V_{k(2i)} \leq H$$

In order to avoid drastic change of velocities, we have the following constraints for velocities:
 $\forall i \in \{1, 2 \dots n\}$.

$$-V_{max} \leq V_{k(2i-1)} \leq V_{max} \quad (5)$$

Where Vmax is a given constant that is no more than max {W*H}.

From the view point of the whole swarm, a vector and a fitness of the best solution that have been found so far are stored in each iteration:

- $P^* = (P_1^*, P_1^* \dots P_{2n}^*)$, register the location of the best solution found so far by all particles.
- $f(P^*)$ register the fitness of P^* .

That is, after finishing the PSO algorithm, P^* and $f(P^*)$ store the location and the fitness value (objective value) of the final solution, respectively.

4.2 Fitness function

Given a placement X_k of mesh routers for particle k, we can obtain a graph G_k underlying the WMN. Recall that the objective of our concerned problem is to maximize the size of the greatest sub-graph component $\phi(G_k)$ and the client coverage $\psi(G_k)$, which can be calculated by Equation (1) and Equation (2), respectively.

$$f(X_k) = \lambda \cdot \phi(G_k) / n + m + (1 - \lambda) \cdot \psi(G_k) / m \quad (6)$$

Where λ is a parameter in the range (0, 1) that controls the balance between the two terms of the equation. Note that the denominator of each term of the equation is used for normalization.

4.3 Updating position

Each iteration of the main loop of PSO updates the velocity vector V_k by the following formula:

$$V_{k'} = \omega [V_k + c1 \cdot r1 \cdot (P_k - X_k) + c2 \cdot r2 \cdot (P^* - X_k)] \quad (7)$$

Where $c = c1 + c2 > 4$, $w = 2 / |2 - c - (c2 - 4c)|$, $r1$ and $r2$ are two random numbers 0 and 1. The position of each particle k by the following formula:

$$X_{k'} = X_k + V_{k'} \quad (8)$$

Where $X_{k'}$ and $V_{k'}$ are the new values of position vector X_k and velocity vector V_k of particle k , respectively.

Since the computation of the above two equations require only basic arithmetic operations, they can be computed in $O(1)$ time. Note that a PSO approach with updating formula of Equation (7) is called the PSO with constriction coefficient [9].

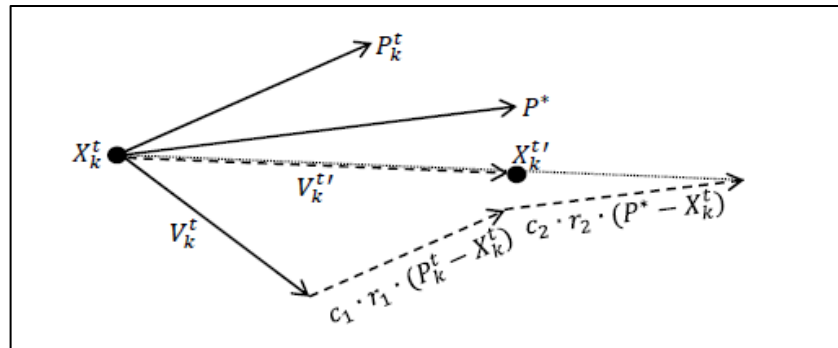


Figure 2.3 clarification of updating the location of particle k

4.4 PSO Algorithm

For each $k \in \{1, 2 \dots \eta\}$ do $t=0$;

If $t > 0$ then

$X_k^t \leftarrow X_k^{t-1}$, and calculate its fitness $f(X_k^t)$.

$V_k^t \leftarrow V_k^{t-1}$

Else

Initialize particle K's position $X_k^t = (X_{k1}^t \dots X_{k(2n)}^t)$ randomly where $X_{k(2i-1)}^t \sim u(0, W)$ and $X_{k(2i)}^t \sim u(0, H)$ for each $i \in \{1, 2, \dots, n\}$. and calculate its fitness $f(X_k)$.

Initialize particle K's velocity $V_k^t = (V_{k1}^t \dots V_{k(2n)}^t)$ randomly where $V_{k(2i-1)}^t \sim u(0, W)$ and $V_{k(2i)}^t \sim u(0, H)$ for each $i \in \{1, 2, \dots, n\}$.

End if

$P_k^t \leftarrow P_k^{t-1}$ and $f(P_k^t) \leftarrow f(P_k^t)$

If $f(P_k^t) > f(P^*)$ then $P^* \leftarrow P_k^t$ and $f(P^*) \leftarrow f(P_k^t)$

End if

End for

Repeat

For each $k \in \{1, 2 \dots \eta\}$ do

Update particle K's velocity V_k^t by ----- (7).

$\forall i \in \{1, 2 \dots 2n\}$, V_i^t is truncated if violating constraint ----- (5).

Update particle K's position X_k^t by ----- (8).

$\forall i \in \{1, 2 \dots 2n\}$, X_i^t is truncated if violating constraint (3) and (4).

Calculate $f(X_k^t)$.

If $f(X_k^t) > f(P^*)$ then

$P_k^t \leftarrow X_k^t$ and $f(P_k^t) \leftarrow f(X_k^t)$.

If $f(P_k^t) > f(P^*)$ then

$P^* \leftarrow P_k^t$ and $f(P^*) \leftarrow f(P_k^t)$.

End if

End if

End if

Until {the maximum iteration τ is reached or $f(P^*)$ exceeds a threshold}

Output $f(P^*)$ as the solution at the t the key time point.

5. Rechargeable router system

We consider establishing a WMN comprising of just sun powered controlled rechargeable mesh routers, albeit self-powered MRs are anything but difficult to deploy, the functioning period of such a MR is compelled by the energy supply from the batteries. Sunlight based board connected to MR can recharge the batteries to draw out the operation. even so, sun based board is made of constrained size and energy revive rate is limited.

5.1 Energy flow model

To portray the change of energy put in battery, we partition the continuous time-line into consecutive slots with equal duration and indexed by $k=1, 2, 3 \dots$, we accept that MRs are generally synchronous to do associations with MCs in each slot. The energy charging and discharging procedure of a router can be displayed by a discrete-time energy flow model as takes after:

- $E(k) = E(k-1) + C(k) - D(k)$, for $k=1, 2, 3 \dots$ (9) [16]

Where $E(k)$ is the residual energy of the router after the k th slot. When $k=0$, $E(0)$ denotes the certain value of initial energy stored in routers. $C(k)$ is the energy harvested in the k th slot, and $D(k)$ is the energy consumed for data transmission in both downlink and uplink in the k th slot.

5.2 Energy charging model

In this work, we receive energy charging model, specifically sun oriented power from naturel condition, the pick energy is dynamic and variable for each schedule opening of a day, we consider the same charging model for every one of the routers, the charging model is expressed as follow:

- $C(k) = C_{Max} * (\frac{-1}{36}k^2 + \frac{2}{3}k - 3)$ (10) [16]

Where C_{Max} is the maximum charging ability of a solar panel, since the charging model follows the same trend every day, we can calculate the sequence number of any time slot k in a day according to $k=\text{mod}(k, 24)$, which is to obtain the remainder after k is divided by 24 h.

5.3 Energy discharging model

In this work, we consider that the energy discharging model is random for each router in each day, in order to bring the truth closer.

6. Conclusion

In this chapter, we have described the problem of placing wireless routers in a WMN. Two variants of the problem have been identified: dynamic placement that dynamically identifies locations and the static placement. And we describe a rechargeable router system for each router (charging and discharging models), and several papers have been presented to solve this problem.

**DESIGN,
IMPLEMENTATION
AND RESULTS**

CHAPTER

03

CHAPTER 03

DESIGN, IMPLEMENTATION AND RESULTS

1. Introduction

In this chapter, we will talk about our implementation and how we adopt the algorithms for solve the problem of router placement in wireless mesh network, and a detailed survey on those algorithms can be found in Chapter 02, where we analyzed all the most important parameters, and In this chapter, we will explain the implementation of each proposed algorithm, and in this work, we opted for an object-oriented programming language, fast and offering the possibility to manipulate high-level graphical interfaces. For this purpose, we chose the C++ language with the development environment Embarcadero C++ Builder, Finally, we will expose the results of the tests performed to evaluate the performance of the algorithm.

2. Embarcadero C++ Builder

C++ Builder is a rapid application development (RAD) environment (Figure 3.1), originally developed by Borland and as of 2009 owned by Embarcadero Technologies, for writing programs in the C++ programming language targeting Windows NT (IA-32 and x64), OS X, iOS and Android. C++Builder combines the Visual Component Library and IDE written in Delphi with a C++ compiler, Develop for many platforms in fast, modern C++ with an enhanced Clang compiler. Seamlessly compile and deploy across platforms and devices. Design a UI once, then quickly specialize for each platform or device - much like sub classing code. Use robust and powerful frameworks for enterprise-strength databases, REST, cloud, and more2. [22]

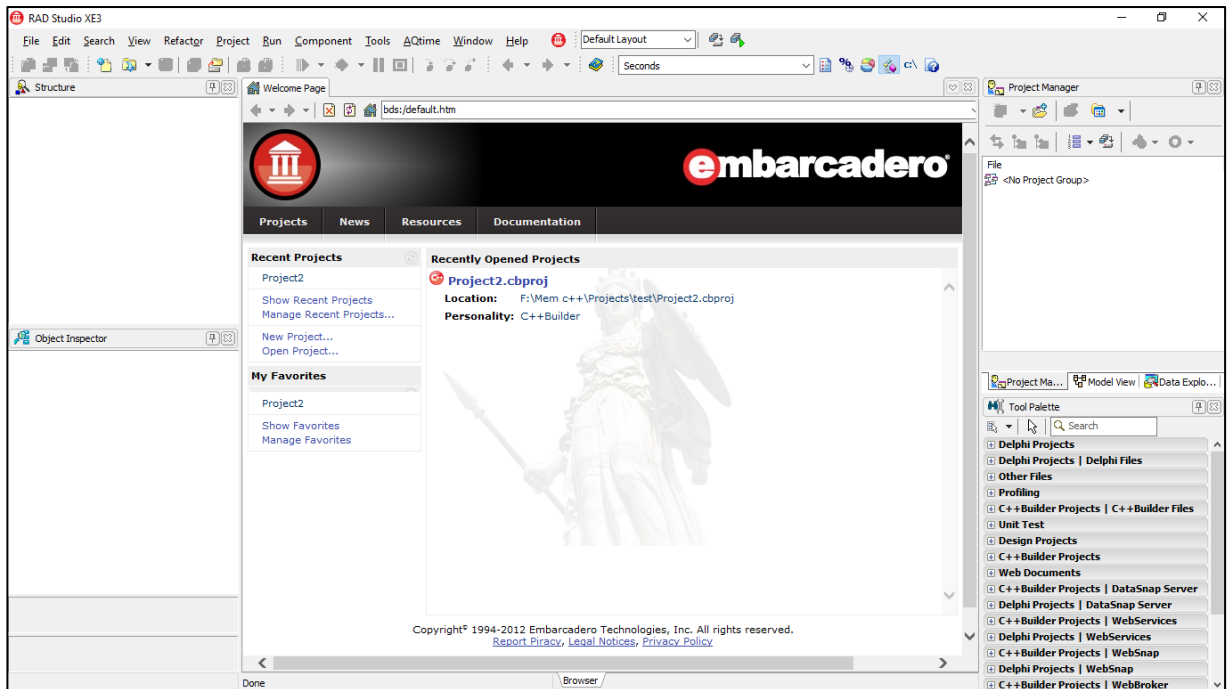


Figure 3.1 Embarcadero C++ Builder

3. Design and implementation

In this work, we want to solve the problem of router placement in wireless mesh network, we combine the PSO algorithm and the flow charging model, we designed our program through C ++, so that we can represent all the elements and apply Algorithms in an appropriate manner, the most important elements of this work are:

- The PSO algorithm and her parameters:
 - Number of routers and clients.
 - Particles, best particles and global best (the best of the best).
 - Positions and velocity.
 - Iteration number.
 - Fitness, Coverage and connectivity.
- The Energy flow model and his parameters:
 - Energy model.
 - Charging model.
 - Discharging model.
 - Iteration number.

3.1 Conception of the PSO algorithm

In this section, we implement the PSO algorithm in C++, we had so much element, variables and functions, so I will talk just about the essential ones.

Graph-fill ():

In this function, the essential parameter is the graph matrix (**graph** [] []), so here we receive the best particle from PSO function and just calculated the distance between the client's positions and all the routers positions from the received particles, if that distance less than the router radius, we allocate that client to the specific router, with adding **1** in the graph between this client and that router, else if the distance more than the radius of the router so we add **0** to the graph (Figure 3.2).

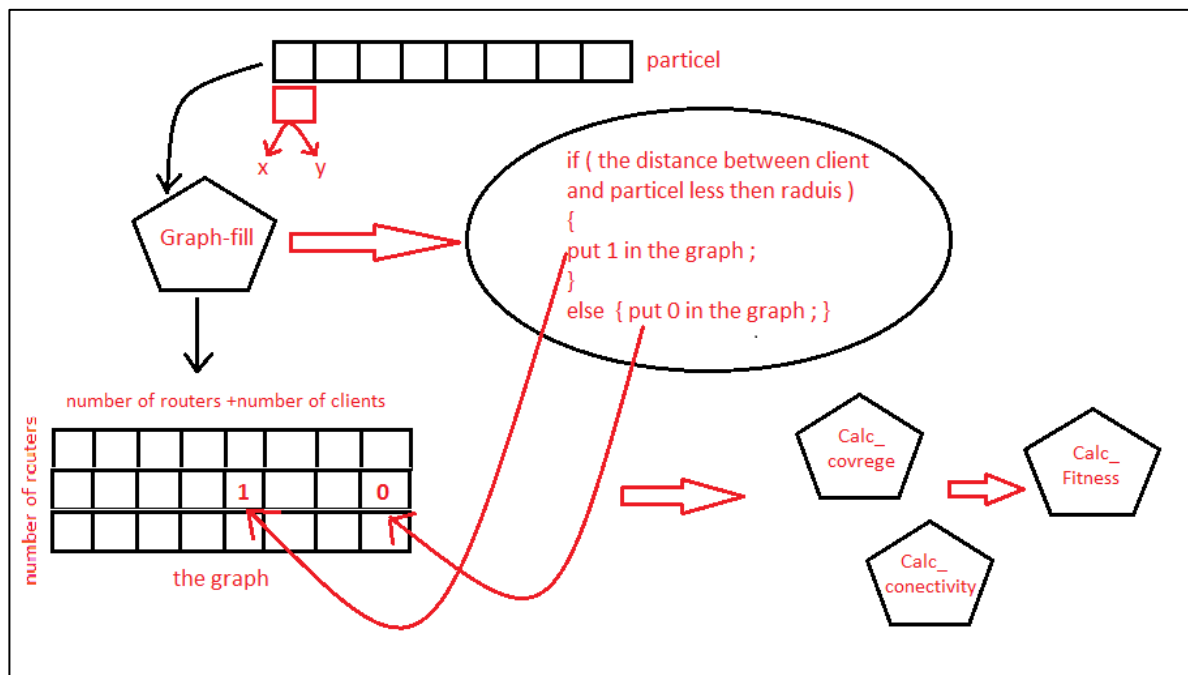


Figure 3.2 implementation description

```

1 void rnp::graphfill(int p) {
2     int n = rr2 + nbrc;
3     double dist = 0;
4     double x, y;
5     rd1 = radius;
6     for (i = 0; i < nbrr; i++) {
7         for (j = 0; j < (nbrr + nbrc); j++) {
8             graph[i][j] = 0;
9         }
10    }
11    for (i = 0; i < rr2; i++) {
12        for (j = 0; j < n; j++) {
13            if (i == j) {
14                graph[i][j] = 0;
15            }
16            else {
17                if (j < rr2) {
18                    x = particles[p][j].x;
19                    y = particles[p][j].y;
20                    rd1 = 2 * radius;
21                }
22                else {
23                    x = client[j - nbrr].x;
24                    y = client[j - nbrr].y;
25                    rd1 = radius;
26                }
27                dist = sqrt(pow(x - particles[p][i].x, 2) +
28                    pow(y - particles[p][i].y, 2));
29
30                if (dist < rd1) {
31                    graph[i][j] = 1;
32                }
33                else {
34                    graph[i][j] = 0;
35                }
36            }
37        }
38    }
39 }

```

Calc-coverage ():

In this function and associate with the result of Graph-fill we calculated the coverage of clients in parameter named **cov** ($\psi(G)$).

```

1 int rnp::calc_covrege() {
2     int covrege = 0;
3
4     for (int jj = rr2; jj < (rr2 + nbro); jj++) {
5         for (int ii = 0; ii < rr2;) {
6
7             if (graph[ii][jj] == 1) {
8                 covrege++;
9                 jj++;
10                ii = 0;
11            }
12            else
13                ii++;
14        }
15    }
16    return covrege;
17 }
18

```

Router-connectivity () and Client-connectivity ():

In this two functions, we calculated the connectivity between routers then between clients and combine them in one parameter named **conn**, the latter means the subnetwork which contains as many as possible clients ($\phi(G)$).

```

1 void rnp::router_connectivity() {
2     for (i = 0; i < rr2; i++) {
3         for (j = 0; j < rr2; j++) {
4             if ((graph[i][j] == 1) && (tab_r[j] != tab_r[i])) {
5                 for (int k = 0; k < rr2; k++) {
6                     if (tab_r[k] == tab_r[j]) {
7                         tab_r[k] = tab_r[i];
8                     }
9                 }
10            }
11        }
12    }
13 }
14
15 for (i = 0; i < rr2; i++) {
16     tab_sg[i] = 0;
17 }
18 for (i = 0; i < rr2; i++) {
19     int bc = tab_r[i];
20     tab_sg[bc]++;
21 }
22 }

```

```

1  void rnp::client_connectivity() {
2      int x;
3      for (j = rr2; j < (rr2 + nbrc); j++) {
4          int dd = 0;
5          while ((graph[dd][j] == 0) && (dd < rr2)) {
6              dd++;
7          }
8          if (dd < rr2) {
9
10             x = tab_r[dd];
11             tab_sg[x]++;
12
13         }
14     }
15 }
16
17

```

Calc-fitness ():

In this function, we calculated the value of fitness associate with the tow parameters **cv** and **conn** with formula (6).

```

1  double rnp::calc_fitness(int cv, int cn) {
2
3      fitness = ((lamda * (double)cn / (double)(nbrc + rr2)) +
4                ((1.0 - lamda) * ((double)cv / (double)nbrc)));
5
6      return fitness;
7
8  }

```

Update-velocity ():

In this function, we updated the velocity after every iteration with the velocity update formula (7).

```

1 void rnp::update_vilo() {
2     c1 = 2.0;
3     c2 = 3.0;
4
5     w = 2.0 / abs(-3.0 - sqrt(5.0));
6     vx = 0.0;
7     vy = 0.0;
8     r1 = static_cast<float>(rand()) / static_cast<float>(RAND_MAX);
9     r2 = static_cast<float>(rand()) / static_cast<float>(RAND_MAX);
10    for (i = 0; i < nbr_of_part; i++) {
11        for (j = 0; j < nbrr; j++) {
12            vx = w * (vilocity[i][j].x + c1 * r1 * (best_part[i][j].x -
13                particles[i][j].x) + c2 * r2 * (gbest_tab[j].x -
14                particles[i][j].x));
15            vy = w * (vilocity[i][j].y + c1 * r1 * (best_part[i][j].y -
16                particles[i][j].y) + c2 * r2 * (gbest_tab[j].y -
17                particles[i][j].y));
18            vilocity[i][j].x = vx;
19            vilocity[i][j].y = vy;
20        }
21    }
22 }
23
24 }

```

Update-position ():

In this function and associate with the result of Update-velocity (), we updated the positions of the routers with the position update formula (8).

```

1 void rnp::update_position() {
2     for (i = 0; i < nbr_of_part; i++) {
3         for (j = 0; j < nbrr; j++) {
4
5             particles[i][j].x = (particles[i][j].x + vilocity[i][j].x) % xi;
6             particles[i][j].y = (particles[i][j].x + vilocity[i][j].y) % yj;
7             if (particles[i][j].x < 0)
8                 particles[i][j].x += xi;
9             if (particles[i][j].y < 0)
10                particles[i][j].y += yj;
11        }
12    }
13 }
14
15 }

```

PSO ():

First, we have **particles**, contain the position (x, y) of our routers, **velocity** contain the speed vector of the position and best particles contain the best position, we initialize all of them randomly (that's the first solution before the update), and calculate the coverage and connectivity after that the fitness with the functions above-mentioned, then extract the best particles.

Second, we will update the velocity and positions of our routers after every iteration with the two function **Update-velocity ()** and **Update-position ()**, after that we will do the same operation as the first step, and extract the global best particle and the final coverage and connectivity of that particles (the final solution).

```

1  void rnp::pso_algo(int nb) {
2      gbest = 0;
3      double fitt = 0.0;
4      nbr_of_part = 20;
5      int cov1, conn1;
6      double best_f = 0;
7      double max_fit = 0.0;
8
9      for (int n = 0; n < nbr_of_part; n++) {
10         for (int a = 0; a < nb; a++) {
11             particles[n][a].x = rand() % xi;
12             particles[n][a].y = rand() % yj;
13
14             best_part[n][a].x = particles[n][a].x;
15             best_part[n][a].y = particles[n][a].y;
16
17             vilocity[n][a].x = rand() % xi;
18             vilocity[n][a].y = rand() % yj;
19         }
20         graphfill(n);
21         cov1 = calc_covrege();
22         fill_tab_r();
23         router_connectivity();
24         client_connectivity();
25         conn1 = max_connectivity();
26         fit_tab[n] = calc_fitness(cov1, conn1);
27
28         if (fit_tab[n] > max_fit) {
29             max_fit = fit_tab[n];
30             bp = n;
31             conn = conn1;
32             cov = cov1;
33         }
34     }

```

```

36 for (int f = 0; f < itrnbr; f++) {
37     update_vilo();
38     update_position();
39     for (int z = 0; z < nbr_of_part; z++) {
40
41         graphfill(z);
42         covl = calc_covrege();
43         fill_tab_r();
44         router_connectivity();
45         client_connectivity();
46         connl = max_connectivity();
47         fitt = calc_fitness(covl, connl);
48
49         if (fitt > fit_tab[z]) {
50             fit_tab[z] = fitt;
51             for (int b = 0; b < nb; b++) {
52                 best_part[z][b].x = partics[z][b].x;
53                 best_part[z][b].y = partics[z][b].y;
54             }
55         }
56         if (fitt > max_fit) {
57             max_fit = fitt;
58             cov = covl;
59             conn = connl;
60             bp = z;
61
62             for (int b = 0; b < nb; b++) {
63                 gbest_tab[b].x = partics[z][b].x;
64                 gbest_tab[b].y = partics[z][b].y;
65             }
66         }
67     }
68 }
69 }
70 }

```

3.2 Design of the energy flow model

in this part, we implement that model In C++, the three-principal's element in this model are: the energy model, the charging model and the discharging model and we implement them all in one function **Energy-flow-model ()**.

Energy-flow-model (): in this function, we implement how the router charge and discharge the energy solar, and how the number of router increment and decrement according to the charging formulas (10), about the discharging model we just assume a simple random one for that.

```
1 void rnp::Energy_Flow_Model() {
2     double init_c = 1000.0;
3     for (int j = 0; j < nbr; j++) {
4         router_fix[j].energy = init_c;
5         router_fix[j].router_state = true;
6     }
7     double k1;
8     int rr1 = nbr;
9     double c;
10    rr2 = rr1;
11    pso_algo(rr1);
12
13    for (int kk = 1; kk < 24; kk++) {
14        for (i = 0; i < nbr; i++) {
15            k1 = fmod(kk, 24.0);
16            c = Cmax * ((-1.0 / 36.0 * (k1 * k1)) + (2.0 / 3.0 * k1) - 3.0);
17            decharge = rand() % 100;
18
19            if (router_fix[i].router_state == false) {
20                decharge = 0;
21            }
22
23            router_fix[i].energy += (c - decharge);
24            if ((router_fix[i].energy >= (0.2 * init_c)) &&
25                (router_fix[i].router_state == false)) {
26                rr2++;
27                router_fix[i].router_state = true;
28            }
29            if (router_fix[i].energy > init_c) {
30                router_fix[i].energy = init_c;
31            }
32        }
33    }
```

```
34
35     if ((router_fix[i].energy <= 0) &&
36         (router_fix[i].router_state == true)) {
37         router_fix[i].energy = 0;
38         router_fix[i].router_state = false;
39         rr2--;
40     }
41
42 }
43
44 if (rr2 != rr1) {
45
46     rr1 = rr2;
47     pso_algo(rr2);
48
49 }
50
51 int il = 0;
52 for (i = 0; i < nbrr; i++) {
53     save_tab[kk].save_r[i].router_state = router_fix[i].router_state;
54     if (router_fix[i].router_state == true) {
55
56         save_tab[kk].save_r[i].x = best_part[bp][il].x;
57         save_tab[kk].save_r[i].y = best_part[bp][il].y;
58         il++;
59     }
60 }
61
62 save_tab[kk].cov_k = cov;
63 save_tab[kk].conn_k = conn;
64 save_tab[kk].fitness_k = calc_fitness(save_tab[kk].cov_k,
65     save_tab[kk].conn_k);
66 }
67
68 }
```

4. Simulation

4.1 Simulation parameters

This three tables present the default parameters of the simulation, the PSO algorithm and Energy Flow Model shown as flow:

Simulation parameters:	
Number of clients	100
Number of routers	15
X	150
Y	150
Radius	20

Table 3.1 Simulation parameters

PSO parameters:	
Number of particles	20
Number of iteration	1000
Lambda	0.5
C	5
r1 & r2	Random (0, 1)
C1 & C2	2.0 & 3.0

Table 3.2 PSO parameters

Energy-Flow-Model parameters:	
Cmax	100
Energy initial of routers	1000.0
Number of Iteration	24

Table 3.3 Energy flow model parameters

The simulation and the analysis are done with four cases:

- Fixing all the parameters and change the number of clients.
- Fixing all the parameters and change the number of routers.
- Fixing all the parameters and change the number of PSO iteration.
- Fixing all the parameters and change the client's topology (the clients position).

The essential elements for evaluate the performance of the algorithm are:

- Fitness.
- Connectivity.
- Coverage.
- Number of routers

And the time interval of the study is the iteration number of the energy flow model, 24 iteration means one day (24h),

Now I will show and describe the interface of our implementation and where and how we do the simulation (Figure 3.1).

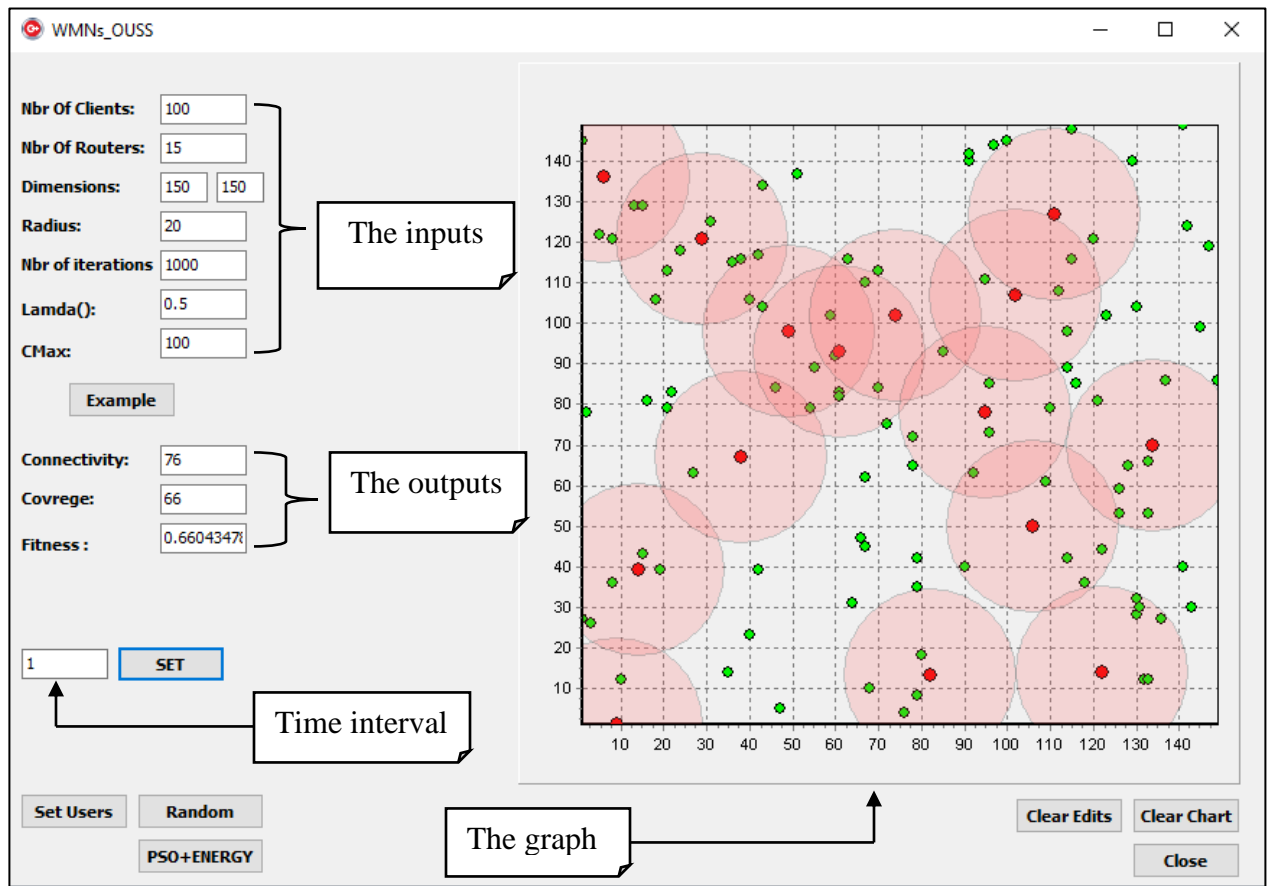


Figure 3.3 Implementation interface (RAD studio XE3)

5. Results

5.1 Case 01 (changing clients number)

In this case we tried to change the clients number to see the change in the algorithm, the interval number of the clients is (50, 100, 150, 200).

- Number of routers variations by contribution of client's number:

In this case at the first all the routers are charged, but after several hours we started to see some of the routers discharged and stop working for example from 5 to 10 there is a vibration and some of the routers stop working, but the algorithm all the time tried to adapted itself when the number of routers changed and tried to give as good performance, and after 3 hours the routers charging again and return to work normally and we return again to the stable state but after 9 hours of working and at night all the routers start dying from 19 to 24 and stop working.

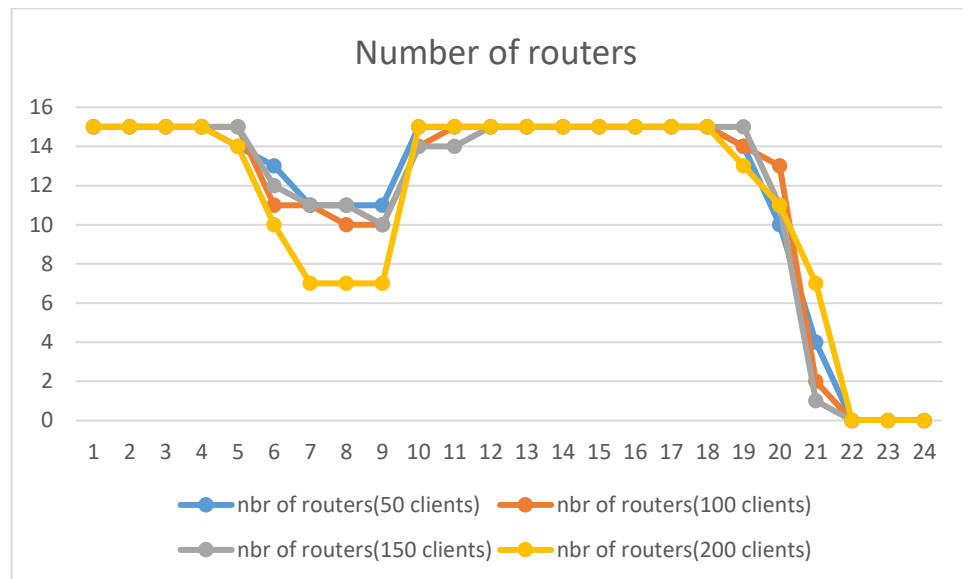


Figure 3.4 number of routers variations by contribution of client's number

- Fitness variations by contribution of client's number:

In this case we must take in consideration the number of routers, because at first when the number of routers is fixed, the value of fitness is stable and good (more than 0.5), but when the number of routers started to change there is a vibration from 5 to 11 and the algorithm also tried to approved itself, just in the case when we had 200 clients we had bad result because the number of routers in that moment is so low and can't cover 200 clients (max router number = 15), and again when all the routers died the value of fitness equal 0 from 22 to 24(at night).



Figure 3.5 fitness variations by contribution of client's number

- Connectivity variations by contribution of client's number:

We see the same here, the number of routers is the basic control element of the results, and when we changed the number of clients and the number of routers is fixed we see good results and good performance specially from 10 to 18 when the routers charged again and when all the routers are died again, the connectivity equal 0.

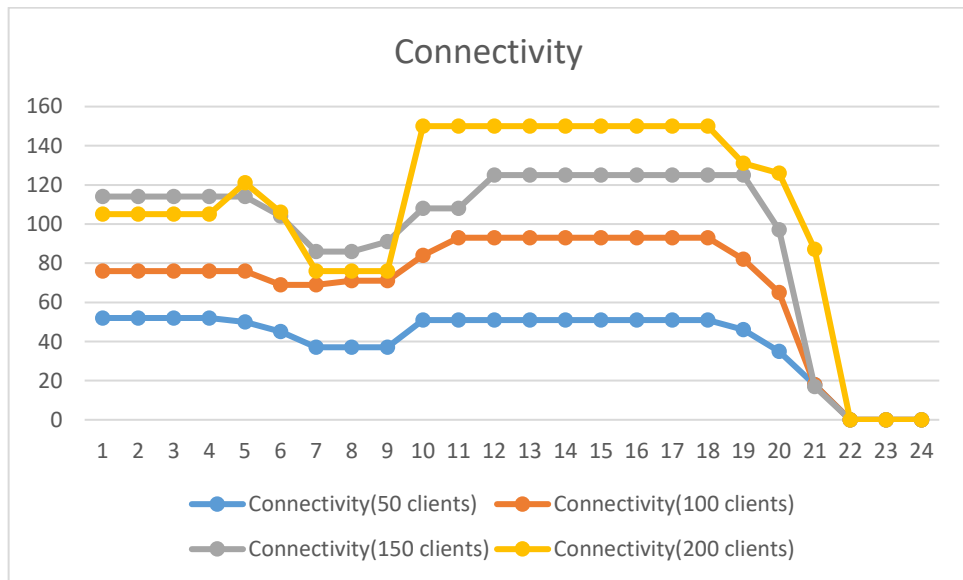


Figure 3.6 connectivity variations by contribution of client's number

- Coverage variations by contribution of client's number:

Like always, we had good results when the number of routers is fixed, and a little vibration when the number of routers is changed of course, and finally when number of routers equal 0, the coverage value equal 0.

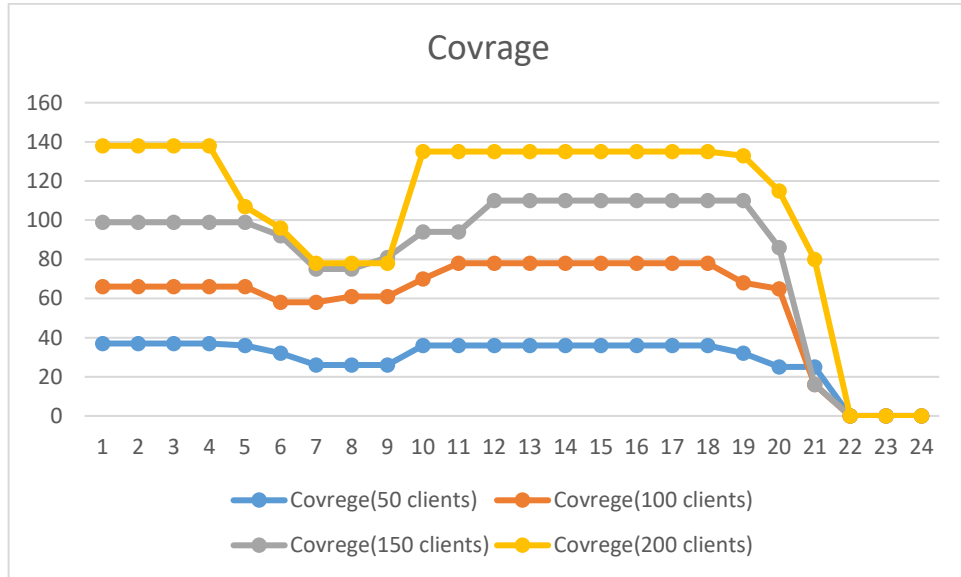


Figure 3.7 coverage variations by contribution of client's number

5.2 Case 02 (changing routers number)

In this case we tried to change the routers number to see the change in the algorithm, the interval number of the routers is (5, 10, 15, 20).

- Number of routers variations by contribution of routers number:

Let's remember here we have 100 clients like a default parameter (table 3.1), and we are trying to change every time the number of routers and like we see in the graph (figure 3.11), the number of routers is changing after several hours from the first start and all the routers died at the end from 22 to 24.

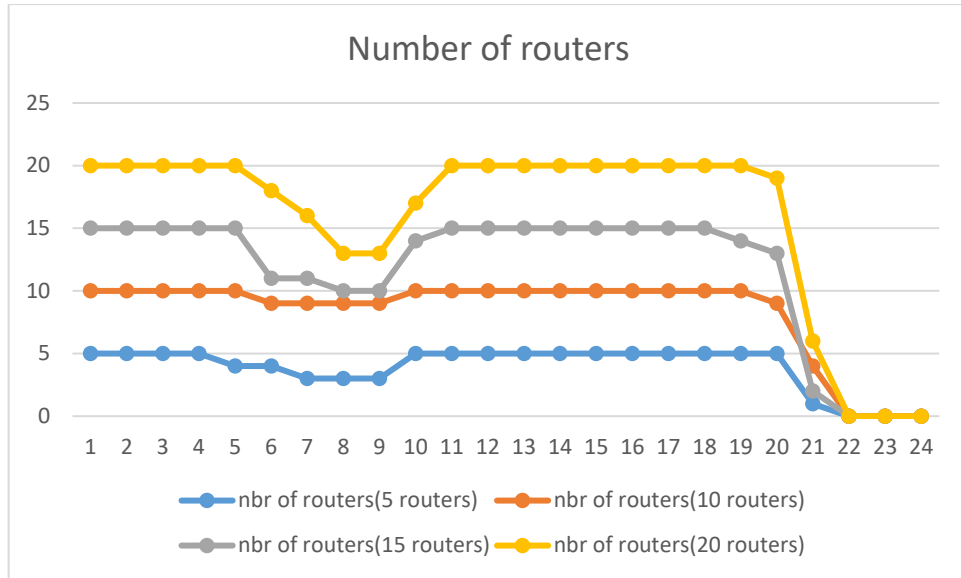


Figure 3.8 number of routers variations by contribution of routers number

- Fitness variations by contribution of routers number:

Here like we see when we had 5 routers we don't get good results because we had 100 clients, but in 10,15 and 20 routers we get a good results and god performance (the value of fitness more than 0.5) from the algorithm also when the number of routers is changed from 5 to 10, in the end when all the routers is died, fitness equal 0.

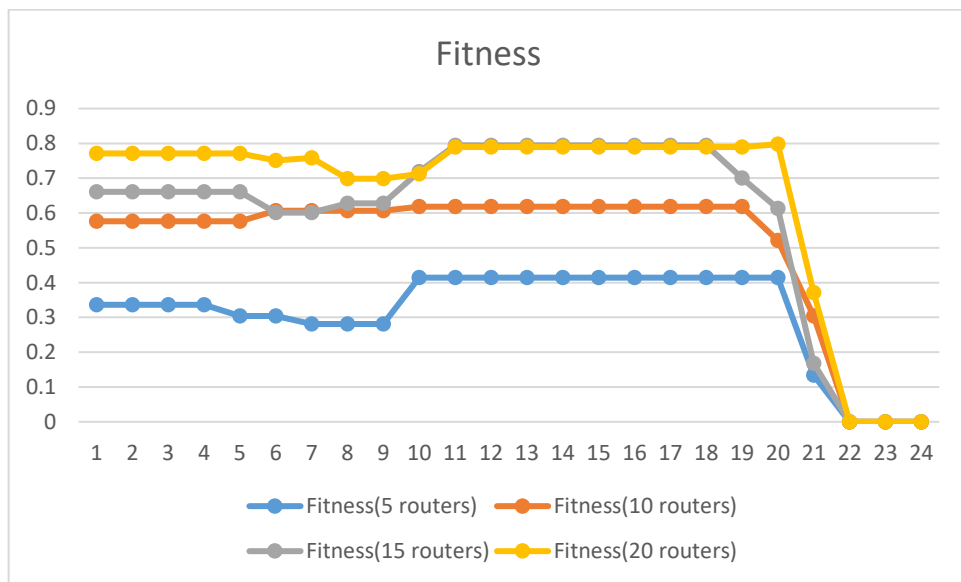


Figure 3.9 fitness variations by contribution of routers number

- Connectivity variations by contribution of routers number:

The same observation shown here, when number of clients is 100 and number of routers more than 5 we had good performance from the algorithm also when the number of routers is decremented, the algorithm adopted itself, and like always when the routers died, the value of connectivity equal 0.

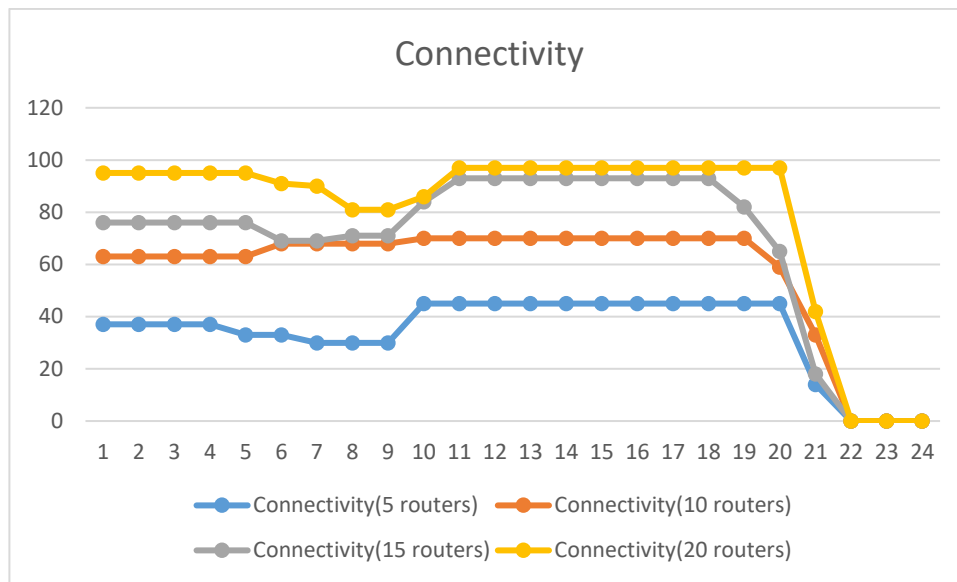


Figure 3.10 connectivity variations by contribution of routers number

- Coverage variations by contribution of routers number:

The same observation shown here, when number of clients is 100 and number of routers more than 5 we had good performance from the algorithm also when the number of routers is decremented, the algorithm adopted itself, and like always when the routers died, the value of coverage equal 0.

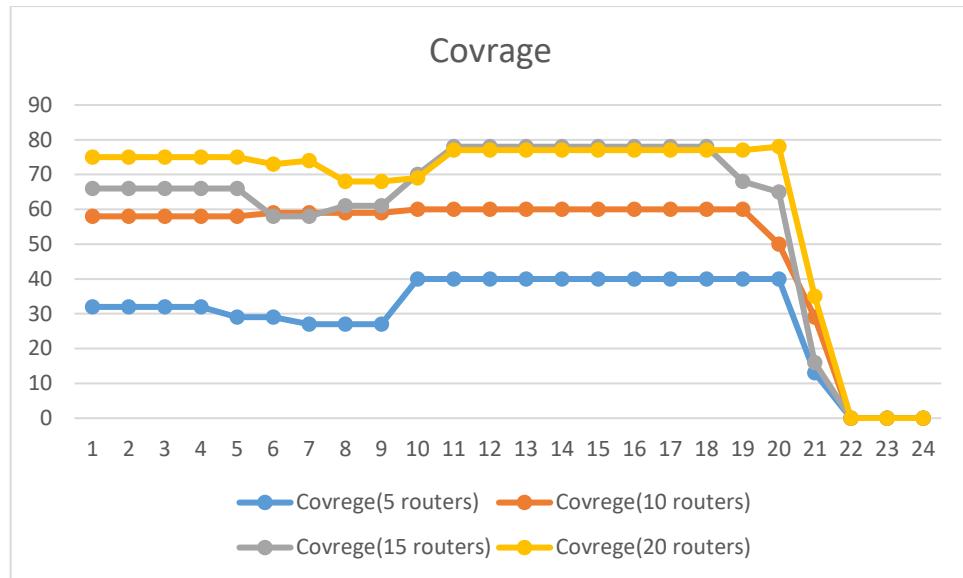


Figure 3.11 coverage variations by contribution of routers number

5.3 Case 03 (changing PSO iteration number)

In this case we tried to change the PSO iteration number to see the change in the algorithm, the interval of iteration number is (100, 500, 1000, 2000).

- Number of routers variation by contribution of PSO iteration number:

Here like we see the same observation, the number of routers decrement after several hours of working, then charged again from 10 to 19, and finally decrement again intel 0 from 22 to 24.

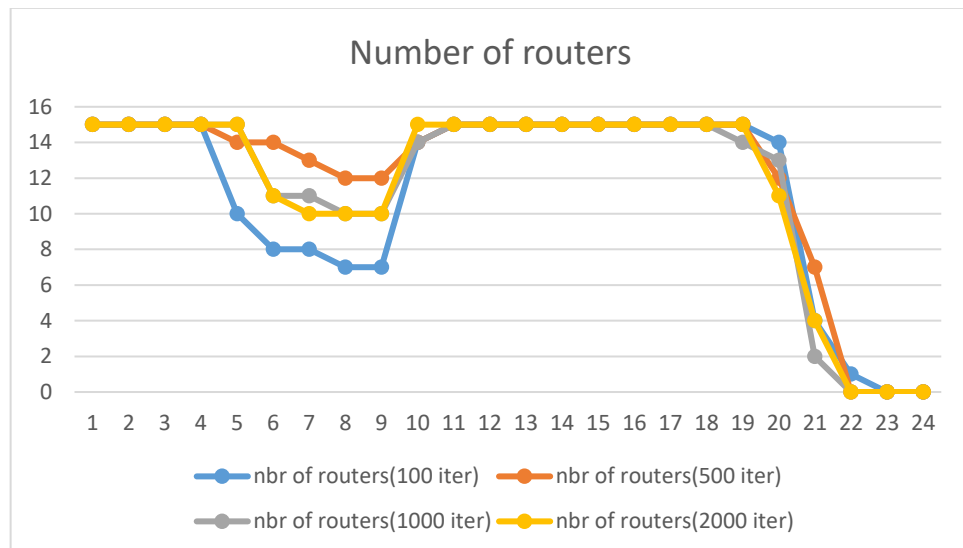


Figure 3.12 number of routers variations by contribution of PSO iteration number

- Fitness variations by contribution of PSO iteration number:

The fitness value is fixed in the first hours with all the PSO iteration number changing, but after the first change of number of routers we started observing that the algorithm give as best performance in 500 iteration and 1000 iterations and give as a good performance in 100 and 2000 iterations, and when the routers died the fitness value equal 0.

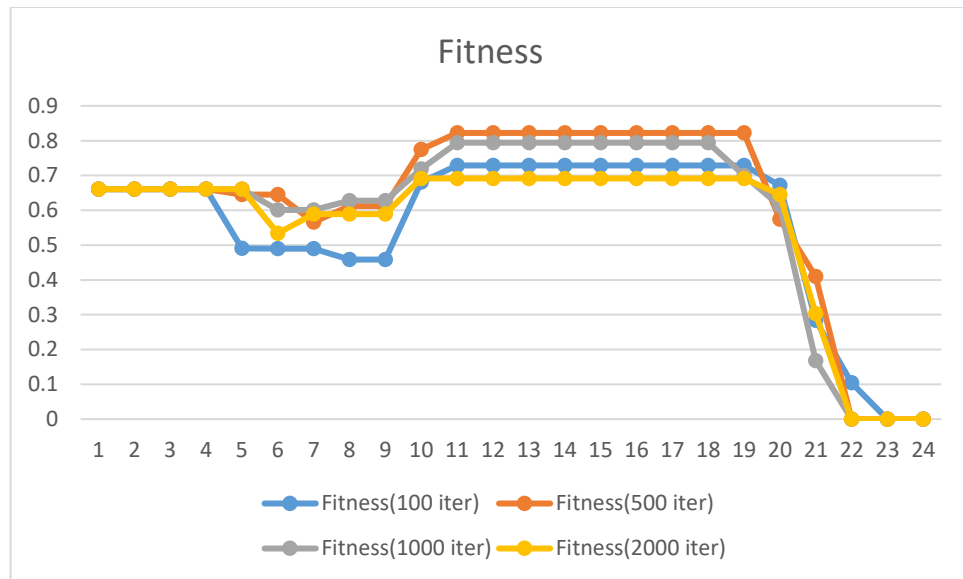


Figure 3.13 fitness variations by contribution of PSO iteration number

- Connectivity variation by contribution of PSO iteration number:

Here when the number of routers fixed and with all the changing of PSO iteration number the algorithm give as the same results at the first hours, then after the first changing of number of routers the algorithm starts work better when PSO iteration number equal 500 and 1000, and give us a good performance when PSO iteration number equal 100 and 2000, and finely when all the routers died the connectivity equal 0.

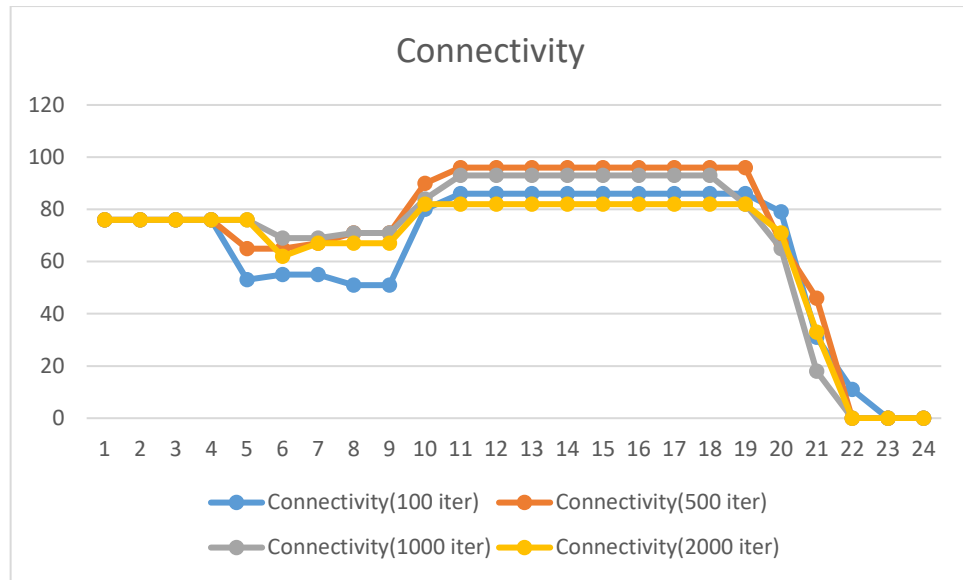


Figure 3.14 connectivity variations by contribution of PSO iteration number

- Coverage variation by contribution of PSO iteration number:

The same observation here too, when the number of routers fixed and with all the changing of PSO iteration number the algorithm give as the same results at the first hours, then after the first changing of number of routers the algorithm starts work better when PSO iteration number equal 500 and 1000, and give us a good performance when PSO iteration number equal 100 and 2000, and finely when all the routers died the coverage value equal 0.

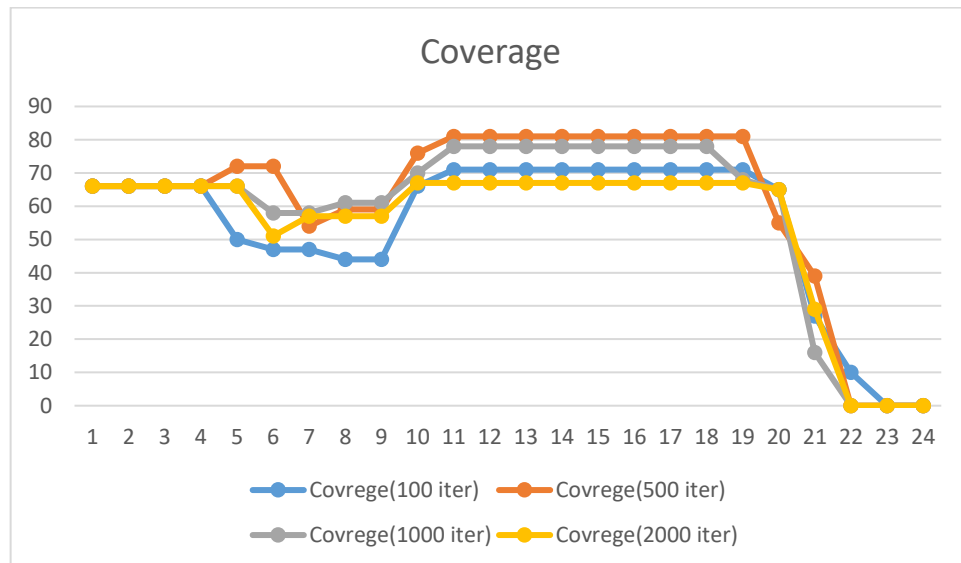


Figure 3.15 coverage variations by contribution of PSO iteration number

5.4 Case 04 (changing the client's positions)

In this case we tried to change the topology (client's positions) to see the change in the algorithm, so we pick six topologies with the same number of clients but different placement of clients, and we fixed all the default parameters.

- The topologies:

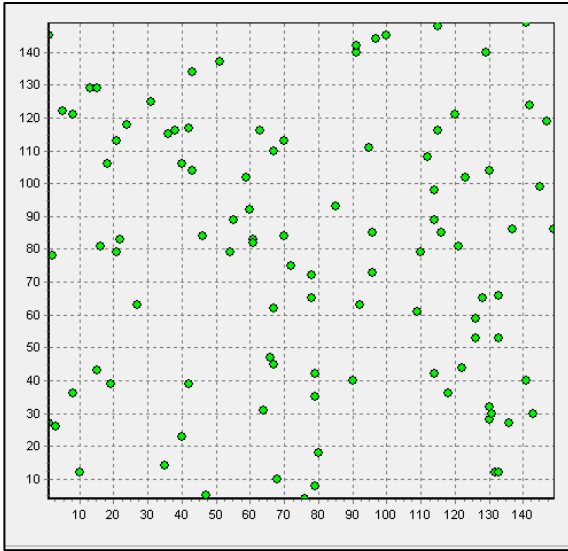


Figure 3.16 Topology 01

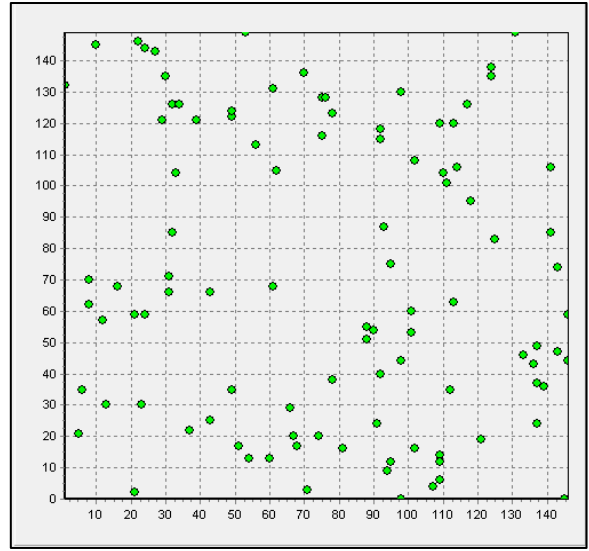


Figure 3.17 Topology 02

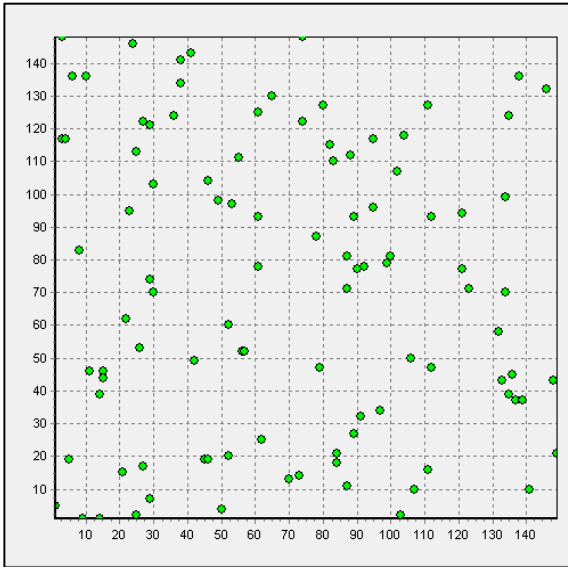


Figure 3.18 Topology 03

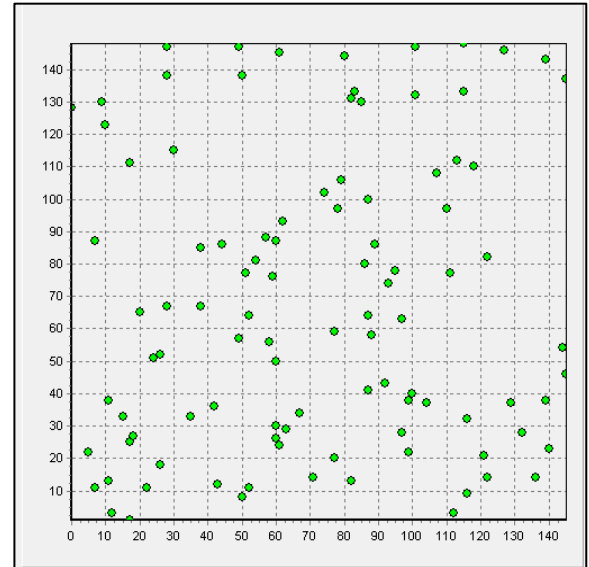


Figure 3.19 Topology 04

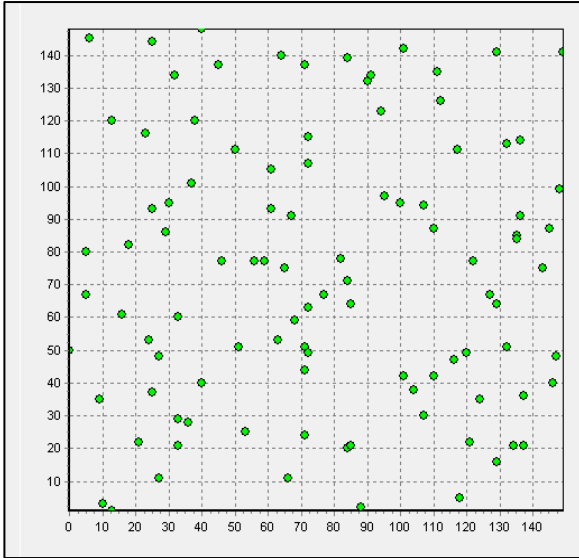


Figure 3.20 Topology 05

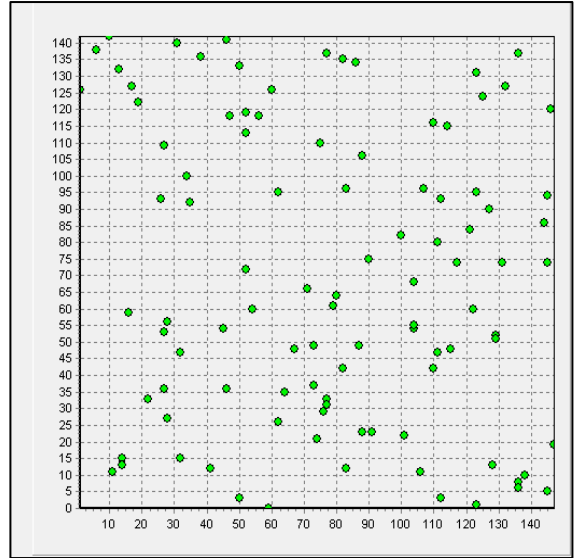


Figure 3.21 Topology 06

After we simulated and analyzed all the topologies each one individually, we calculated the average between them consider the four essential elements **Fitness**, **Connectivity**, **Coverage** and **Number of routers** and we study the variation of them with the default parameters of the algorithms (100 clients, 15 routers,).

- The average of routers number variation in six topologies:

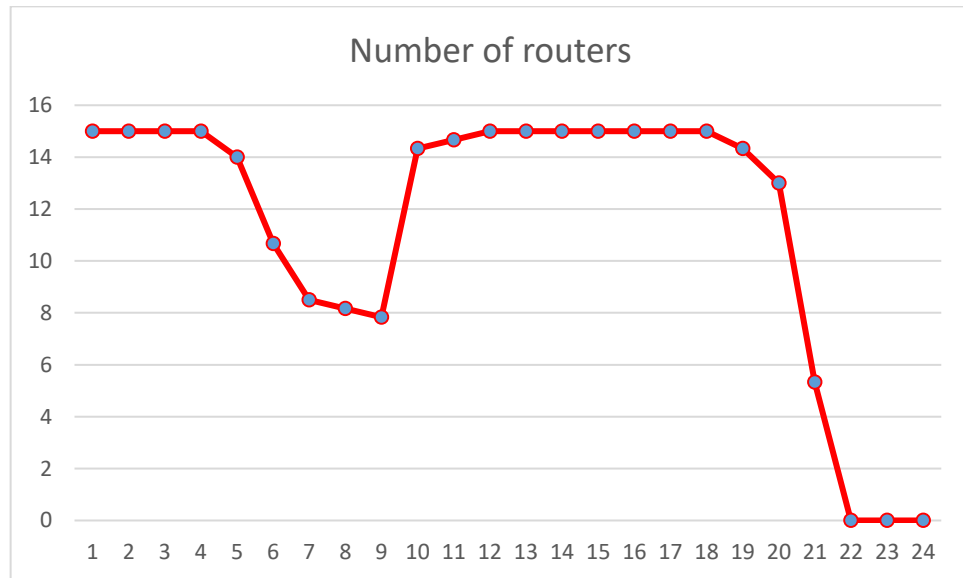


Figure 3.22 The average of routers number variation in six topologies

- The average of fitness variation in six topologies

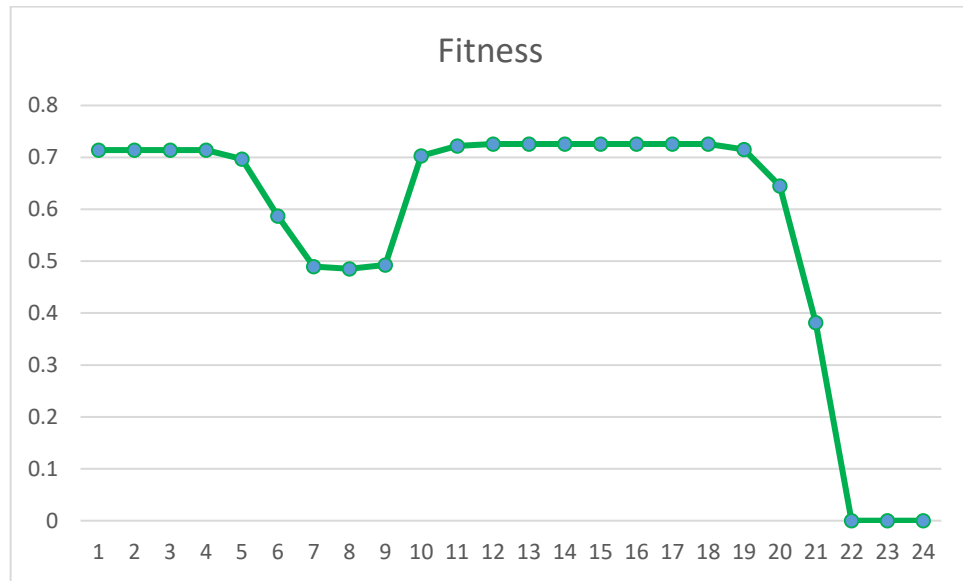


Figure 3.23 The average of fitness variation in six topologies

- The average of connectivity variation in six topologies

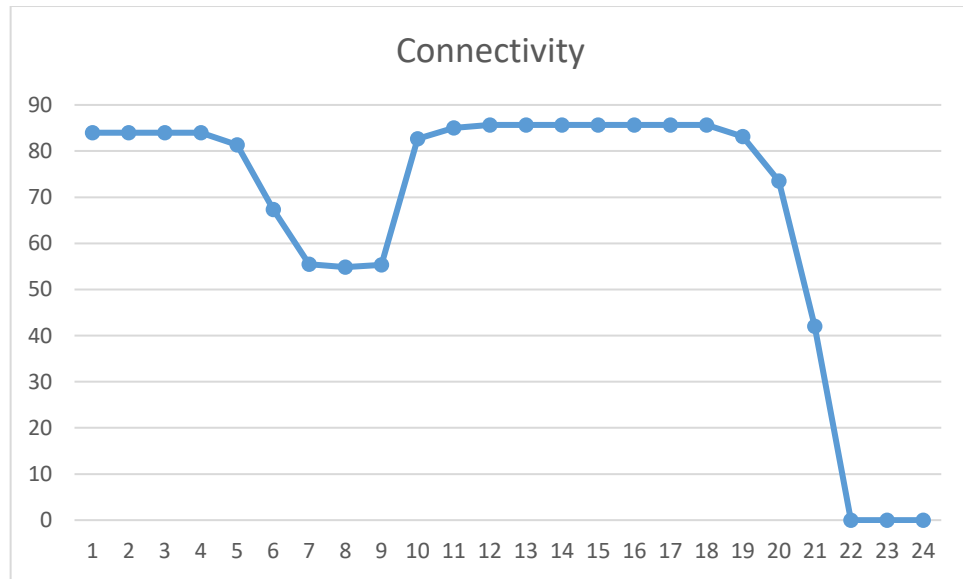


Figure 3.24 The average of connectivity variation in six topologies

- The average of coverage variation in six topologies

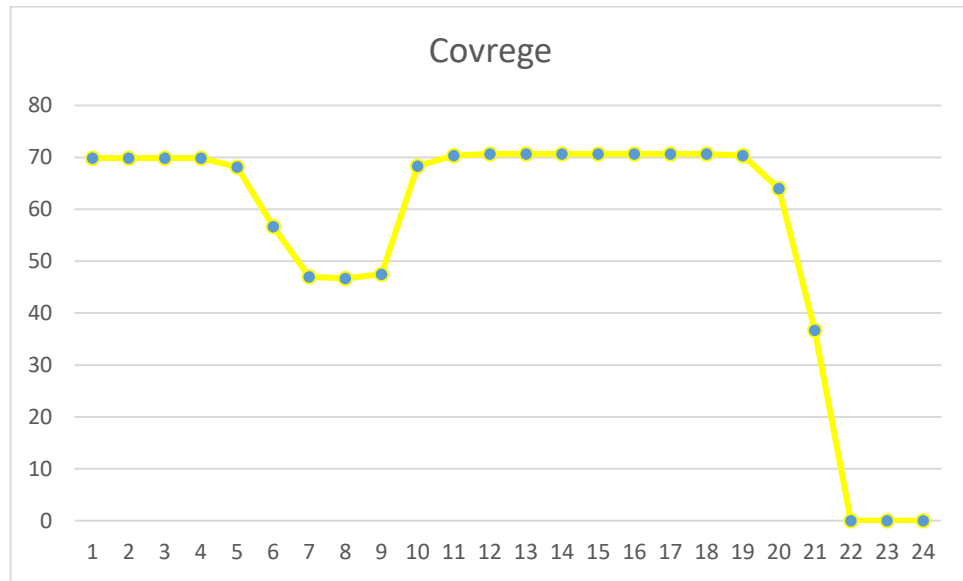


Figure 3.25 The average of coverage variation in six topologies

6. Conclusion

We presented the conception and implementation of algorithms as well as those proposed for study and analysis. The analysis is done according to some parameters that directly affect the performance of routing placement algorithms in WMN, and we will expose the results of the work performed to evaluate the performance of the algorithm.

GENERAL CONCLUSION

GENERAL CONCLUSION

The problem of router nodes placement (RNP) in wireless mesh network (WMNs), is one from the most problems that the researchers work on in the last years.

So, in this project we chose the PSO (Particle Swarm Optimization), to solve that problem but participating with the Energy Flow Model, and we proposed a combination between them.

The analysis of the performance of the algorithms used and proposed was made according to the relevant parameters of the problem of router nodes placement (RNP) in wireless mesh network (WMNs). This allowed us to deepen our investigation of the behavior of our implementation.

As future work, the considered RNP problem can be addressed using other techniques such as genetics algorithm and other metaheuristics, and study the performance of that combination then analyze the results and compare it with our work, we can also implement a real discharging model for the mesh routers (at this time we just implement a random discharging model), and we can also consider as future work the optimal placement off mesh gateways for internet connection.

REFERENCES

- [1] A. Barolli, F. Xhafa, C. Sanchez, M. Takizawa, A study on the effect of mutation in genetic algorithms for mesh router placement problem in wireless mesh networks, in: Proceedings of 5th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS 2011), IEEE Press, 2011, pp. 32-39.
- [2] Chun-Cheng Lin, Y.-L. L. (2012). "Particle Swarm Optimization Applied to Mobile Node Placement in Wireless Mesh Networks."
- [3] F. Xhafa, A. Barolli, C. Sanchez, L. Barolli, A simulated annealing algorithm for router nodes placement problem in wireless mesh networks, *Simulation Modelling Practice and Theory* 19 (10) (2011) 2276-2284.
- [4] F. Xhafa, C. Sanchez, L. Barolli, Genetic algorithms for efficient placement of router nodes in wireless mesh networks, in: Proceedings of 24th IEEE International Conference on Advanced Information Networking and Applications (AINA 2010), IEEE Press, 2010, pp. 465-472.
- [5] J. Kennedy, The particle swarm: Social adaptation of knowledge, in: Proceedings of IEEE International Conference on Evolutionary Computation, IEEE Press, 1997, pp. 303-308.
- [6] Lina, C.-C.: 'Dynamic Router Node Placement in Wireless Mesh Networks: A PSO Approach with Constriction Coefficient and Its Convergence Analysis', 2016
- [7] M. L. S. J. Jun, "The nominal capacity of wireless mesh networks, *IEEE Wireless Communications*," 2003.
- [8] M. L. Sichitiu, "Wireless mesh network: opportunities and challenges," *Dept. of Electrical and Computer Engineering North Carolina State University*.
- [9] M. Clerc, J. Kennedy, The particle swarm - explosion, stability, and convergence in a multidimensional complex space, *IEEE Transactions on Evolutionary Computation* 6 (1) (2002) 58-73.
- [10] Peter Dely, A.K., Nico Bayer: 'OpenFlow for Wireless Mesh Networks', 2011
- [11] R. Eberhart, J. Kennedy, A new optimizer using particles swarm theory, in: Proceedings of 6th International Symposium on Micro Machine and Human Science, 1995, pp. 39-43.
- [12] S. C. L. Krishnamurthy, M. Yarvis, J. Chhabra, C. Ellison, C. Brabenac, E. Tsui, "Meeting the demands of the digital home with high-speed multi-hop wireless networks," *Intel Technology Journal*, 2002.
- [13] S. Tierney, "Mesh Networks," *whitepaper of community_networking.org*.
- [14] T. O. Shinji Sakamoto, Leonard Barolli, Makoto Ikeda, Fatos Xhafa, Kazunori Uchida, "Application of WMN-SA Simulation System for WMN Node Placement in a Realistic Scenario," IEEE 29th International Conference on Advanced Information Networking and Applications, 2015.

REFERENCES

- [15] X. W. b. Ian F. Akyildiz a, *, Weilin Wang b, "Wireless mesh networks: a survey," *ELSEVIER*, p. 43, 2004.
- [16] Xiaoli Huan, B. W., Yijun Moa, Laurence T. Yang b (2014). "Rechargeable router placement based on efficiency and fairness in green wireless mesh networks." Elsevier.
- [17] Computer.howstuffworks, available: <http://computer.howstuffworks.com/how-wireless-mesh-networks-work.htm>, accessed (22/02/2017).
- [18] Research.microsoft, available: <http://research.microsoft.com/mesh/>, accessed (11/04/2017).
- [19] Meshdynamics, available: www.meshdynamics.com, accessed (11/04/2017).
- [20] Meshnetworks, available: <http://www.meshnetworks.com>, accessed (03/02/2017).
- [21] Securedgenetworks, available: <http://www.securedgenetworks.com/blog/3-Key-Benefits-of-Wireless-Mesh-Networks>, accessed (16/02/2017).
- [22] Embarcadero, available: <https://www.embarcadero.com/products/cbuilder>, accessed (09/05/2017).

ملخص

هذا العمل متعلق بمشكل الأماكن المناسبة لأجهزة التوجيه, من أجل الشبكات اللاسلكية الخضراء, والهدف هو العثور على الموقع المثالي لجهاز التوجيه لتوفير أفضل اتصال في الشبكة (العدد الأقصى من أجهزة التوجيه المتصلة) والتغطية (العدد الأقصى للمستعملين المشمولين). تحقيقا لهذه الغاية اخترنا نهج سرب الجسيمات الأمثل ولكن مع تركيبه مع نموذج لتدفق الطاقة بعد الدراسة والتحليل لأداء هذا المزيج مع أربعة عناصر أساسية لتقييم الأداء, وتظهر النتائج التجريبية نوعية التركيبية المقترحة من خلال تحليل الحساسية, وتبين أيضا أن التوليفة المقترحة يمكن أن تتكيف مع التغيرات الطوبولوجية. كلمات مفتاحية: الشبكات اللاسلكية الخضراء, نهج سرب الجسيمات الأمثل, نموذج لتدفق الطاقة.

Abstract

This work considers the router node placement (RNP) for green wireless mesh networks (WMNs). The aim is to find the optimal positions of router nodes to provide the best network connectivity (the maximal number of connected routers) and coverage (maximal number of covered clients). To this end, we chose the Particle Swarm Optimization approach (PSO), but with combination with the Energy flow model. We studied and analyzed the performance of that combination with four essential parameters coverage, connectivity, fitness and number of routers. Experimental results show the quality of the proposed combination through sensitivity analysis, and also show that our proposed combination can adapt to the topology changes.

Keywords: Router node placement (RNP), Green wireless mesh network, particle swarm optimization (PSO), Energy Flow Model.

Résumé

Ce travail considère le placement du noeud du routeur (RNP) pour les réseaux de maille sans fils vert (WMN). L'objectif est de trouver les positions optimales des nœuds du routeur pour fournir la meilleure connectivité réseau (le nombre maximal de routeurs connectés) et la couverture (nombre maximal de clients couverts). À cette fin, nous avons choisi l'approche d'optimisation des embrayages de particules (PSO), mais avec la combinaison du modèle de flux d'énergie. Nous avons étudié et analysé la performance de cette combinaison avec quatre paramètres essentiels la couverture, la connectivité, le fitness et le nombre de routeurs. Les résultats expérimentaux montrent la qualité de la combinaison proposée grâce à l'analyse de sensibilité, et montrent également que notre combinaison proposée peut s'adapter aux changements de topologie.

Mots-clés: Emplacement du noeud du routeur (RNP), réseau de maillé sans fil (WMN), optimisation des embrayages de particules (PSO), Energy Flow Model.