Fukuoka Institute of Technology

Graduate School of Engineering

Implementation of Simulation Systems and Testbed for WMNs: Simulation and Experimental Results

by

Tetsuya ODA

Adviser: Prof. Leonard Barolli

March 2016

Contents

List of Figures

List of Tables

Acknowledgement

Working three years to complete my Ph.D. course was not an easy task. The help of many professors and colleagues was needed. First, I would like to thank my advisor Professor Leonard Barolli, for his continuous efforts and support during these six years. He was always ready to listen and to advice my ideas. He taught me how to be practical and persistent in my research without giving up.

I would like to give special thanks to Professor Shigeto Nishida, Dr. Makoto Ikeda and Professor Shijie Zhu as members of inspection committee. They gave me very good advices and suggestion to improve the quality of my thesis.

I thank Professor Kazunori Uchida and Professor Jiro Iwashige, for their comments and support. I also thank Professor Fatos Xhafa for his long talks and innovative ideas in my research field. A special thank goes to Professor Makoto Takizawa and Professor Tomoya Enokido, who always supported me spiritually and gave a lot of feedback in my presentations. I would like to thank also Dr. Keita Matsuo, Dr. Tao Yang, Dr. Elis Kulla, Dr. Evjola Spaho and Dr. Masahiro Hiyama for their kind support, advices and friendship.

My research in Fukuoka Institute of Technology was also inspired by all members of INA laboratory. I thank all students of INA Laboratory studying for many hours in the laboratory, inspiring me to work harder.

Finally I would like to thank my parents, Toyofumi and Chiemi, my brother Kenta, for their spiritual support. Therefore, I want to dedicate this thesis to my beloved family. Their educational support is priceless.

Abstract

A Wireless Mesh Network (WMN) can be defined as a collection of mobile nodes, which form a highly resource constrained network and a dynamic topology. Because of the dynamic topology, routing procedures and protocols are a key field of testing and research. In a research environment, research tools are required to test, verify and identify problems of an algorithm or protocol. These tools are classified in three major techniques: simulators, emulators and real-world testbeds. In most of research in WMNs, their performance is evaluated in both quantitative and qualitative aspects. Throughput performance, routing efficiency, energy consumption are some of the key issues that are addressed frequently on WMNs. The WMN technology has to ensure a certain degree of security and scalability and provide the infrastructure for distributed collaborative computing.

In this thesis, we design and implement simulation systems and a testbed in order to analyse the performance of different routing protocols, architectures, OSs, queue management algorithms and mobility models. From the evaluation results, we found that the mobility of nodes brings oscillations in performance and route instabilities. We found that multiple flows traffic decreases the performance of the network.

The contributions of our work are: 1) implementation and evaluation of a different meta-heuristics methods for mesh router node placement; 2) implementation of a simulation tool for WMNs using NS3; 3) implementation and evaluation of a Raspberry Pi based WMN testbed; 4) application of WMN testbed in real environments, considering different scenarios; 5) evaluation of WMN testbed in different OSs; 6) evaluation of WMN testbed considering distributed concurrent processing; 7) our research work give insights about future developments and integration of WMNs as an important technology of wireless communication.

The outline of the thesis is as follows. In Chapter 1 is shown the background and the motivation of the thesis. In Chapter 2, we introduce general aspects of wireless networks. We discuss wireless architectures and wireless technologies giving advantages and disadvantages for each of them. We present WMNs in Chapter 3. We discuss problems of WMNs and describe the routing protocols and their features. In Chapter 4, we introduce general aspects of meta-heuristics methods. We present Genetic Algorithms (Gas), Tabu Search (TS), Hill Climbing (HC) and Simulated Annealing (SA). We show in details the GA and its operators in Chapter 5. The simulation systems are presented in Chapter 6. We give details of radio propagation models, mobility models and other parameters. Then, we show different scenarios and the traffic data that we used for simulations. In Chapter 7, we introduce the implemented testbed. We give details of technical and environment settings. Then we show the experimental scenarios. In Chapter 8 and Chapter 9, we discuss the results of the simulations and experiments, respectively. Chapter 10 concludes the thesis. We present some concluding remarks and future work.

Chapter 1

Introduction

1.1 Background

Wireless Mesh Networks (WMNs) can be seen as a special type of wireless ad-hoc networks. WMNs are based on mesh topology, in which every node (representing a server) is connected through wireless links to one or more nodes, enabling thus the information transmission in more than one path. The path redundancy is a robust feature of mesh topology. Compared to other topologies, mesh topology does not need a central node, allowing networks based on it to be self-healing. These characteristics of networks with mesh topology make them very reliable and robust networks to potential server node failures.

There are a number of application scenarios for which the use of WMNs is a very good alternative to offer connectivity at a low cost. It should also mentioned that there are applications of WMNs which are not supported directly by other types of wireless networks such as cellular networks, ad hoc networks, wireless sensor networks and standard IEEE 802.11 networks. There are many applications of WMNs in Neighboring Community Networks, Corporative Networks, Metropolitan Area Networks, Transportation Systems, Automatic Control Buildings, Medical and Health Systems, Surveillance and so on.

There are a lot of works done on testbed for WMN and Mobile Ad-hoc Network (MANET). In [1], the authors analyze the performance of an outdoor ad-hoc network, but their study is limited to reactive protocols such as Ad hoc On Demand Distance Vector (AODV) and Dynamic Source Routing (DSR). The authors of [2] perform outdoor experiments of non standard pro-active protocols. Other ad-hoc experiments are limited to identify MAC problems, by providing insights on the one-hop MAC dynamics as shown in [3].

In [4], the authors present an experimental comparison of OLSR using the standard hysteresis routing metric and the Expected Transmission Count (ETX) metric in a 7 by 7 grid of closely spaced Wi-Fi nodes to obtain more realistic results. The throughput results are similar to our previous work and are effected by hop distance [5]. In [6], the authors did not care about the routing protocol. In [7], the disadvantage of using hysteresis routing metric is presented through simulation and indoor measurements.

Also, there are some other research works done for WMNs. In [8], the authors present the architecture, security, and management of SwanMesh. The test results show that the developed network maintained a stable bandwidth after multiple hops. In [9], the authors present different experiments to measure the link and end-to-end delays over the WMN test-bed. The measured link delays are used to construct an empirical histogram. Our experiments reveal that irrespective of the number of hops along the paths and type of traffic crossing the link, the empirical histograms almost have the same general shapes.

In [10], the authors discuss how WMNs can be practically deployed to support wireless multihop communications in a campus - wide area. They have deployed a real WMN at the campus and analyzed the performance for multihop heterogeneous traffic. In another work [11], the authors present COLSR (Cognitive Optimized Link State Routing) in Wireless Mesh Network, which is an extended version of OLSR Protocol. They discuss the generation, reputed-trust mechanism and re-routing for avoiding congestion in WMNs.

1.2 Research Goal

In WMNs, the mesh routers provide network connectivity services to mesh client nodes. The good performance and operability of WMNs largely depends on placement of mesh routers nodes in the geographical deployment area to achieve network connectivity, stability and client coverage.

We considered the version of the mesh router nodes placement problem in which we are given a grid area where to deploy a number of mesh router nodes and a number of mesh client nodes of fixed positions (of an arbitrary distribution) in the grid area [12–14]. Our goal is to implement mesh router nodes placement system that is based on Intelligent Algorithms to find an optimal location assignment for mesh routers in the grid area in order to maximize the network connectivity.

We use the topology generated by WMN-GA system and evaluate by simulations the performance for different distributions of mesh clients considering two architectures of WMNs by sending multiple Constant Bit Rate (CBR) flows in the network. As evaluation metrics we considered throughput, delay, jitter, fairness index and energy.

Finally, we implement a WMN testbed and investigate the performance of OLSR and parallel distributed processing in different scenarios. For evaluation, we considered hop count, delay, jitter and processing time metrics.

1.3 Research Interests and Related Work

Until now, many researchers performed valuable research in the area of multi-hop wireless networks by computer simulations and experiments [15]. Most of them are focused on throughput improvement and they do not consider mobility [16].

WMNs are attracting a lot of attention from wireless research. Node placement problems have been investigated for a long time in the optimization field due to numerous applications in location science (facility location, logistics, services, etc.).

The main issue of WMNs is to achieve network connectivity and stability as well as QoS in terms of user coverage. Several heuristic approaches are found in the literature for node placement problems in WMNs [17–20]. As node placement problems are known to be computationally hard

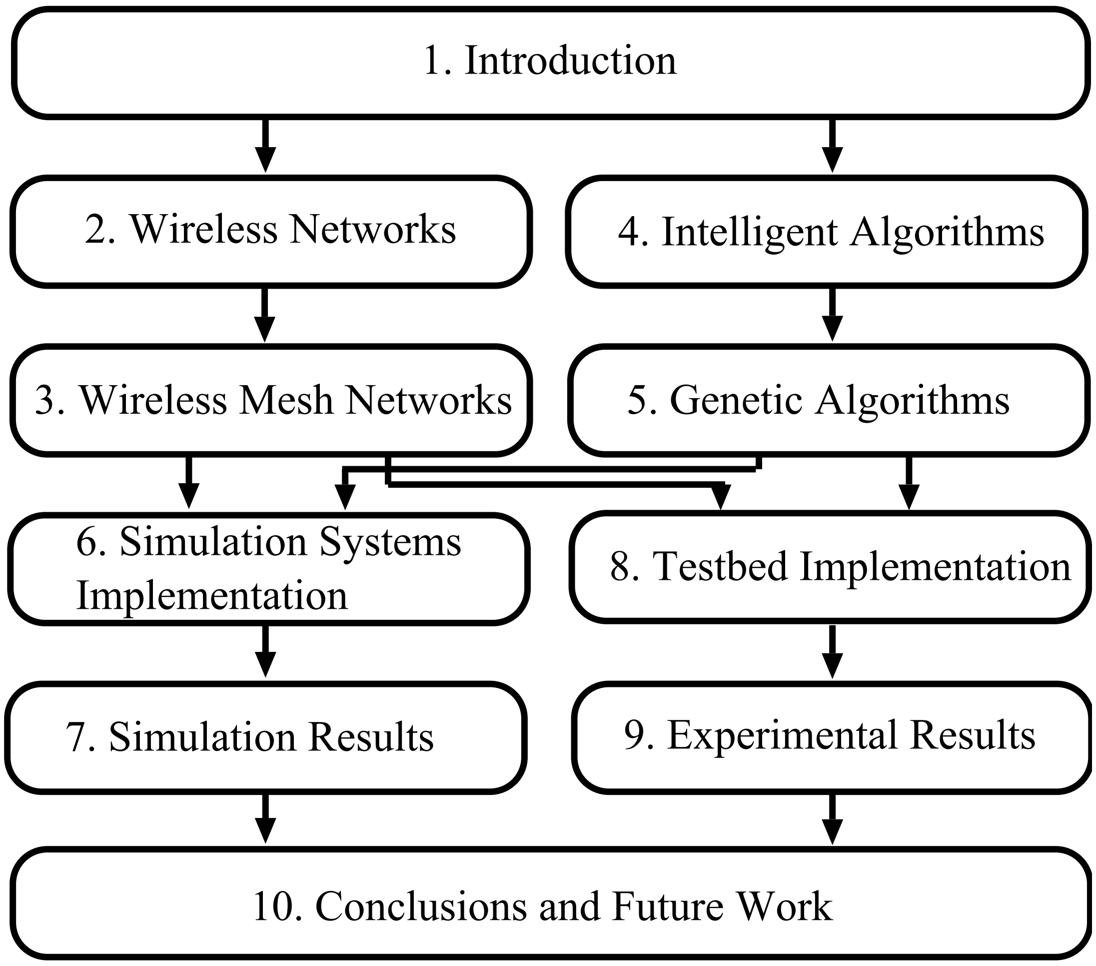


Fig. 1.1 The structure of the thesis.

to solve for most of the formulations [21, 22], GAs have been recently investigated as effective resolution methods. However, GAs require the user to provide values for a number of parameters and a set of genetic operators to achieve the best GA performance for the problem [23–28].

1.4 Thesis Contribution

In this thesis, we contribute in the research field as following:

- Implementation and evaluation of a WMN testbed;
- Implementation and evaluation of a mesh router placement optimization methods using different meta-heuristics for WMNs;
- Implementation of a simulation tool for WMNs using ns3;
- Application of WMN testbed in real environments, considering different scenarios;
- Evaluation of different WMN MAC protocols, routing protocols and transport protocols in different scenarios using ns3;
- Evaluation of distributed concurrent processing in different scenarios using testbed;
- Give insights about future developments and integration of WMN as an important technology of wireless communications.

1.5 The Structure of The Thesis

The outline of the thesis is as follows. In Chapter 1 is shown the background and the motivation of the thesis. In Chapter 2, we introduce general aspects of wireless networks. We discuss wireless architectures and wireless technologies giving advantages and disadvantages of each. We give insights of WMNs in Chapter 3. We discuss issues and problems of WMNs, and describe routing protocols and their properties. We describe Intelligent Algorithm in Chapter 4. We discuss issues and problems of Metaheuristic methods, and describe methods and their properties. We give insights of GAs in Chapter 5. We present a simple GA and some applications of GAs. The simulation system is presented in Chapter 6. We give details on radio propagation models, mobility models and other parameters used in simulations. In Chapter 8, we present the design and implementation of our testbed. We give details on technical settings and environment assumptions. The scenarios and the way of implementation are described in details. Later we show the moving scenarios and the traffic data that we used during experiments. In Chapter 7 and Chapter 9, we discuss the results of our experiments and simulations, respectively. Chapter 10 concludes the thesis, giving an insight of learnt lessons and future works in this field.

Chapter 2

Wireless Networks

2.1 Wireless Network Introduction

Wireless networks have evolved with great speed during the last decades and it seems like in the future this speed will keep going. A telecommunication network, in which no wires are used to create the interconnections, is refered to as Wireless Network. Since now many technologies and standarts are developed using wireless communications.

In this chapter, we are describing some of basic concepts of wireless networks and some of their applications

2.2 Wireless Architecture

Wireless networks can be built using two network architectures: infrastracture architecture and ad hoc architecture. A simple example to make a comparison between the two is shown in Figure 2.1.

2.2.1 Infrastracture Architecture

In general, the wireless networks are used to extend wired networks in areas where it was almost impossible to install wires. Many wireless units connect wirelessly to one unit, which is wired to the wide network. This unit has a very critical role in keeping the network connected. We called this node an Access Point (AP) or Base Station (BS), meaning that each node can have access to the network only by accessing this central node. Even though two nodes may be near each other, they both need to be connected with the AP.

APs usually transmit with more power than other units, to ensure a given coverage area. They are also responsible for coordinating access to simultaneous transmission from all units in the coverage area. It assigns transmission channels to the units. These channels can be *frequencies* (Frequency Division Multiple Access), *time slots* (Time Division Multiple Access) *orthogonal codes* (Code Division Multiple Access) or a mixture of the above mentioned methods.

In the infrastructure architecture, wireless transmissions occur only in the last hop of communication, where all units in the coverage area share the bandwidth of the wireless channel.

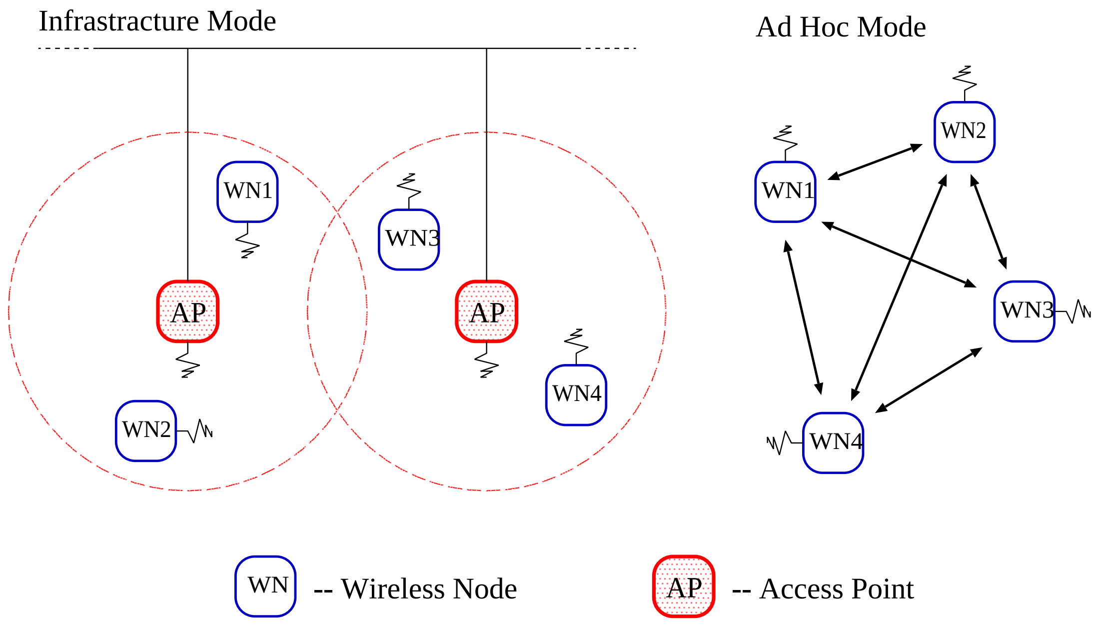


Fig. 2.1 Ad-Hoc and infrastructure mode.

2.2.2 Ad Hoc Architecture

In ad hoc architecture, units create a temporary and dynamic network, without any aid from wired networks. All units are independent of each other and can cooperate to maintain network connectivity. Ad hoc architecture is characterized by a random and dynamic topology and by multi-hop communication. No wired support is needed, so these networks can exist by their own.

Unlike infrastructure architecture, in ad hoc architecture units should provide multi-hop transmission by being able to forward packets for other units. This makes the units operate in both end device mode as well as router mode. The MANET (Mobile Ad hoc NETwork) work group of IETF is formed to support Ad Hoc issues and improvements.

In Fig. 2.1, in infrastructure architecture of a 802.11b, even though their transmission range cover each other geographical position, nodes WN1 and WN3 are part of different infrastructures, separated by APs. Thus, they can communicate only through their respective access points. While, in Ad Hoc mode, each node can communicate with every other node which is inside its transmission range. This means that Ad Hoc networks do not need the aid of any central device. By avoiding the centralized administration of the network in ad hoc infrastracture, the “one point of failure” is also avoided.

2.3 Wireless vs. Wired

The evolution from wired networks to wireless networks has lead to some issues due to some problem-posing phenomena. These phenomena, should be addressed correctly, when deploying the communication algorithm. Three of the most problematic phenomena are discussed in following.

2.3.1 Collision

When two units in the same network try to communicate simultaneously in the same channel, collision occurs. In wired networks, switching devices are used to allow units to take turns sending packets, while in wireless networks communication is done through an antenna, which usually is omni-directional. This makes it more difficult to control the collision issue, because a single antenna can be used only for receiving or only for transmitting in a certain given time. Thus, if two units try to transmit messages to the same third party unit, this unit will not understand neither of the messages.

2.3.2 Unidirectional Links

In wired networks a link is always available from both sides communicating, being a two-way link. While, in wireless networks this situation is not always true. The units may have different antenna characteristics, the receiving and transmitting circuits may provide different power levels, and there may be interference from other sources. These conditions cause some links to be unidirectional, being one-way links. This means a unit should be aware of the availability of both direction links, before transmitting any signal.

2.3.3 Asymmetric Links

Another phenomena which may occur due to radio irregularities, is the asymmetric links. An asymmetric link has different network parameters for downstreaming and upstreaming (like an ADSL line). These links may cause problems if not taken into account by the communicating units.

2.4 The Wireless Channel

Communication of nodes in Ad Hoc networks are done through wireless trancievers. Thus, the wireless channel is an important block of any model used to describe a wireless system. A more detailed description can be found in [29].

A radio channel between a transmitter unit u and a receiver unit v is established if and only if the power of the radio signal received by node v is above the sensitivity threshold. Theoretically, there exists a direct wireless link between a transmitter unit u and a reciever unit v if $P_r \geq \beta$, where P_r is the power of the signal received by v, and β denotes the sensitivity threshold. The exact value of β depends on the features of the wireless transceiver and on the communication data rate. If we increase the data rate for a given radio, the value of β will be increased. The received power P_r is affected by the power P_t used by unit u to transmit, and on the path loss, which models the wireless signal degradation with distance. Denoting with $PL(u, v)$ the path loss between units u and v, we can write:

$$P_r = \frac{P_t}{PL(u, v)}. \tag{2.1}$$

Modeling path loss is one of the most difficult tasks of the wireless system designer. The mechanisms that affect the radio signal propagation can be classified into three major categories: reflection, diffraction and scattering.

- When electromagnetic waves hit the surface of a large object (earth surface, large buildings etc.), compared to the wavelength of the propagating signal, *reflection* occurs.
- *Diffraction* occurs when there are objects with sharp edges lying on the radio path between the transmitter and the receiver.
- Sometimes several small objects, (as compared to the signal wavelength) may happen to be in between the transmitter and the receiver of the radio signal. In this case *scattering* occurs.

Taking into account these mechanisms, makes radio wave propagation an extremely complex phenomenon, which is heavily influenced by environmental factors. We will explain shortly three widely-used path loss models.

2.4.1 The Free Space Propagation Model

The free space propagation model is used to describe radio signal propagation when between the transmitter and the receiver there is no obstructions, Line-Of-Sight (LOS). Denoting with $P_r(d)$ the power of the radio signal received by a node located at distance d from the transmitter, we have:

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L}, \tag{2.2}$$

where $\mathbf{G_t}$ is the transmitter antenna gain, $\mathbf{G_r}$ is the receiver antenna gain, $\mathbf{L}$ is the system loss factor not related to propagation and λ is the wavelength in meters.

By simplifying Equation (2.2) and denoting C_f the constants, which depends only on tranciever characteristics, a more simple equation derives:

$$P_r(d) = C_f \frac{P_t}{d^2}. \tag{2.3}$$

Equation (2.3) shows the decreasing of the received power is proportional to the square of the distance d that separates the transmitter and the receiver. Combining Equation (2.3) with the sensitivity threshold, we can claim that the transmitted message can be correctly received if and only if $d \leq \sqrt{C_f P_t}$. In other words, the coverage area of a wireless node transmitting at power P_t is a disk of radius $\sqrt{C_f P_t}$ centered at the transmitter.

2.4.2 The Two-Ray Ground Model

In most of the cases, the signal sent from the transmitter to the receiver follows multiple radio paths. For this reason, the free space propagation model is not always correct. A more accurate approach to modelling the propagation of the radio signal is the two-ray ground model, which considers

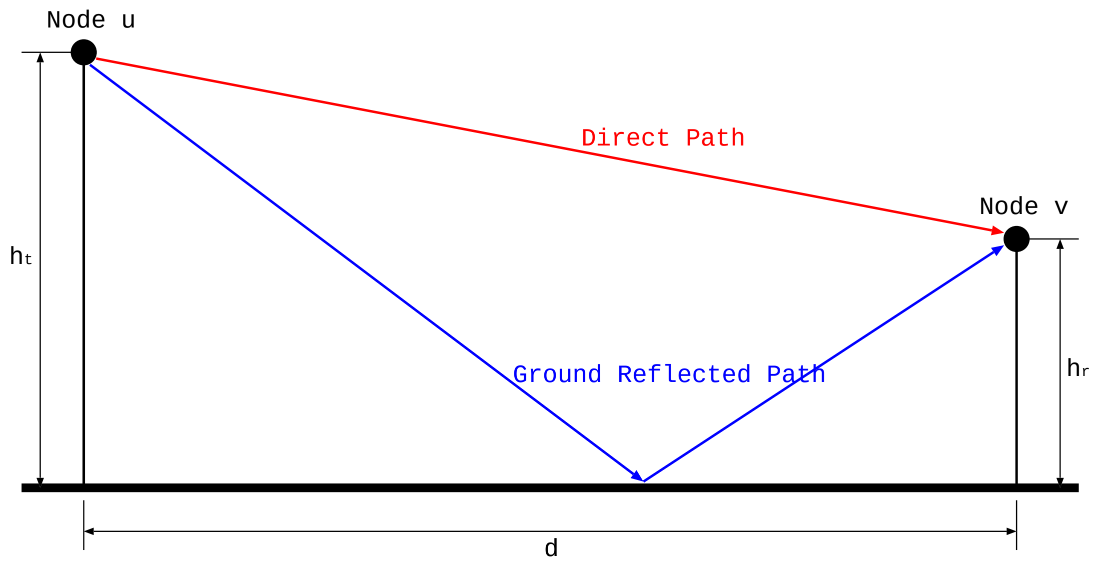


Fig. 2.2 Two-ray Ground Propagation Model.

two propagation paths: the direct path and a ground reflected propagation path*1 between the transmitter and the receiver (see Fig. 2.2).

The radio signal sent by node u reaches node v through the direct path, and through a ground reflected path. The received power at distance d, in the two-ray ground propagation model is given by the following formula:

$$P_r(d) = P_t G_t G_r \frac{h_t^2 h_r^2}{d^4}, \tag{2.4}$$

where $\mathbf{h_t}$ is the transmitter antenna height and $\mathbf{h_r}$ is the receiver antenna height.

If the sender and the receiver are relatively far from each other ($d \gg \sqrt{h_t h_r}$), and denoting C_t the constants, which depends only on tranciever characteristics, the following simplified formula can be written:

$$P_r(d) = C_t \frac{P_t}{d^4}. \tag{2.5}$$

In Equation (2.5), it can be easily noticed that the decreasing of radio signal power is in proportional to the distance between nodes raised to the fourth power, instead of to the square, in the Free Space model. Combining Equation (2.5) with the sensitivity threshold, we can claim that the transmitted message can be correctly received if and only if $d \leq \sqrt[4]{C_t P_t}$. In other words, the coverage area of a wireless node transmitting at power P_t is a disk of radius $\sqrt[4]{C_t P_t}$ centered at the transmitter.

2.4.3 The Shadowing Model

The shadowing (log-distance) model has been derived combining analytical and empirical methods. Empirical methods are based on field measurements and statistical calculation on the experimental data. This model, which can be seen as a mixture of both the free space and the two-ray ground models, indicates that the average shadowing path loss is proportional to the separation distance d

*1 This is not to be misinterpreted as Multipath Fading.

Table 2.1 Values of the path loss exponent in different environments.

Environment	α
Open Space	2
Urban Area	$2.7 - 3.5$
Indoor LOS	$1.6 - 1.8$
Indoor no LOS	$4 - 6$

raised to a certain exponent α, which is called *the path loss exponent*, or *distance-power gradient*.

$$P_r(d) \approx \frac{P_t}{d^\alpha}. \tag{2.6}$$

From Equation (2.6), we can claim that the radio coverage region in this model is a disk of radius proportional to $\sqrt[\alpha]{P_t}$ centered at the transmitting node. The value of α depends on the environmental conditions, and it has been experimentally evaluated in many scenarios. The author of [29], provides us with some values of α. Tab. 2.1 summarizes some of these values.

2.5 Overview of DCF and EDCA Protocols

In our study, we consider two distributed access methods: DCF from legacy 802.11 [30] and EDCA from 802.11e [31]. The centralised access methods, Point Coordination Function (PCF) [30] and Hybrid Controlled Channel Access (HCCA) [31] are not considered as they are rarely implemented in hardware devices [32].

2.5.1 Distributed Coordination Function (DCF)

DCF is a random access scheme based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme. A legacy DCF station with a packet to send will first sense the medium for activity. If the channel is idle for a Distributed Inter-Frame Space (DIFS), the station will attempt to transmit after a random back-off period. This period is referred as the Contention Window (CW). The value for the CW is chosen randomly from a range $[0, 2^n - 1]$, i.e.

$$CW_{min} \leq CW \leq CW_{max} \tag{2.7}$$

where n is PHY dependent. Initially, CW is set to the minimum number of slot times CW_{min}, which is defined per PHY in microseconds [30]. The randomly chosen CW value, referred as the back-off counter, is decreased each slot time if the medium remains idle. If during any period the medium becomes busy, the back-off counter is paused and resumed only when the medium becomes idle. On reaching zero, the station transmits the packet in the physical channel and awaits an acknowledgment (ACK). The transmitting station then performs a post back-off, where the back-off procedure is repeated once more. This is to allow other stations to gain access to the medium during heavy contention.

If the ACK is not received within a Short Inter-Frame Space (SIFS), it assumes that the frame was lost due to collision or being damaged. The CW value is then increased exponentially and the back-off begins once again for retransmission. This is referred as the Automatic Repeat Request (ARQ) process. If the following retransmission attempt fails, the CW is again increased exponentially, up until the limit CW_{max}. The retransmission process will repeat for up to 4 or 7 times, depending on whether the short retry limit or long retry limit is used. Upon reaching the retry limit the packet is considered lost and discarded. The retry limit is manufacturer dependent and can vary considerably.

2.5.2 Enhanced Distributed Channel Access (EDCA)

The enhanced access method EDCA builds on the legacy DCF process and introduces four different Access Categories (ACs) or traffic classes for service differentiation at the MAC layer. This is achieved by varying the size of CW in the backoff mechanism on a per category basis. Service differentiation is provided by the following methods:

■Arbitration Inter-Frame Space (AIFS) This is similar to the DIFS used in DCF, except the AIFS can vary according the access category.

■Variable Contention Window By giving higher priority traffic smaller contention windows, less time is spent in the back-off state, resulting in more frequent access to the medium.

■Transmission Opportunity (TxOP) This allows a station that has access to the medium to transmit a number of data units without having to contend for access to the medium. In fact this is a form of frame bursting. The TxOP limit is defined per traffic class.

Multiple AC queues can exist on a single station, contending with each other for the physical medium. This is regarded as virtual contention.

2.6 Ad Hoc Routing Protocols

There are many reasons why mobile ad hoc networking is being researched by many organizations and institutes around the globe. The dynamic nature and the lack of infrastructure of these networks, is asking more and more implementation of networking strategies and paradigms, to be able to provide efficient communication. Along with that, the variety of applications of MANETs in different scenarios, have made research interests growing in this field.

In MANETs, the well-known TCP/IP structure is used by the nodes to make the communication happen. However, due to their mobility and low resource capacities, for the MANETs to function efficiently, one should modify each layer of TCP/IP stack. Thus, many routing protocols and algorithms are developed and proposed, and each author of each of the protocols, claims improvements over existing approaches, for specific network scenarios. For a routing protocol to function efficiently in MANETs, it should have the following features:

- Self starting and self organizing,
- Multi-hop, loop-free paths,
- Dynamic topology maintenance,
- Rapid convergence,

- Minimal network traffic overhead,
- Scalable to large networks.

The routing protocols are separated into two main categories:

1. Reactive Ad hoc Routing Protocols (RMRP).
2. Proactive Ad hoc Routing Protocols (PMRP).

Adaptive or Hybrid Routing Protocols are also available, but these protocols use features of both RMRP and PMRP, mixed together. In this section, we give a short description of routing protocol catgories, some routing protocols for each category and some features of each. A review of these routing protocols when used in large scalable MANETs, can be found in [33].

2.6.1 Reactive Routing

In contrary with PMRPs, in RMRPs, routes are determined and maintained each time nodes require them to send data to a destiantion. In this category of routing protocols, the main control overhead is the route discovery traffic. Route discovery is done by flooding a route request packet in the network. When destination (or some node which has information about destination) is reached, a route reply packet is sent back via *link reversal*, or via *flooding* to probably find a better route. Reactive protocols can be classified into two categories: hop-by-hop routing and source routing.

In Hop-by-hop Routing, data packet headers consist only of the destination address and the next hop address. Thus, data packets are routed independetly by each node, based on local information, making routes adaptable to dynamically changing topology in MANETs. In this strategy, each node should have to maintain information about all active routes, and stay updated with all its neighbors. Although this is a disadvantage in MANETs, in this scenario topology information is fresher so we have better routes.

In Source Routed on-demand protocols each data packet is told the complete route from source to destination. Intermediate nodes, route these packets according to the information kept in the header of each packet. Thus, they do not need to maintain fresh routing information for each active route. They also do not need to maintain neghbour connectivity. In large networks source routing protocols do not scale well due to the added route overhead by bigger headers, and the increase of route failure probability (more nodes in a route).

RMRPs are designed to lower the overhead in proactive ones. Thus the main advantage of reactive routing is that, the bandwidth is used only when needed to find a route. The process of finding a route starts with a flooding and this usually brings initial delays. Its worth to mention some well-known RMRPs: Ad hoc On-demand Distance Vector (AODV), Dynamic Source Routing (DSR) and Temporally Ordered Routing Algorithm (TORA). We will describe AODV protocol in the following.

■Ad hoc On-demand Distance Vector (RFC 3561) AODV is one of the most popular reactive routing protocol for MANETs. Lets see how this routing protocols works in a general view. For most detailed description see [34].

As a reactive (on demand) protocol, when a node wants to transmit data, it first starts a route discovery process, by flooding a RREQ (Route Request) packet. The RREQ packet are forwarded by all the nodes by which it is recieved, until the destination is found. On the way to destination, the RREQ informs all the intermediate nodes about a route to the source. When the RREQ reaches the destination, destination sends a Route Reply (RREP) packet which follows the reverse path discovered by RREQ. This informs all intermediate nodes about a route to the destination node. After RREQ and RREP are delivered to their destination, each intermediate node on the route knows what node to forward data packets in order to reach source or destination. Thus data packets do not need to carry addresses of all intermediate nodes in the route. It just carries the address of the destination node, decreasing noticeably routing overheads.

A third kind of routing message, called route error (RERR), allows nodes to notify errors, for example, because a previous neighbor has moved and is no longer reachable. If the route is not active (i.e., there is no data traffic flowing through it), all routing information expires after a timeout and is removed from the routing table.

AODV is based on DSDV and DSR algorithms. The best advantage to DSR and DSDV is that in AODV, packets being sent (the RREP packet also) carry only the address of the destination and not the addresses of all the intermediate nodes to make the delivery. This lowers routing overheads. In AODV the route discovery process may last for a long time, or it can be repeated several times, due to potential failures during the process. This introduces extra delays, and consumes more bandwidth as the size of the network increases.

2.6.2 Proactive Routing

Proactive routing protocols, function in a way that each node maintains routing information to every other node (or nodes located in a specific part) in the network, in one or many tables or lists. This means that all routes are maintained during all the time of network operation. Topology changes, which is very frequent in MANETs brings a lot of traffic control information exchanged between nodes. PMRPs differ among each other in the way each node updates and detects the routing information, and the number of tables used to keep different types of information. Although the routes in PMRPs are always available, constant overhead is created by control traffic. Some of the most popular PMRPs are: Destination Sequenced Distance Vector (DSDV), Fisheye State Routing (FSR) and Optimized Link State Routing (OLSR). We will present OLSR in the following.

■Optimized Link State Routing (RFC3626) The OLSR protocol for mobile ad hoc networks is a PMRP. It is developed as a MANET compatible version of the classical link state algorithm. OLSR source code is available online, and it can be found in `http://www.olsr.org`. The new concept OLSR brought to MANET, is MultiPoint Relaying (MPR). Lets explain in short details the functioning of OLSR. The OLSR protocol can be divided in to three main modules:

- Neighbor/link sensing,
- Optimized flooding/forwarding (MPR),
- Link-State messaging and route calculation.

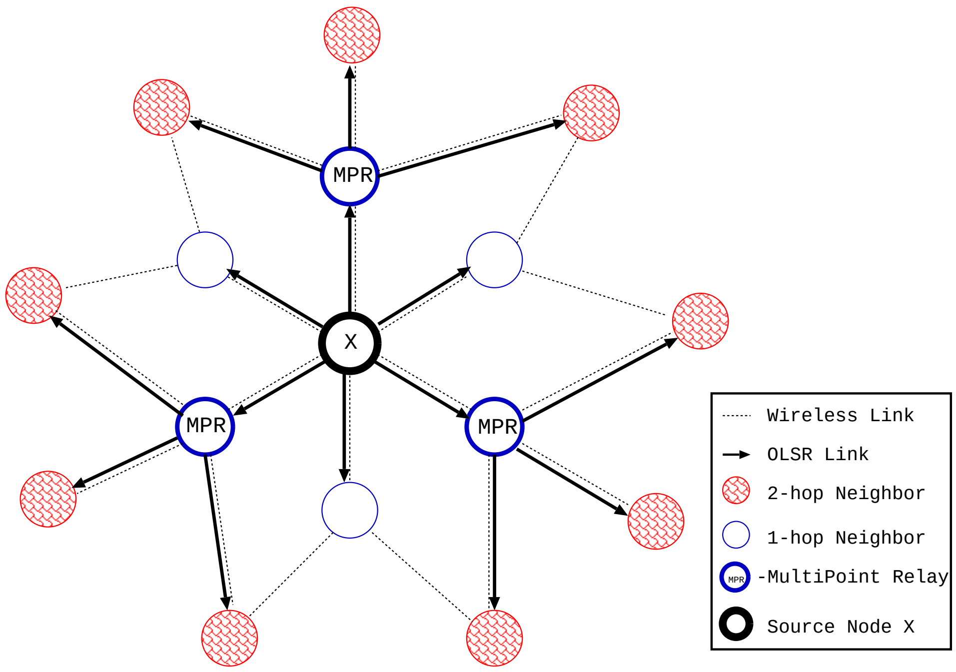


Fig. 2.3 MPRs selection, reaching all 2-hop neighbors

Neighbor and link sensing is realized by sending HELLO packets. All nodes transmit HELLO packets at a given interval. The 3-way handshake performed by two neighbors creates link information for both nodes. The HELLO packets also contain information about all active neighbors, so each node knows about 1-hop and 2-hop neighbors. Topology Control (TC) packets are also exchanged between nighbors to keep track of topology changing.

If OLSR would make a regular flooding of HELLO packets, too much unwanted traffic would flow on the network. The optimization of OLSR consists on exactly decreasing this traffic overhead. This is done by introducing the new concept of MPR. Node X choses a set $mpr(X)$ of MPRs from its 1-hop neighbors, so that all 2-hop neoghbors of node X is reached via the set $mpr(X)$. Flooding and forwarding is thus optimized in this way. A node recieveing a packet from node X, forwards or floods it, only if the node itself is in the set $mpr(X)$ of MPRs of node X. Fig. 2.3 shows how node X handles the selection of MPRs to cover all its 2-hop neighbors. Another optimizization consists on MPRs chosing to report only links between itself and its MPR selectors*2. So, partial information is distributed into the network.

An OLSR node, has one routing table, one neighbor table and one topology table. The routing table consists of 4 entries: destination address, next hop address, number of hops to destination and local interface address. The information of the routing table is acquired from TC messages (topological set) and HELLO messages (local link information). Changes occuring in both topology and local neighbors cause the routing table to be updated*3. The routing table is changed if one of the following happens.

- Neighbour link appear or dissapear,

*2 Node B selects node A, as one of its MPRs, at the same time also node C selects A as one of its MPRs. Node B and node C are both MPR selectors of node A.

*3 As OLSR is a proactive protocol, table is updated for every node changing in the network.

- Two-hops neighbor is created or removed,
- Topological link is appeared or lost,
- The multiple interface association information changes.

These leads to a call of *Route Calculation* function, usually performed by the shortest path algorithm. After recalculating the routes, the routing table is updated with the new information. Oldest versions of *olsrd*, calculate the shortest path with the hop-count as a target metric. Latest *olsrd* software have been equipped with Link Quality (LQ) extension, which uses the packetloss rate as target metric. This metric is called Expected Transmission Count (ETX) and is defined as $ETX(i) = 1/(NI(i) * LQI(i))$, where $NI(i)$ is the packet arrival rate seen by a node on the i-th link during the window W and $LQI(i)$ is the estimation of the packet arrival rate seen by its neighbor on the same link. This LQ extension enhances the packet delivery ratio in comparison with the old technique. Authors in [35] have found the optimal value of LQWS (Link Quality Window Size) for TCP flow, to be exactly 10. In [36] can be found the RFC3626 document for more detailed descriptions.

Anyway, the OLSR protocol is not implemented in practical scenarios. Routing tables taking a long time to build, routing loops and flapping routes are some of several issues that OLSR shows. A new routing protocol started to be developed, in order to overcome these issues. This new protocol will be described in the following.

2.6.3 Adaptive and Hybrid Routing Protocols

Proactive and reactive routing protocols have both its pros and cons. Thinking to get the best possible approach for a routing strategy, hybrid routing protocols appeared as trying to use the best features of both proactive and reactive. These protocols are designed to be scalable to large network. The whole network is separated in hierarchical regions, usually geographical. Some nodes are grouped trees, some trees or clusters, some clusters are grouped in a domain, and so on. Nodes within a region stay updated proactively, while to send data to a node in another region, route discovery process starts reactively. This strategy, lowers the route discovery overhead and supports very good scalability to larger networks.

■**Hybrid Wireless Mesh Protocol** The IEEE 802.11s draft defines a default routing protocol called the Hybrid Wireless Mesh Protocol (HWMP). Every IEEE 802.11s compliant device is required to implement HWMP and to be capable of using it. HWMP is located on layer 2, this means, it uses MAC addresses.

The nodes of a WMN are called Mesh Points (MPs) in IEEE 802.11s. A MP is an IEEE 802.11 station that has mesh capabilities in addition to the basic station functionality. This means that it can participate in the mesh routing protocol and can forward data frames on behalf of other MPs according to the IEEE 802.11s standard. The MPs can be end customer devices such as laptops as well as infrastructure devices such as APs.

The MPs with additional AP functionality are called Mesh AP (MAPs). Conventional WLAN clients, which are non-mesh IEEE 802.11 stations (STAs), can connect through the MAPs to the WMN. The MPs with additional portal functionality are called Mesh Portals Points (MPPs).

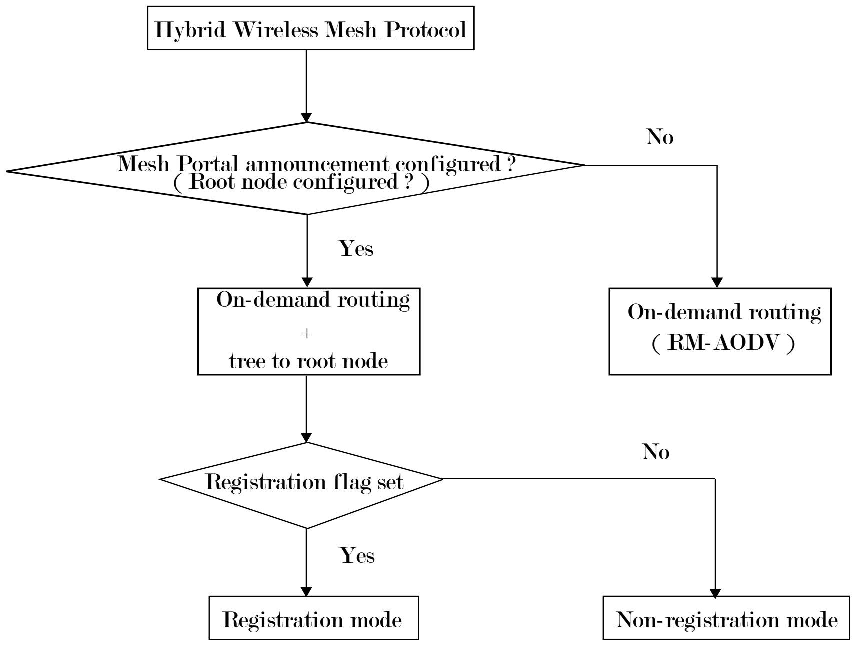


Fig. 2.4 Configuration process of HWMP.

They can bridge data frames to other IEEE 802 networks, especially to a wired network such as an Ethernet.

The IEEE 802.11s WMNs will be applicable to a large variety of usage scenarios. The four most important usage scenarios identified by the task group [37, 38] are:

- residential for wireless home networks;
- office for wireless networks in office environments;
- campus/community/public access for wireless backhaul meshes for Internet access;
- public safety for flexible and fast setup of wireless communications for emergency staff.

The HWMP is a hybrid routing protocol. It has both reactive components and proactive components. The hybrid routing configuration process is shown in Fig. 2.4.

The foundation of HWMP is an adaptation of AODV [34] to radio-aware link metrics and MAC addresses. It is the basic, reactive component of HWMP. The on-demand path setup is achieved by a path discovery mechanism that is very similar to the one of AODV. If a MP needs a path to a destination, it broadcasts a Path REQuest message (PREQ) into the WMN. The MPs will rebroadcast the updated PREQ whenever the received PREQ corresponds to a newer or better path to the source. Similarly, the requested destination MP will respond with a Path REPly message (PREP) whenever a received PREQ corresponds to a newer or better path to the source. Intermediate MPs that have already a valid path to the requested destination, can respond with a PREP, if the Destination Only flag (DO flag) is not set. Depending on the new Reply and Forward flag (RF flag), they can also rebroadcast the updated PREQ. This will result in a current path metric in addition to the fast path discovery.

The proactive component of HWMP is the extension with a proactive routing tree to specially designated MPs. Any MP that is configured to be a root MP, will periodically broadcast proactive PREQ messages or Root ANNouncement messages (RANNs) into the WMN, which will create and maintain a tree of paths to the root MP. There are three different, configurable mechanisms for the proactive tree-building available in HWMP.

The IEEE 802.11 Task Group "s" is standardizing the Airtime Link Metric, which tries to minimize medium usage by taking into account not only the datarates that a given link can support, but also the probability of success on the transmission of frames. The Airtime Link Metric formula:

$$c_a = \left(O_{ca} + O_p + \frac{B_t}{r}\right)\frac{1}{1 - ept}, \tag{2.8}$$

where O_{ca} is a channel access overhead latency that varies according to the PHY layer implementation, O_p is a protocol overhead latency, B_t is the test frame size (1024 bytes), r is the data rate in Mb/s at which the MP would transmit a test frame and e_{pt} is the measured test frame error rate.

2.7 Mobile Ad-hoc Networks (MANETs)

Recently, MANETs have drawn too much attention as a research field. As a result of this considerable research activity, the basic mechanisms that enable wireless ad hoc communication have been designed and standardized. But the usage in real-life application, is still in its testing steps. The future seems to be so bright for MANETs, which will take advantage of its most distinguishable characteristics, mobility and multi-hop, to take the place of wired multi-hop backbones, because they are so easy and inexpensive to be implemented, even in areas where infrastructure is impossible to appear.

MANETs are the networks of the future in many applications. By using Ad hoc philosophy, each user (node) gets in and out of the network whenever it finds it convenient, without being noticed by other users. This means, that even in the worst case of an unexpected failure of a node, dhe network is still up. This brings a lot of advantages in using MANETs in many applications. Some of the most common application fields, in which MANETs have great success are:

Military Scenarios, in battlefield tend to make a very quick deployment of information networks, and without the use of any infrastructure or centralized administration.

Sensor Networks, are a very interesting application of MANETs, where a bunch of sensors, equipped with a radio antenna, are able to send usefull collected information to where we want, or compute agregate values of the parameters sensed in the environment.

Students Campus. That is a very useful application, for every environment where density of wireless terminals is high enough to cover the intended area.

Conferences. The property of quick-deployment and mobility, make MANETs an adaptive tool to keep everyone connected in the conferences.

OLPC One Laptop Per Child project [39] is being implemented in developing countries, where sometimes there is no proper infrastructure for all laptops to stay connected. MANETs provide the solution here.

2.8 Vehicular Ad-hoc Network (VANET)

VANETs is a mobile ad hoc network to provide communication among nearly vehicle and between vehicle and nearby vehicle or vehicle and nearby fixed equipment, so this called vehicle to vehicle (V2V) and vehicle to infrastructure communication [40].

As we discuss above the main application area of the VANETs is to provide safe, secure and enjoyable ride for the passengers For this each vehicle places a special electronic device inside the vehicle so that vehicle can connect to the ad hoc network. In the VANETs network there are no fixed infrastructure and server communication. Each vehicle has a VANETs device which works as a node in the ad hoc network. The entire vehicle can communicate with each other by using the wireless network [41].

There are also internet and multimedia facility for passenger. The VANETs system also sends the emergency signal to the other vehicle in the case of the accident or sudden breaking. The other advance improvement in the architecture is driver assistant system. In this the driver can assist the thing beyond his field of vision. By this security system the driver can get information about accident traffic jam, signal information, resulting in more efficient and safe driving.

2.9 Wireless Sensor and Actor Networks (WSANs)

2.9.1 WSAN Architectures

The main functionality of WSANs is to make actors perform appropriate actions in the environment, based on the data sensed from sensors and actors. When important data has to be transmitted (an event occurred), sensors may transmit their data back to the sink, which will control the actors' tasks from distance, or transmit their data to actors, which can perform actions independently from the sink node. Here, the former scheme is called Semi-Automated Architecture and the latter one Fully-Automated Architecture. Obviously, both architectures can be used in different applications. In the Fully-Automated Architecture are needed new sophisticated algorithms in order to provide appropriate coordination between nodes of WSAN. On the other hand, it has advantages, such as low latency, low energy consumption, long network lifetime [42], higher local position accuracy, higher reliability and so on.

2.9.2 WSAN Challenges

Some of the key challenges in WSAN are related to the presence of actors and their functionalities.

- Deployment and Positioning: At the moment of node deployment, algorithms must consider to optimize the number of sensors and actors and their initial positions based on applications [43, 44].
- Architecture: When important data has to be transmitted (an event occurred), sensors may transmit their data back to the sink, which will control the actors' tasks from distance

or transmit their data to actors, which can perform actions independently from the sink node [45].

- Real-Time: There are a lot of applications that have strict real-time requirements. In order to fulfill them, real-time limitations must be clearly defined for each application and system [46].
- Coordination: In order to provide effective sensing and acting, a distributed local coordination mechanism is necessary among sensors and actors [45].
- Power Management: WSAN protocols should be designed with minimized energy consumption for both sensors and actors [47].
- Mobility: Protocols developed for WSANs should support the mobility of nodes [48, 49] where dynamic topology changes, unstable routes and network isolations are present.
- Scalability: Smart Cities are emerging fast and WSAN, as a key technology will continue to grow together with cities. In order to keep the functionality of WSAN applicable, scalability should be considered when designing WSAN protocols and algorithms [44, 49].

2.10 Internet of Things (IoT)

The Internet of Things (IoT) is a recent communication paradigm that envisions a near future, in which the objects of everyday life will be equipped with microcontrollers, transceivers for digital communication, and suitable protocol stacks that will make them able to communicate with one another and with the users, becoming an integral part of the Internet [50, 51]. The IoT concept, hence, aims at making the Internet even more immersive and pervasive. Furthermore, by enabling easy access and interaction with a wide variety of devices such as, for instance, home appliances, surveillance cameras, monitoring sensors, actuators, displays, vehicles, and so on, the IoT will foster the development of a number of applications that make use of the potentially enormous amount and variety of data generated by such objects to provide new services to citizens, companies, and public administrations. This paradigm indeed finds application in many different domains, such as home automation, industrial automation, medical aids, mobile healthcare, elderly assistance, intelligent energy management and smart grids, automotive, traffic management, and many others [52].

2.11 Ambient Intelligence (AmI)

In the future, small devices will monitor the health status in a continuous manner, diagnose any possible health conditions, have conversation with to persuade them to change the lifestyle for maintaining better health, and communicates with the doctor, if needed [53]. The device might even be embedded into the regular clothing fibers in the form of very tiny sensors and it might communicate with other devices including the variety of sensors embedded into the home to monitor the lifestyle. For example, people might be alarmed about the lack of a healthy diet based on the items present in the fridge and based on what they are eating outside regularly.

The AmI paradigm represents the future vision of intelligent computing where environments support the people inhabiting them [54–56]. In this new computing paradigm, the conventional input and output media no longer exist, rather the sensors and processors will be integrated into everyday objects, working together in harmony in order to support the inhabitants [57]. By relying on various artificial intelligence techniques, AmI promises the successful interpretation of the wealth of contextual information obtained from such embedded sensors, and will adapt the environment to the user needs in a transparent and anticipatory manner. An AmI system is particularly identified by several characteristics:

- **Context Aware:** It exploits the contextual and situational information.
- **Personalized:** It is personalized and tailored to the needs of each individual.
- **Anticipatory:** It can anticipate the needs of an individual without the conscious mediation of the individual.
- **Adaptive:** It adapts to the changing needs of individuals.
- **Ubiquity:** It is embedded and is integrated into our everyday environments.
- **Transparency:** It recedes into the background of our daily life in an unobtrusive way.

Besides characteristics such as transparency and ubiquity, an important characteristic of ambient intelligence is the intelligence aspect. By drawing from advances in Artificial Intelligence (AI), AmI systems can be even more sensitive, responsive, adaptive, and ubiquitous. While ambient intelligence draws from the field of AI, it is not synonymous with AI [58]. In addition to the AI sub-areas such as reasoning, activity recognition, decision making and spatio-temporal logic, an ambient intelligence system has to rely upon advances in variety of other fields. Some example areas include "sensor networks" to facilitate the data collection, "robotics" to build actuators and assistive robots, and "human computer interaction" to build more natural interfaces.

Nowadays, we are surrounded by various computing devices such as personal computers, smart phones, GPS, tablets, various sensors such as RFID tags, infrared motion sensors, as well as biometric identification sensors. The widespread presence of such devices and sensors and accompanying services such as location service has already sparked the realization of ambient intelligence. In addition, recent computational and electronics advancements have made it possible for researchers to work on ambitious concepts such as smart homes, and to bring us one step closer to the full realization of ambient intelligence in our daily environments.

Chapter 3

Wireless Mesh Networks

3.1 Introduction to WMNs

WMNs are formed by several wireless terminals, which can be mobile or semi-mobile. These terminals, or nodes, do not have a pre-estabilished infrastracture, meaning that they create a fast and temporary network whenever they are deployed in an environment. Each of the nodes has a wireless interface and communicate with each other over radio or infrared links in a Peer-to-Peer (P2P) design. Examples of WMN nodes are, notebooks and PDAs used widely now everywhere. In general, nodes in WMNs are mobile, and their movement is random and difficult to be modelled, but according to mankind lifestyle, we always try to implement similar models to what happens in real life. Some nodes can be static as well. They can be used to interconnect the Ad Hoc Network to another network or the Internet, or the user simply is sitting with his notebook and using network resources.

WMNs need to have implemented some mechanisms as follows

- If provides internetworking, an Internet access mechanisms is needed.
- Self configuring networks requires an address allocation mechanism.
- Mechanism to detect and act on merging of existing networks.
- Security mechanisms.
- Nodes must be able to relay traffic since communicating nodes might be out of range.
- Multihop operation requires a routing mechanism designed for mobile nodes.

WMNs are dynamically self-organized and self-configured, with the nodes in the network automatically establishing an ad hoc network and maintaining the mesh connectivity. WMNs are comprised of two types of nodes: mesh routers and mesh clients. Other than the routing capability for gateway/bridge functions as in a conventional wireless router, a mesh router contains additional routing functions to support mesh networking. Through multi-hop communications, the same coverage can be achieved by a mesh router with much lower transmission power. To further improve the flexibility of mesh networking, a mesh router is usually equipped with multiple wireless interfaces built on either the same or different wireless access technologies. In spite of all these differences, mesh and conventional wireless routers are usually built based on a similar hardware platform.

Mesh routers have minimal mobility and form the mesh backbone for mesh clients. Thus, although mesh clients can also work as a router for mesh networking, the hardware platform and software for them can be much simpler than those for mesh routers. For example, communication protocols for mesh clients can be light-weight, gateway or bridge functions do not exist in mesh clients, only a single wireless interface is needed in a mesh client, and so on.

In addition to mesh networking among mesh routers and mesh clients, the gateway/bridge functionalities in mesh routers enable the integration of WMNs with various other networks. Conventional nodes equipped with wireless network interface cards (NICs) can connect directly to WMNs through wireless mesh routers. Customers without wireless NICs can access WMNs by connecting to wireless mesh routers through, for example, Ethernet. Thus, WMNs will greatly help users to be always-on-line anywhere, anytime.

Consequently, instead of being another type of ad-hoc networking, WMNs diversify the capabilities of ad-hoc networks. This feature brings many advantages to WMNs, such as low up-front cost, easy network maintenance, robustness, reliable service coverage, etc. Therefore, in addition to being widely accepted in the traditional application sectors of ad hoc networks, WMNs are undergoing rapid commercialization in many other application scenarios such as broadband home networking, community networking, building automation, highspeed metropolitan area networks, and enterprise networking.

To date, several companies have already realized the potential of this technology and offer wireless mesh networking products. A few testbeds have been established in university research labs. However, for a WMN to be all it can be, considerable research efforts are still needed. For example, the available MAC and routing protocols are not scalable; throughput drops significantly as the number of nodes or hops in WMNs increases. Thus, existing protocols need to be enhanced or re-invented for WMNs. Researchers have started to revisit the protocol design of existing wireless networks, especially of IEEE 802.11 networks, ad hoc networks, and wireless sensor networks, from the perspective of wireless mesh networking. Industrial standards groups, such as IEEE 802.11, IEEE 802.15, and IEEE 802.16, are all actively working on new specifications for WMNs.

3.2 Architectures of WMNs

In this section, we describe the architectures of WMN. The architecture of the nodes in WMNs [59, 60] can be classified according to the functionalities they offer as follows:

Infrastructure/Backbone WMNs: In this architecture, mesh routers form an infrastructure for clients. 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 a backbone for conventional clients and enables integration of WMNs with existing wireless networks, through gateway/bridge functionalities in mesh routers. Conventional clients with an Ethernet interface can be connected to mesh routers via Ethernet links. For conventional clients with the same radio technologies as mesh routers, they can directly communicate with mesh routers. If different radio technologies are used, clients must communicate with their base stations that have Ethernet connections to mesh routers.

Client WMNs: Client meshing provides peer-to-peer networks among client devices. 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. Client WMNs are usually formed using one type of radios on devices. Thus, a Client WMN is actually the same as a conventional ad hoc network. However, 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.

Hybrid WMNs: This architecture is the combination of infrastructure and client meshing.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 WMNs.

The characteristics of WMNs are outlined below, where the hybrid architecture is considered for WMNs, since it comprises all the advantages of WMNs:

- WMNs support ad hoc networking, and have the capability of self-forming, self-healing, and self-organization.
- WMNs are multi-hop wireless networks, but with a wireless infrastructure/backbone provided by mesh routers.
- Mesh routers have minimal mobility and perform dedicated routing and configuration, which significantly decreases the load of mesh clients and other end nodes.
- Mobility of end nodes is supported easily through the wireless infrastructure.
- Mesh routers integrate heterogeneous networks, including both wired and wireless. Thus, multiple types of network access exist in WMNs.
- Power-consumption constraints are different for mesh routers and mesh clients.
- WMNs are not stand-alone and need to be compatible and interoperable with other wireless networks.

Therefore, WMNs diversify the capabilities of ad-hoc networks instead of simply being another type of ad hoc network. These additional capabilities necessitate new algorithms and design principles for the realization of WMNs.

3.3 Application Scenarios of WMNs

There are a number of application scenarios for which the use of WMNs is a very good alternative to offer connectivity at a low cost. It should also mentioned that there are applications of WMNs which are not supported directly by other types of wireless networks such as cellular networks, ad hoc networks, wireless sensor networks and standard IEEE 802.11 networks.

■Neighboring Community Networks In a community, the usual solution is to deploy ADSL or cable. However, there are a number of limitations that WMNs can improve as shown in following.

- A large percentage of areas between the houses could not receive wireless services.
- A broadband gateway between different houses could not be shared and wireless services should be established individually.
- A single path to each neighbor can communicate with the rest of neighbors or with the outside.

■Corporative Networks This scenario corresponds to having a small network for an office or a medium sized network for all offices of a building or even a network to communicate offices located in different buildings. Other similar scenarios include airports, hotels, shopping centers or sports centers.

■Metropolitan Area Networks Deploying WMNs in metropolitan areas has a number of advantages. The physical layer provides a higher average transmission to any cellular network and need not depend on a wiring. Also, deploying such infrastructure is much cheaper than cable or fiber and can be easily and rapidly deployed in areas with few resources, which have never had any network before.

■Other Scenarios There are many more scenarios for which WMNs can be used. We mention some of them in following.

- *Transportation Systems*: Provide information services to passengers, remote monitoring of vehicle safety and communications by the driver.
- *Automatic Control Buildings*: In buildings there are several electrical devices to be controlled, including light, elevator, air conditioning, and so on.
- *Medical and Health Systems*: In a hospital information monitoring and diagnosis must be transmitted from one room to another.
- *Surveillance*: In corporate buildings, shopping malls and stores need broadband data transmission (images and videos basically) for monitoring and surveillance purposes.

3.4 WMNs Evaluation Techniques

In a research environment, research tools are required to test, verify and identify problems of an algorithm or protocol. These tools are classified in three major techniques: simulators, emulators and real-world testbeds. We will describe them shortly in the following.

3.4.1 Simulations

A simulation system consists of many assumptions and artificial modeling, in order to reach a certain realistic degree. However, these assumptions and modeling can have errors and in some cases, some realistic effects are not even considered, e.g. gray zones effect [61] are not considered in the well-known simulator *ns-2*. In the early phases of the development of an algorithm or protocol, usually after the analytical modeling, simulations can give a quick and inexpensive result regarding the theoretical performance. Moreover, we can keep unchanged the simulated conditions and parameters and run the simulations as many times as we want.

3.4.2 Emulators

With a higher degree of realism than simulators, emulators can still control the repeatability of tests and use real hardware combined with simulation software, to conduct experiments in controlled conditions. They use artificial assumptions which are sometimes unrealistic. Emulators can be divided into physical layer emulators and MAC layer emulators. Physical layer emulators, e.g. EWANT [62], use the attenuation of the radio signal to emulate movement or obstacles. MAC layer emulators use MAC filter tools, e.g. Dummynet [63], to decide network topology and emulate mobility. Emulators have higher costs than simulators because they use real hardware.

3.4.3 Real-World Testbeds

Real-world testbeds have the higher level of realism because they are not based on assumptions about the experimental conditions. In testbeds, when mobility is present, the node changes its geographical location, which can have different effects on the performance. Testbeds are usually used on the final stages of the development of an algorithm or protocol. Simulation and emulation systems can make assumptions based on experimental results provided by testbeds. However, real-world testbed implementations have higher costs for hardware software and working hours. Also, the repeatability of tests in a testbed is a complicated and costly task.

Chapter 4

Intelligent Algorithms

4.1 Introduction to Intelligent Algorithms

Optimization in simulation has been a topic of intense investigation for decades, but for most of this period, the research effort was rarely transferred to simulation practice [64]. This has changed dramatically over the past decade, however, and optimization routines are now part of most commercial simulation software. These simulation softwares use metaheuristics that have been most extensively studied to solve deterministic combinatorial optimization problems [65–67].

In this chapter, we address the use of metaheuristics (or Intelligent Algorithms) in simulation optimization. In simulation practice, the choice and design of an algorithm always boils down to computational efficiency [68]. Rapid progress of the search and demonstrating improvement over the initial solution therefore takes precedence over possible convergence statements, and provably obtaining a global optimum is rarely a significant concern. While this is certainly a reasonable stance in practice, a consequence of this view may be that the simulation takes on a subservient role to the optimization [66]. This is an undesirable property for simulation optimization software. Thus, the challenge to the research community is to shift the focus back to the simulation itself, by rigorously accounting for the simulation noise in the optimization routines, and establishing convergence results that are both well founded and useful in practice.

Metaheuristics are designed to tackle complex optimization problems where other optimization methods have failed to be either effective or efficient. These methods have come to be recognized as one of the most practical approaches for solving many complex problems, and this is particularly true for the many real-world problems that are combinatorial in nature. The practical advantage of metaheuristics lies in both their effectiveness and general applicability. In the early research literature, specialized heuristics were typically developed to solve complex combinatorial optimization problems. This required a new approach to every problem and lessons learned from one problem did not always generalize well to a different class of problems. On the other hand, with the emergence of more general solution strategies, including such metaheuristics as tabu search, genetic algorithms, and simulated annealing, the main challenge has become adapting the metaheuristics to a particular problem or problem class. This usually requires much less work than developing a specialized heuristic for a specific application, which makes metaheuristics an appealing choice for implementation in general purpose software. Furthermore, a good metaheuristic implementation is likely to provide near optimal solutions in reasonable computation times. For further reading, [69] provide a good introduction and general reference to many of the most popular metaheuristics.

The applicability of metaheuristics as a preferred method over other optimization methods is primarily to find good heuristic solutions to complex optimization problems with many local optima and little inherent structure to guide the search. The metaheuristic approach to solving such problem is to start by obtaining an initial solution or an initial set of solutions, and then initiating an improving search guided by certain principles. The structure of the search has many common elements across various methods. In each step of the search algorithm, there is always a solution (or a set of solutions) θ_k, which represents the current state of the algorithm. Many metaheuristics, including simulated annealing, tabu search and variable neighborhood search are solution-to-solution search methods, that is, θ_k is a single solution or point $\theta_k \in \Theta$ in some solution space Θ. Others, including genetic algorithms, scatter search, and the nested partitions method, are set-based, that is, in each step θ_k represents a set of solutions $\theta_k \in \Theta$. However, the basic structure of the search remains the same regardless of whether the metaheuristics is solution-to-solution or set-based.

Given a neighborhood $N(\theta_k)$ of the solution (set), a candidate solution (set) $\theta^c \subset N(\theta_k)$ is selected and evaluated. This evaluation involves calculating or estimating the performance of the candidate solution(s) and comparing them with the performance of θ_k and sometimes with each other. Based on this evaluation, the candidate may be either accepted, in which case $\theta_{k+1} = \theta^c$ rejected, in which case $\theta_{k+1} = \theta_k$. A metaheuristic framework is shown in Algorithm 1.

Algorithm 1 Metaheuristic framework

1: Obtain an initial solution (set) θ_0
2: and set $k = 0$.
3: **for** Until stopping criterion is satisfied. **do**
4: Identify the neighborhood $N(\theta_k)$ of the current solution(s).
5: Select candidate solution(s) $\theta^c \subset N(\theta_k)$ from the neighborhood.
6: Accept the candidate(s) and set $\theta_{k+1} = \theta^c$ or reject it and set $\theta_{k+1} = \theta_k$.
7: Increment $k = k + 1$.
8: **end for**

The reason for the meta-prefix is that metaheuristics do not specify all the details of the search, which can thus be adapted by a local heuristic to a specific application. Instead, they specify general strategies to guide specific aspects of the search. For example, tabu search uses a list of solutions or moves called the tabu list, which ensures the search does not revisit recent solutions or becomes trapped in local optima. The tabu list can thus be thought of as a restriction of the neighborhood. On the other hand, methods such as genetic algorithm specify the neighborhood as all solutions that can be obtained by combining the current solutions through certain operators. Other methods, such as simulated annealing, do not specify the neighborhood in any way, but rather specify an approach to accepting or rejecting solutions that allows the method to escape local optima. Finally, the nested partitions method is an example of a set-based method that selects candidate solutions from the neighborhood with a probability distribution that adapts as the search progresses to make better solutions be selected with higher probability.

All metaheuristics share the elements of selecting candidate solution(s) from a neighborhood of the current solution(s) and then either accepting or rejecting the candidate(s). With this perspective, each metaheuristic is thus defined by specifying one or more of these elements, but allowing others to be adapted to the particular application. This may be viewed as both strength and a liability. It implies that we can take advantage of special structure for each application, but it also means that the user must specify those aspects, which can be complicated. For the remainder of this section, we briefly introduce a few of the most common metaheuristics and discuss how they fit within this framework. Three of those methods will then be analyzed in more detail as we discuss how to apply them for simulation optimization in subsequent sections.

One of the earliest metaheuristics is simulated annealing [70–72], which is motivated by the physical annealing process, but within the framework here simply specifies a method for determining if a solution should be accepted. As a solution-tosolution search method, in each step it selects a candidate $\theta^c \subset N(\theta_k)$ from the neighborhood of the current solution $\theta_k \in \Theta$. The definition of the neighborhood is determined by the user. If the candidate is better than the current solution it is accepted, but if it is worse it is not automatically rejected, but rather accepted with probability

$$P[\text{Accept } \theta^c] = e^{\frac{f\theta_k - f\theta^c}{T_k}}, \tag{4.1}$$

where $f : \Theta \to \mathbb{R}$ is an objective function to be minimized, and T_k is a parameter called the temperature. Clearly, the probability of acceptance is high if the performance difference is small and T_k is large. The key to simulated annealing is to specify a cooling schedule $\{T_k\}_{k=1}^{\infty}$ which the temperature is reduced so that initially inferior solutions are selected with a high enough probability so local optimal are escaped, but eventually it becomes small enough so that the algorithm converges.

Tabu search is another popular metaheuristics [73–75]. As is the case for simulated annealing, it is a solution-to-solution search method where the neighborhood is specified by the user. However, the defining characteristic of tabu search is in how solutions are selected from the neighborhood. In each step of the algorithm, there is a list L_k of solutions that were recently visited and are therefore tabu. The algorithm looks through all of the solutions of the neighborhood that are not tabu and selects the best one, that is,

$$\theta^c = \underset{\theta \in N(\theta_k) \cap L_k}{\arg\min} f(\theta), \tag{4.2}$$

where as before $f : \Theta \to \mathbb{R}$ is to be minimized. The candidate solution θ^c is accepted even if it is worse than the current solution, that is, $P[Accept\theta^c] = 1$. Accepting inferior solutions allows the search to escape local optima, and the tabu list prevents the search from immediately reverting to the previous solution [73].

Other popular solution-to-solution metaheuristics include the greedy randomized adaptive search procedure (GRASP) and the variable neighborhood search (VNS). The defining property of GRASP is its multi-start approach that initializes several local search procedures from different starting points. The advantage of this is that the search becomes more global, but on the other hand, each search cannot use what the other searches have learned, which introduces some inefficiency. The VNS is interesting in that it uses an adaptive neighborhood structure, which changes based on the performance of the solutions that are evaluated. More information on GRASP can be found in [76], and for an introduction to the VNS approach we refer the reader to [77].

Several metaheuristics are set-based or population based, rather than solution-to-solution. This includes genetic algorithms and other evolutionary approaches, as well as scatter search and the nested partitions method. All of these methods are readily adapted to simulation optimization, and both genetic algorithms and the nested partitions method are discussed in detail in later sections. All evolutionary algorithms are based on the idea of natural selection, where a population of solutions evolves, or improves, through a series of genetic operators [78–80]. This includes survival of the fittest or best solutions, crossover, which is simply a combination of two fit solutions, and mutation, which is a slight modification to fit solutions. From the perspective of the general framework, these operators define the neighborhood of a solution set. Thus, given a current set $\theta_k \subseteq \Theta$ of solutions, the neighborhood is defined as

$$N(\theta_k) = N^{\text{crossover}}(\theta_k) \cup N^{\text{mutation}}(\theta_k) \cup \theta_k, \tag{4.3}$$

where

$$N^{\text{crossover}}(\theta_k) = \{\Psi \in \Theta | \Psi \text{ is the crossover of two solutions } \xi_1, \xi_2 \in \theta_k\}, \tag{4.4}$$

$$N^{\text{mutation}}(\theta_k) = \{\Psi \in \Theta | \Psi \text{ is a mutation of some } \xi, \in \theta_k\}. \tag{4.5}$$

For evolutionary methods, the key feature is therefore this innovative definition of a neighborhood, which allows the search to quickly and intelligently traverse large parts of the solution space.

Scatter search is another metaheuristic related to the concept of evolutionary search. In each step a scatter search algorithm considers a set, $\theta_k \subseteq \Theta$, of solutions called the reference set. Similar to the genetic algorithm approach, these solutions are then combined into a new set $\theta_{k+1} \subseteq \Theta$. However, as opposed to the genetic operators, in scatter search the solutions are combined using linear combinations, which thus define the neighborhood $N(\theta_k)$. For references on scatter search, we refer the reader to [81]. The final method is the nested partitions method [82], which like genetic algorithms and scatter search is a set-based metaheuristic. Unlike any of the previous methods, however, the nested partitions method is global in that the neighborhood is always the entire set of feasible solutions. Thus, given a current set $\theta_k \subseteq \Theta$ of solutions, the neighborhood is always $N(\theta_k) = \Theta$. In addition to the global perspective, the defining element of this method is the adaptive distribution that is used to obtain sample solutions from this neighborhood. By going through a sequence of set partitions, with each partition nested within the last, the sampling is concentrated in sets that are considered promising, that is, where the optimal solution is believed to be contained. Once the random candidate solutions have been generated according to this adaptive distribution, they are always accepted, and the search continues with a new sampling distribution.

Although the metaheuristics discussed in this section are usually considered separately in the literature, they have many common elements that make it possible to analyze them within a common framework. An interesting step in this direction is the generalized hill-climbing (GHC) algorithm framework of [83]. The GHC framework is general enough to cover various stochastic local search algorithms, including both simulated annealing and tabu search. Analysis of the GHC can be found in [84, 85].

4.2 Genetic Algorithm

In this section, we present GA. As an approach to global optimization, the GAs have been found to be applicable to optimization problems that are intractable for exact solutions by conventional methods [79, 86]. It is a set-based search algorithm, where at each iteration it simultaneously generates a number of solutions. In each iteration, a subset of the current set of solutions is selected based on their performance and these solutions are combined into new solutions. The operators used to create the new solutions are survival, where a solution is carried to the next iteration without change, crossover, where the properties of two solutions are combined into one, and mutation, where a solution is modified slightly. The same process is then repeated with the new set of solutions. The crossover and mutation operators depend on the representation of the solution, but not on the evaluation of its performance. They are thus the same even though the performance is estimated using simulation. The selection of solutions, however, does depend on the performance. The general principle is that high performing solutions (which in GAs are referred to as fit individuals) should have a better change of both surviving and being allowed to create new solutions through crossover. The simplest approach is to order the solutions $J(\theta_{[1]}) \leq J(\theta_{[2]}) \leq \ldots \leq J(\theta_{[n]})$, and only operate on the best solutions. If a strict selection of say, the top k solutions, were required, this would complicate the issue significantly in the simulation optimization context, and considerable simulation effort would have to be spent to obtain an accurate ordering of the solutions.

Fortunately, GAs appear to be quite robust with respect to which solutions are selected to create the next set. Indeed, a purely deterministic selection of the top k solution is typically not the best approach for deterministic problems, and some randomness is usually introduced into the process. A popular example of this is the roulette strategy, which several authors have used for the application of genetic algorithms to stochastic problems (see e.g., [87–91]). In the roulette strategy the probability of selecting a solution θ is calculated as follows:

$$P(\theta) = \frac{\hat{f}(\theta)}{\sum\limits_{\text{ALL } \theta} \hat{f}(\theta)}. \tag{4.6}$$

Here $\hat{f}(\theta)$ is an estimate of the fitness function $f : \Theta \rightarrow \mathbb{R}$ that measures the quality of the solution and is to be maximized (higher value implies more fit). Thus, every solution has a positive probability of being selected, but the fitter solutions are selected with higher probability. Assuming an unbiased simulation estimate, this statement will continue to be true when the roulette strategy is applied to simulation optimization. However, existing studies in this area are based almost exclusively on numerical evaluations and any the claim is simply made that genetic algorithms are robust and hence applicable to simulation optimization. While the robustness of genetic algorithms with respect to noise in the selection method is unquestionable, it would be desirable to have research that provides a better understanding into how much noise is acceptable, and thus how much simulation effort should be devoted to the evaluation of solutions during the search.

In another study [92] the GA is used together with statistical selection method to develop a system for simulation optimization. Their approach consists of three phases. First, there is an initialization phase, where the parameters of the algorithms are specified. Second, there is the actual search phase, that is, the usual iteration of the genetic algorithm selection, crossover, and mutation operators. Finally, then the search terminates there is a solutions phase where the alternatives generated by the GA search are evaluated using ranking and selection. In particular, they use a screening and selection procedure to first quickly filter out inferior solutions and then determine the best solution by carrying out additional simulation runs for the remaining solutions. Their design choice is to use what is called anti-ranks and a q-tournament selection [93], which also appears to be quite robust with regards to the simulation noise.

4.3 Tabu Search

Tabu search was introduced by [74,75] to solve combinatorial optimization problems and it has been used effectively for simulation optimization, most notably by the OptQuest simulation optimization software [65]. It is a solution-to-solution method and the main idea is to make certain moves or solutions tabu, that is they cannot be visited as long as they are on what is called the tabu list. The tabu list L_k is dynamic and after each move, the latest solution θ_k, or the move that resulted in this solution, is added to the list and the oldest solution or move is removed from the list. Another defining characteristic of tabu search is that the search always selects the best non-tabu solution from the neighborhood, even if it is worse than the current solution. This allows the search to escape local optima, and the tabu list ensures that the search does not revert back. Tabu search numerous other elements, such as long-term memory that restarts the search, with a new tabu list, at previously found high quality solutions, and a comprehensive treatment of this methodology can be found in [73].

4.4 Hill Climbing

The Stochastic Hill Climbing algorithm is a Stochastic Optimization algorithm and is a Local Optimization algorithm (contrasted to Global Optimization). It is a direct search technique, as it does not require derivatives of the search space. Stochastic Hill Climbing is an extension of deterministic hill climbing algorithms such as Simple Hill Climbing (first-best neighbor), Steepest-Ascent Hill Climbing (best neighbor), and a parent of approaches such as Parallel Hill Climbing and Random-Restart Hill Climbing.

The strategy of the Stochastic Hill Climbing algorithm is iterate the process of randomly selecting a neighbor for a candidate solution and only accept it if it results in an improvement. The strategy was proposed to address the limitations of deterministic hill climbing techniques that were likely to get stuck in local optima due to their greedy acceptance of neighboring moves.

The Algorithm 2 provides a pseudocode listing of the Stochastic Hill Climbing algorithm for minimizing a cost function, specifically the Random Mutation Hill Climbing algorithm applied to a maximization optimization problem.

Algorithm 2 Best neighbor selection for maximization

```
Input: Iter_max, ProblemSize.
Current ← RandomSolution(ProblemSize).
for iter_i ∈ Iter_max do
    Candidate ← RandomNeighbor(Current);
    if (Cost(Candidate) ≥ Cost(Current)) then
        Current ← Candidate
    end if
end for
return (Current);
```

4.5 Simulated Annealing

Simulated annealing (SA) is a random-search technique which exploits an analogy between the way in which a metal cools and freezes into a minimum energy crystalline structure (the annealing process) and the search for a minimum in a more general system; it forms the basis of an optimisation technique for combinatorial and other problems.

Simulated annealing was developed in 1983 to deal with highly nonlinear problems. SA approaches the global maximization problem similarly to using a bouncing ball that can bounce over mountains from valley to valley. It begins at a high "temperature" which enables the ball to make very high bounces, which enables it to bounce over any mountain to access any valley, given enough bounces. As the temperature declines the ball cannot bounce so high, and it can also settle to become trapped in relatively small ranges of valleys. A generating distribution generates possible valleys or states to be explored. An acceptance distribution is also defined, which depends on the difference between the function value of the present generated valley to be explored and the last saved lowest valley. The acceptance distribution decides probabilistically whether to stay in a new lower valley or to bounce out of it. All the generating and acceptance distributions depend on the temperature.

It has been proved that by carefully controlling the rate of cooling of the temperature, SA can find the global optimum. However, this requires infinite time. Fast annealing and very fast simulated reannealing (VFSR) or adaptive simulated annealing (ASA) are each in turn exponentially faster and overcome this problem.

Chapter 5

Genetic Algorithms

5.1 Introduction to Genetic Algorithms

The goals of creating artificial intelligence and artificial life stem from the very beginnings of the computer age [94]. The earliest computer scientists –Alan Turing, John Von Neumann, Norbert Weiner, and others– were motivated in large part by visions of imbuing computer programs with intelligence, with the life-like ability to self-replicate, and with the adaptive capability to learn and to control their environments. These early pioneers of computer science were as much interested in biology and psychology as in electronics, and looked to natural systems as guiding metaphors for how to achieve their visions. From the earliest days, computers were applied not only to calculating missile trajectories and deciphering military codes, but also to modeling the brain, mimicking human learning, and simulating biological evolution.

GAs were first described by John Holland in the 1960s and further developed by Holland and his students and colleagues at the University of Michigan in the 1960s and 1970s. Holland' s goal was to understand the phenomenon of "adaptation" as it occurs in nature and to develop ways in which the mechanisms of natural adaptation might be imported into computer systems. Holland' s 1975 book *Adaptation in Natural and Artificial Systems* [95] presented the GA as an abstraction of biological evolution and gave a theoretical framework for adaptation under the GA. Holland's GA is a method for moving from one population of "chromosomes" (e.g., strings of "bits" representing candidate solutions to a problem) to a new population, using "selection" together with the genetics-inspired operators of crossover, mutation, and inversion. Each chromosome consists of "genes" (e.g., bits), with each gene being an instance of a particular "allele" (e.g., 0 or 1). The selection operator chooses those chromosomes in the population that will be allowed to reproduce, with fitter chromosomes producing on average more offspring than less fit ones. Crossover exchanges subparts of two chromosomes, roughly mimicking biological recombination between two single-chromosome ("haploid") organisms; mutation randomly changes the allele values of some locations in the chromosome; and inversion reverses the order of a contiguous section of the chromosome, thus rearranging the order in which genes are arrayed.

5.2 The Appeal of Evolution

The mechanisms of evolution seem well-suited for some of the most pressing computational problems in many fields. Many computational problems involve search through a huge number of possibilities for solutions. One example is the problem of computational protein engineering, in which an algorithm is sought that will search among the vast number of possible amino-acid sequences for a protein with specified properties. Another example is searching for a set of rules that will predict the ups and downs of a financial market such as foreign currency. Such search problems can often benefit from an effective use of parallelism, in which many different possibilities are explored simultaneously in an efficient way. Many computational problems require a computer program to be adaptive to continue to perform well in a changing environment. This is typified by problems in robot control in which a robot has to perform a task in a variable environment, or computer interfaces that need to adapt to the idiosyncrasies of an individual user. Other problems require computers to be innovative to construct something truly new and original, such as a new algorithm for accomplishing a computational task, or even a new scientific discovery. Finally, many computational problems require complex solutions that are difficult to program by hand. A striking example is the problem of creating artificial intelligence. Early on, AI practitioners believed that it would be straightforward to encode the rules that would confer intelligence in a program; expert systems are a good example. Nowadays, many AI researchers believe that the “rules” underlying intelligence are too complex for scientists to encode in a “top-down” fashion, and that the best route to artificial intelligence is through a “bottom-up” paradigm. In such a paradigm, human programmers encode simple rules, and complex behaviors such as intelligence emerge from these simple rules. Connectionism (i.e., the study of computer programs inspired by neural systems) is one example of this philosophy [96]; evolutionary computation is another. Biological evolution is an appealing source of inspiration for addressing these problems.

Evolution is a method of searching among an enormous number of possibilities for solutions. In biology, the enormous set of possibilities is the set of possible genetic sequences, and the desired “solutions” are high-fitness organisms organisms able to survive and reproduce in their environments. Thus, the mechanisms of evolution can inspire computational search methods. Of course the fitness of a biological organism depends on many factors for example, how well it can weather the physical characteristics of its environment and how well it can compete with or cooperate with the other organisms around it. The fitness criteria continually change as creatures evolve, so evolution is searching a constantly changing set of possibilities. Searching for solutions in the face of changing conditions is precisely what is required for adaptive computer programs. Furthermore, evolution is a massively parallel search method: rather than working on one species at a time, evolution tests and changes millions of species in parallel. Finally, viewed from a high level the “rules” of evolution are remarkably simple: species evolve by means of random variation (via mutation, recombination, and other operators), followed by natural selection in which the fittest tend to survive and reproduce, thus propagating their genetic material to future generations. Yet these simple rules are responsible, in large part, for the marvelous complexity we see in the biosphere, including human intelligence.

5.3 Elements of Genetic Algorithms

A GA searches through a space of "chromosomes", each of which represents a candidate solution to a given problem (in some cases, a solution consists of a set of chromosomes). Most methods called "GAs" have at least the following elements in common: populations of chromosomes, selection according to fitness, crossover to produce new offspring, and random mutation of new offspring.

The chromosomes in a GA population most often take the form of bit strings (i.e., strings of 1s and 0s); each bit position ("locus") in the chromosome has two possible values ("alleles"), 0 and 1. These biological terms are used in the spirit of analogy with real biology, though the entities they refer to are, of course, much simpler than the real biological ones. The search takes place by processing populations of chromosomes, changing from one such population to another. The GA most often requires a "fitness function" that assigns a score (fitness) to each chromosome in the current population. The fitness of the chromosome depends on how well that chromosome solves the problem at hand.

For example, a common application of GAs is function optimization, where the goal is to find a set of parameter values that maximizes, say, a complex multiparameter function. As a simple example, one might want to maximize the real-valued one-dimensional function

$$f(x) = x + |sin(32x)| \tag{5.1}$$

over all values of x between 0 and π [97]. Here the candidate solutions are values of x, which can be encoded as bit strings representing real numbers. The fitness calculation translates a given bit string into a real number x and then evaluates the function at that value. The fitness of a string is the function value at that point.

The simplest form of GA involves three types of operators:

- Selection: This operator selects chromosomes in the population for reproduction. The fitter the chromosome, the more times it is likely to be selected to reproduce.
- Crossover: This operator exchanges subsequences of two chromosomes to create two offspring. For example, the strings

  $$10000100 \; and \; 11111111 \tag{5.2}$$

  could be crossed over after the third locus in each to produce the two offspring

  $$10011111 \; and \; 11100100 \tag{5.3}$$

  This operator roughly mimics biological recombination between two single-chromosome (haploid) organisms. Most higher organisms have chromosomes in pairs and are thus "diploid".
- Mutation: This operator randomly flips some bits in a chromosome. For example, the string 00000100 might be mutated in its second position to yield 01000100. Mutation can occur at each bit position in a string with some probability, usually very small (e.g., 0.001).

5.4 A Simple Genetic Algorithm

Given a clearly defined problem to be solved and a bit-string representation for candidate solutions, the simple GA works as follows: Each iteration of this process is called a "generation". A GA is

Algorithm 3 A Simple Genetic Algorithm

```
1: Start with a randomly generated population of N L-bit chromosomes (candidate solutions to a problem).
2: Calculate the fitness F(x) of each chromosome x in the population.
3: for Repeat the following steps (a)–(c) until N offspring have been created: do
4:    (a) Select a pair of parent chromosomes from the current population, with the probability of selection being
      an increasing function of fitness. Selection is done "with replacement", meaning that the same chromosome
      can be selected more than once to become a parent.
5:    (b) With probability pc (the crossover probability), cross over the pair at a randomly chosen point (chosen
      with uniform probability) to form two offspring. If no crossover takes place, form two offspring that are exact
      copies of their respective parents.
6:    (c) Mutate the two offspring at each locus with probability pm (the mutation probability), and place the
      resulting chromosomes in the new population.
7: end for
8: Replace the current population with the new population.
9: Go to step 2.
```

typically iterated for anywhere from 50 to 500 or more generations, which is called a "run". At the end of a run, there are often one or more highly fit chromosomes in the population. Since randomness plays a large role in each run, two runs with different random number seeds will generally produce different detailed behavior. GA researchers often report statistics (such as the best fitness found and generation at which best fitness was found) averaged over many different runs of the GA on the same problem.

This simple procedure is the basis for most applications of GAs. There are a number of details to fill in, such as the size of the population and the probabilities of crossover and mutation, and the success of the algorithm often depends greatly on these details. There are also far more complicated versions of GAs, for example, GAs that work on representations other than bit strings or GAs that have different types of crossover and mutation operators.

5.5 Some Applications of Genetic Algorithms

The algorithm described above is very simple, but variations on this basic theme have been used in a large number of scientific and engineering problems and models, including the following:

- Optimization: GAs have been used in a wide variety of optimization tasks, including numerical optimization as well as combinatorial optimization problems such as circuit layout and job-shop scheduling.
- Automatic Programming: GAs have been used to evolve computer programs for specific tasks, and to design other computational structures, such as cellular automata and sorting networks.

- Machine learning: GAs have been used for many machine-learning applications, including classification and prediction tasks such as the prediction of weather or protein structure. GAs have also been used to evolve aspects of particular machine-learning systems, such as weights for neural networks, rules for learning classifier systems or symbolic production systems, and sensors for robots.
- Economic models: GAs have been used to model processes of innovation, the development of bidding strategies, and the emergence of economic markets.
- Immune system models: GAs have been used to model various aspects of the natural immune system including somatic mutation during an individual's lifetime and the discovery of multi-gene families during evolutionary time.
- Ecological models: GAs have been used to model ecological phenomena such as biological arms races, host-parasite co-evolution, symbiosis, and resource flow in ecologies.
- Population genetics models: GAs have been used to study questions in population genetics, such as "Under what conditions will a gene for recombination be evolutionarily viable?"
- Interactions between evolution and learning: GAs have been used to study how individual learning and species evolution affect one another.
- Models of social systems: GAs have been used to study evolutionary aspects of social systems, such as the evolution of cooperation, the evolution of communication, and trail-following behavior in ants.

Chapter 6

Simulation Systems Implementation

6.1 Introduction to Mesh Router Node Placement Optimization

WMNs [98] have become an important networking infrastructure due to their low cost and increased high speed wireless Internet connectivity. In a WMN there are two types of nodes: mesh routers and mesh clients. Mesh routers are similar to normal routers but incorporate also additional functions to support mesh networking, and are usually equipped with multiple interfaces to work with different wireless technologies. Another feature of this type of routers with respect to usual routers is their ability to provide the same coverage with much less transmitter power through multi-hop communications. Also, mesh routers can be installed on a dedicated machine or on a general purpose machine. On the other hand, mesh clients have the necessary functions for mesh networking and could also be able to act as routers but do not have the functionality of a gateway or bridge and their single wireless interface with the hardware and software platform is much simpler than in the case of mesh routers. WMNs are based on mesh topology, in which every node (representing a server) is connected to one or more nodes, enabling thus the information transmission in more than one path. The path redundancy is a robust feature of this kind of topology. Compared to other topologies, mesh topology needs not a central node, allowing networks based on such topology to be self-healing. These characteristics of networks with mesh topology make them very reliable and robust networks to potential server node failures.

Fast development of WMNs is pushed by their low cost nature that makes them an economical alternative for providing wireless Internet connectivity, especially in developing countries, avoiding costs of deployment and maintenance of wired Internet infrastructures. Applications of WMNs include WMNs for urban areas, community networking, metropolitan area networks, municipal wireless mesh networks, corporative networks, medical systems, transport systems, surveillance systems, etc. [99]. In all these applications, WMNs provide cost-efficient broadband wireless Internet connectivity to a group of users. As a case study of the use of such networks we can highlight the “One hundred dollar laptop” project developed at MIT for schools in developing countries. The objective of the project is to establish a mesh network to create a robust and inexpensive infrastructure from students’ laptops. The connections made by the laptops would not need an internal infrastructure, like Internet, since one of the computers on the grid may share the connection with the neighboring nodes.

Several optimization problems are showing their usefulness to the efficient design of WMNs. These problem are related, among others, to optimizing network connectivity, user coverage and stability. The resolution of these problems turns out to be crucial for optimized network performance. Most important optimization problems in WMNs deal with computing optimal placement of nodes (mesh router nodes, gateways and distribution of mesh client notes), so that network performance is optimized. However, node placement problems are known for their hardness to solve to optimality and therefore heuristics methods are used to near-optimally solve such problems. In this work we present some optimization problems in WMNs and different heuristic methods such as GA, TS, HC, SA for solving them near-optimally. We formulate the optimization problems using multi-objective optimization models. Thus, for the mesh router nodes placement, the bi-objective optimization problem is obtained consisting in the maximization of the size of the giant component in the mesh routers network (for measuring network connectivity) and that of user coverage.

Purely random placements would produce poor performance due to far from optimal router placement as a result. Therefore, using more efficient methods is crucial for node placement nodes in WMNs. Due to computational intractability of the problem, exact methods can only solve to optimality small size instances, and therefore heuristic and meta-heuristic approaches are the de facto approach to solve the problem for practical purposes.

6.2 Node Placement Problems and Their Applicability to WMNs

Node placement problems have been long investigated in the optimization field due to numerous applications in location science (facility location, logistics, services, etc.) and classification (clustering). In such problems, we are given a number of potential facilities to serve to costumers connected to facilities aiming to find locations such that the cost of serving to all customers is minimized. In traditional versions of the problem, facilities could be hospitals, polling centers, fire stations serving to a number of clients and aiming to minimize some distance function in a metric space between clients and such facilities. One classical version of the problem is that of p-median problem, defined as follows.

Definition 1: Given a set $\mathscr{F}$ of m potential facilities, a set $\mathscr{U}$ of n users, a distance function $d : \mathscr{U} \to \mathscr{F}$, and a constant $p \leq m$, determine which p facilities to open so as to minimize the sum of the distances from each user to its closest open facility.

The problem, which is known for its intractability, has many application not only in location science but also in communication networks, where facilities could be servers, routers, etc., offering connectivity services to clients. In WMNs mesh routers provide network connectivity services to mesh client nodes. The good performance and operability of WMNs largely depends on placement of mesh routers nodes in the geographical deployment area to achieve network connectivity, stability and user coverage. The objective is to find an optimal and robust topology of the mesh router nodes network to support connectivity services to clients.

Facility location problems are thus showing their usefulness to communication networks, and more especially from Wireless Mesh Networks (WMNs) field [98, 100, 101]. In a general setting, location models in the literature have been defined as follows. We are given:

(a) a universe $\mathscr{U}$, from which a set $\mathscr{C}$ of mesh client input positions is selected;
(b) an integer, $\mathscr{N} \geq 1$, denoting the number of facilities to be deployed;
(c) one or more metrics of the type $d : \mathscr{U} \times \mathscr{U} \rightarrow \mathscr{R}_+$, which measure the quality of the location; and,
(d) an optimization model.

The optimization model takes in input the universe where facilities are to be deployed, a set of client positions and returns a set of positions for facilities that optimize the considered metrics. It should be noted that different models can be established depending on whether the universe is considered: (a) continuous (universe is a region, where clients and facilities may be placed anywhere within the continuum leading to an uncountably infinite number of possible locations); (b) discrete (universe is a discrete set of predefined positions); and, (c) network (universe is given by an undirected weighted graph; in the graph, client positions are given by the vertices and facilities may be located anywhere on the graph). For most formulations, node placement problems are shown to be computationally hard to solve to optimality [21, 22, 100, 102] and therefore heuristic and meta-heuristic approaches are useful approaches to solve the problem for practical purposes.

6.3 Optimization Problems

Different optimization problems can be formulated based on the objectives to optimize and a set of different constraints, such as topological restrictions, battery restrictions, QoS requirements, etc. Some optimization problems are related to minimize the cost of the WMN, such as minimizing the number of mesh router nodes to deploy, while others focus on the WMN performance, such as computing optimal placement of an a priori fixed number of mesh router nodes. The presence of many objectives is in fact a main challenge. These objectives include minimizing the number of mesh routers, maximizing network connectivity, maximizing user coverage, minimizing energy consumption (especially in wireless and mobile networks), minimizing communication delay, maximizing throughput, minimizing deployment cost, etc. And, additionally, there could be certain constraints to take into account such as topological restrictions of the geographical area, interference model, etc. It should also be noted that some of the objectives are contradicting, in the sense that trying to optimize some objective goes in detriment to the optimization of another objective.

A. Minimizing the number of mesh routers in WMN

This problem aims to minimize the number of mesh router nodes under the restrictions of achieving network connectivity and client coverage [22]. Reducing the number of mesh routers has a direct implication for cost reduction of WMN.

Definition 2: Given a network graph $G = (V, E)$ of candidate mesh router nodes V, and a set of gateway nodes GW, find a subgraph $V' \subseteq V$ of mesh router nodes such that every mesh router from V' is connected to a gateway node $g \in GW$ by at least one path and V' network provides full coverage to local domain. The problem can be seen as a variation of p-median problem (see Definition 1). Other variations of the problem can be obtained by considering that the gateway nodes are a priori fixed in the geographical area. The complexity of the problem, which *per se* is *Np-hard*, is increased if the set of gateway nodes is to be placed in the geographical area so as to optimize traffic effort. Additionally, one may consider to optimally assign mesh router nodes to gateway nodes.

B. Gateway node placement

In the Gateway Node Placement (GNP) problem, the objective is to find a placement of gateway nodes such that several parameters of the WMN are optimized. Depending on the parameters to be optimized and the QoS constraints for the WMN, different versions of the problem can be formulated.

Definition 3: Given a WMN graph, a number k of gateways to deploy, a set of parameters $\mathscr{P}$ to optimize and a set of constraints $\mathscr{C}$ to satisfy, find a placement of the k gateway nodes in the WMN so that the parameters in $\mathscr{P}$ are optimized under constraints $\mathscr{C}$.

Optimizing communication delay and cost: Wong et al. [103] formulated two versions of GNP so that communication delay and communication cost are optimized.

Maximizing the minimum flow throughput: Muthaiah et al. [104] formulated the GNP on single gateway multi-hop WMNs. Similarly, Li et al. [105] considered the gateway placement in order to optimize the throughput in WMNs.

C. Mesh router nodes placement

In this problem, we are given a grid area arranged in cells where to distribute a number of mesh router nodes and a number of mesh client nodes of fixed positions (of an arbitrary distribution) in the grid area. The objective is to find a location assignment for the mesh routers to the cells of the grid area that maximizes the network connectivity and client coverage. Network connectivity is measured by the size of the giant component of the resulting WMN graph, while the user coverage is simply the number of mesh client nodes that fall within the radio coverage of at least one mesh router node.

An instance of the problem consists as follows.

- N mesh router nodes, each having its own radio coverage, defining thus a vector of routers.
- An area $W \times H$ where to distribute N mesh routers. Positions of mesh routers are not pre-determined, and are to be computed.
- M client mesh nodes located in arbitrary points of the considered area, defining a matrix of clients.

It should be noted that network connectivity and user coverage are among most important metrics in WMNs and directly affect the network performance. Nonetheless, network connectivity is usually considered as more important than user coverage.

Notice from the above definition that mesh client nodes can be arbitrarily situated in the given area. For evaluation purposes, it is, however, interesting to consider concrete distributions of mesh client nodes such as Uniform, Normal, Exponential and Weibull distributions.

In fact, we can formalize an instance of the problem by constructing an adjacency matrix of the WMN graph, whose nodes are router nodes and client nodes and whose edges are links between nodes in the mesh network. Each mesh node in the graph is a triple $v =< x, y, r >$ representing the 2D location point and r is the radius of the transmission range. There is an arc between two nodes u and v, if v is within the transmission circular area of u. It should be noticed here that the deployment grid area is partitioned by cells, representing graph nodes, where we can locate mesh router nodes. We assume that in a cell, both a mesh router node and a mesh client node can be placed.

The objective is to place mesh router nodes in cells of considered area to maximize network connectivity and user coverage. Network connectivity and user coverage are among most important metrics in WMNs. The former measures the degree of connectivity of the mesh nodes while the later refers to the number of mesh client nodes connected to the WMN. Both objectives are crucial to WMN and directly affect the network performance; nonetheless, network connectivity is considered as more important than user coverage.

D. Multi-objective optimization model

For optimization problems having two or more objective functions, two models are usually considered: the hierarchical and simultaneous optimization.

Hierarchic model: In this model, the objectives are classified (sorted) according to their priority. Thus, for a problem having k objectives sorted as follows

$$f_1 \succ f_2 \succ \cdots \succ f_k \tag{6.1}$$

means that f_1 is the most important objective and f_k is the least important objective. The optimization procedure would first optimize according to f_1 until no further improvements are possible. Then, the algorithm optimizes according to f_2 subject to not worsening the value achieved for f_1, and so on. For the two objective case, $f_1 \succ f_2, f_1$, is considered as primary objective and f_2 as secondary one. This model is useful when for design or deployment needs some parameters (objectives) are considered of more priority than others. For instance, network connectivity, which is crucial to WMN design, could be considered a primary objective and the user coverage a secondary one. It should however be noted that the final solution computed by the optimization procedure need not to be optimal and could be far from optimal for the less priority objectives.

Simultaneous model: In the simultaneous approach, all objectives are simultaneously optimized. Thus, for a problem having k objectives $f_1, f_2, ..., f_k$, the optimization procedure tries to optimize at the same time all the objectives, which actually leads to computing the so called Pareto front which contains the optimal solutions. In some cases, it is possible to apply the sum model in which the k objectives are reduced to two objectives:

$$f = \lambda_1 f_1 + \lambda_2 f_2 + \cdots + \lambda_k f_k, \sum_{i=1}^{i=k} \lambda_i = 1, \lambda_i > 0. \tag{6.2}$$

E. Mesh client nodes distributions

An important issue when formulating mesh node placement is whether the client nodes are stationary or mobile nodes. WMN with stationary nodes arise in many real situations, for instance, in a neighboring community. In stationary nodes case, the positions of the clients are a priori known although the mesh client nodes can be arbitrarily situated in the given area. In the later case of WMN with mobile nodes, the position of client nodes can change over time. It could as well be considered the case of a WMN where we have both stationary and mobile nodes, for instance, in a neighborhood users inside the homes are stationary and users along the roads are mobile. In both cases, it is interesting, however, to consider concrete distributions of client mesh nodes. For instance, it has been shown from studies in real urban areas or university campuses that users tend to cluster to hotspots. The Uniform, Normal, Exponential and Weibull distributions for client mesh nodes can be considered.

Several heuristic approaches are found in the literature for node placement problems in WMNs [18–20, 104, 106]. Given the complexity of node placement problems, most authors have proposed the use of simple heuristic methods or more advanced search methods such as Genetic Algorithms. In the following subsections we present different resolution methods, including exact and heuristic methods for solving node placement problems in WMNs. We exemplify the applicability of heuristic methods for the case of solving mesh router nodes problem. We have considered methods from two different families, namely, local search methods (HC, SA and TS) and population-based methods (GAs). The former are known for their capability to exploit the solution space by constructing a path of visited solutions, while the later methods use a population of individuals aiming to largely explore the solution space.

6.4 Ad hoc Methods for Mesh Router Nodes Placement

Ad hoc methods for placement of mesh routers are simple methods that explore different possible placement topologies. Their usefulness is two fold: (a) enabling fast computation of simple solutions for mesh router nodes placement, and (b) initializing other more sophisticated methods such as evolutionary algorithms that use population of solutions. We have consider seven ad hoc methods, namely: 1) Random, 2) ColLeft, 3) Diagonal, 4) Cross, 5) Near, 6) Corners, and 7) HotSpot. We briefly describe them next. It should be noted that in all considered methods, there is a pattern in placement of mesh router nodes, meaning that most of the node placements follow the pattern.

- *Random placement*: In this method, mesh router nodes are uniformly at random distributed in the grid area.
- *ColLeft placement*: This method places almost all mesh routers at the left side of the grid area. Some mesh routers could be placed at other parts of the grid area. The method is usually applicable when the number of mesh routers is (proportionally) smaller than grid area height, for instance, one third of the height.

- *Diagonal placement*: In this method, mesh routers are concentrated along the (main) diagonal of the grid area. Again, this method is appropriate when the grid area fulfils some conditions such as the height and width must have similar values (we considered the case of 10 [%] difference in their values) so that we can trace the diagonals.
- *Cross placement*: This method tends to place mesh routers along both diagonals of the grid area. Similar conditions as the ones for Diagonal placement are required to ensure applicability of the method.
- *Near placement*: In this method mesh routers are concentrated in the central zone of the grid area. To apply the method, minimum and maximum (user specified) values are considered to trace a rectangle in the central part of the grid area; routers are distributed in the rectangle cells.
- *Corners placement*: This method distributes the mesh routers in the corners of the grid area. The considered areas in the corners are fixed by user specified parameter values.
- *HotSpot placement*: This method starts by placing the most powerful mesh router in the most dense zone (in terms of client nodes) of the grid area; next, the second most powerful mesh router is placed in the second most dense zone, and so on until all routers are placed. This method seems particularly suited when distribution of mesh clients is not known a priori. It should be noted however that this method has a greater computational cost as compared to other methods due to the computation of denseness property.

6.5 Neighborhood Search-based Algorithms

The ad hoc methods presented in the previous section explore simple topologies for the placement of the mesh router nodes in the grid area. However, their solutions could be far from optimal ones. Algorithms based on neighborhood exploration are simple yet more powerful than ad hoc methods. The main idea is exploring the neighborhood of an initial solution by means of local moves and iterate until a stopping condition is met (for instance, there is no further improvement on the quality of solution.)

Different neighborhood search algorithms can be obtained depending on neighborhood structure, the acceptance criteria of next solution, etc. We present the pseudo-algorithm of neighborhood search in Algorithm 4. Starting from an initial solution, the algorithm first selects a movement type, that is the way the small local perturbation is performed, which defines the neighborhood structure. Then, iteratively, the algorithm computes the best neighbor of the current solution, namely, the best solution in the neighborhood of the current solution. If the best neighbor improves fitness of current solution, the current solution is moved to the best neighbor and so on.

The computation of the best neighboring solution is presented in Algorithm 5. It should be noted that the exploration of the neighborhood can be done in different ways. For instance, we can systematically generate all movements, and hence, examine all possible solutions in the neighborhood; or, in case of large neighborhoods, just a pre-fixed number of movements is generated and corresponding neighboring solutions are examined.

Algorithm 4 Neighborhood search based algorithm for maximization. f is the fitness function.

Generate an initial solution s_0 using an ad hoc method;
$s = s_0$; $s^* = s_0$; $f^* = f(s_0)$;
Choose a movement type;
repeat
  $s' = BestNeighbor(s)$;
  if $f(s') \leq f(s)$ **then**
    $s^* = s$
    $f^* = f(s)$
    return s^*, f^*
  end if
  $s = s'$;
until (stopping condition is met)

Algorithm 5 Best neighbor selection for maximization

s is the current solution.
t is a movement type.
$best\_neighbor\_sol = s$;
repeat
  Apply movement m to solution s; $s' = m(s)$;
  if $f(s') \geq f(s)$ **then**
    $best\_neighbor\_sol = s'$;
  end if
until (all or a pre-fixed number of neighbor solutions are examined)
return $best\_neighbor\_sol$;

6.6 WMN-GA System

In this section, we present WMN-GA system. Our system can generate instances of the problem using different distributions of client and mesh routers.

For the network configuration, we use: distribution, number of clients, number of mesh routers, grid size, radius of transmission distance and the size of subgrid.

For the GA parameter configuration, we use: number of independent runs, GA evolution steps, population size, population intermediate size, crossover probability, mutation probability, initial methods, select method.

GAs have shown their usefulness for the resolution of many computationally hard combinatorial optimization problems. They are, of course, a strong candidate for efficiently solving mesh router nodes placement problem in WMNs. For the purpose of this work we have used the *template* given in Algorithm 6.

As can be seen from the template, several parameters intervene in the GAs: population size, intermediate population size, number of evolution steps, crossover probability, mutation probability and parameters for replacement strategies. On the other hand, there are the (families of) genetic operators: crossover operators, mutation operators, selection operators and replacement operators. As there are potentially large range values for parameters and different versions of operators, their tuning becomes crucial to the GA's performance.

Algorithm 6 : Pseudo-code of WMN-GA

Generate the initial population P^0 of size μ; $t = 0$.
Evaluate P^0;
while not termination-condition **do**
  Select the parental pool T^t of size λ;
  $T^t := Select(P^t)$;
  Perform crossover procedure on pairs of individuals in T^t with probability p_c; $P^t_c := Cross(T^t)$;
  Perform mutation procedure on individuals in P^t_c with probability p_m; $P^t_m := Mutate(P^t_c)$;
  Evaluate P^t_m ;
  Create a new population P^{t+1} of size μ from individuals in P^t and/or P^t_m ;
  $P^{t+1} := Replace(P^t; P^t_m)$
  $t := t + 1$;
end while
return Best found individual as solution;

We explain in details the GA operations in following.

■Encoding The encoding of individuals (also known as chromosome encoding) is fundamental to the implementation of GAs in order to efficiently transmit the genetic information from parents to offsprings.

In the case of the mesh router nodes placement problem, a solution (individual of the population) contains the information on the current location of routers in the grid area as well as information on links to other mesh router nodes and mesh client nodes. This information is kept in data structures, namely, `pos_routers` for positions of mesh router nodes, `routers_links` for link information among routers and `client_router_link` for link information among routers and clients. Based on these data structures, the size of the giant component and the number of users covered are computed for the solution.

It should be also noted that routers are assumed to have different radio coverage, therefore to any router could be linked a number of clients and other routers. Obviously, whenever a router is moved to another cell of the grid area, the information on links to both other routers and clients must be computed again.

■Fitness Evaluation The fitness function is of particular importance in GAs as it guides the search towards most promising areas of the solution space. Furthermore, in our case, we face an optimization problem with multiple criteria, including size of giant component, number of users covered, minimization of number of routers to deploy, and so on.

■Selection Operators In the evolutionary computing literature we can find a variety of selection operators, which are in charge of selecting individuals for the pool mate [107]. The operators considered in this work are those based on *Implicit Fitness Re-mapping* technique. It should be noted that selection operators are generic ones and do not depend on the encoding of individuals.

- *Random Selection*: This operator chooses the individuals uniformly at random. The problem is that a simple strategy does not consider even the fitness value of individuals and this may lead to a slow convergence of the algorithm.

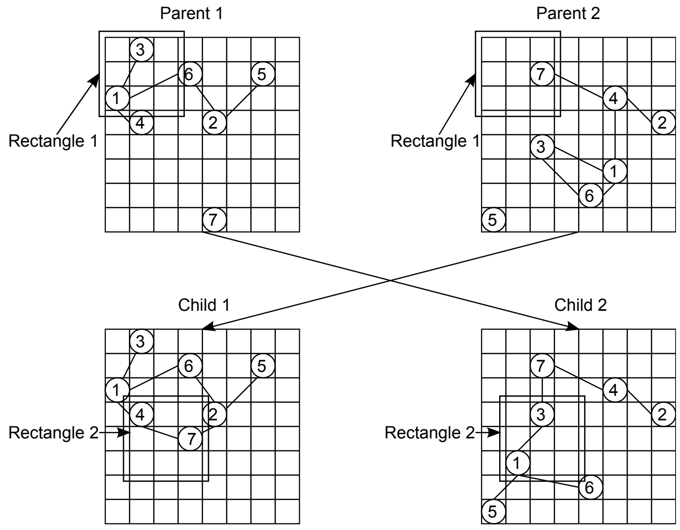


Fig. 6.1 Cross Region Crossover.

- *Best Selection*: This operator selects the individuals in the population having higher fitness value. The main drawback of this operator is that by always choosing the best fitted individuals of the population, the GA converges prematurely.
- *Linear Ranking Selection*: This operator follows the strategy of selecting the individuals in the population with a probability directly proportional to its fitness value. This operator clearly benefits the selection of best endowed individuals, which have larger chances of being selected. We used this operator for our simulations.
- *Exponential Ranking Selection*: This operator is similar to Linear Ranking but now probabilities of ranked individuals are weighted according to an exponential distribution.
- *Tournament Selection*: This operator selects the individuals based on the result of a tournament among individuals. Usually winning solutions are the ones of better fitness value but individuals of worse fitness value could be chosen as well, contributing thus to avoiding premature convergence. Particular cases of this operator are the *Binary Tournament* and $N-$*Tournament Selection*, for different values of N.

■Crossover Operators The crossover operators are the most important ingredient of GAs. Indeed, by selecting individuals from the parental generation and interchanging their *genes*, new individuals (descendants) are obtained. The aim is to obtain descendants of better quality that will feed the next generation and enable the search to explore new regions of solution space not explored yet.

There exist many types of crossover operators explored in the evolutionary computing literature. It is very important to stress that crossover operators depend on the chromosome representation. This observation is especially important for the mesh router nodes problem, since in our case, instead of having strings we have a grid of nodes located in a certain positions. The crossover operator should thus take into account the specifics of mesh router nodes encoding. We have considered the following crossover operator, called *intersection operators* (denoted `CrossRegion`, hereafter), which take in input two individuals and produce in output two new individuals (see Algorithm 7).

Algorithm 7 Crossover Operator

Input: Two parent individuals P_1 and P_2; values H_g and W_g for height and width of a small grid area;
Output: Two offsprings O_1 and O_2;
Select at random a $H_g \times W_g$ rectangle RP_1 in parent P_1. Let RP_2 be the same rectangle in parent P_2;
Select at random a $H_g \times W_g$ rectangle RO_1 in offspring O_1. Let RO_2 be the same rectangle in offspring O_2;
Interchange the mesh router nodes: Move the mesh router nodes of RP_1 to RO_2 and those of RP_2 to RO_1;
Re-establish mesh nodes network connections in O_1 and O_2 (links between mesh router nodes and links between client mesh nodes and mesh router nodes are computed again);
return O_1 and O_2;

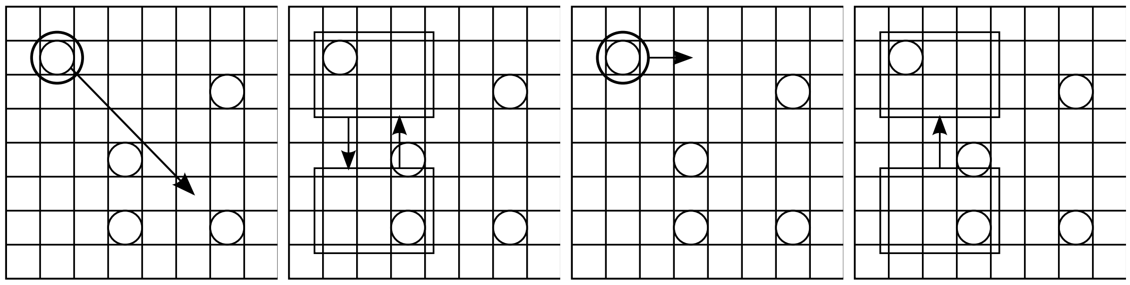
■Mutation Operators Mutation operator is one of the GA ingredients. Unlike crossover operators, which achieve to transmit genetic information from parents to offsprings, mutation operators usually make some small local perturbation of the individuals, having thus less impact on newly generated individuals.

Crossover is "a must" operator in GA and is usually applied with high probability, while mutation operators when implemented are applied with small probability. The rationale is that a large mutation rate would make the GA search to resemble a random search. Due to this, mutation operator is usually considered as a secondary operator.

In the case of mesh routers node placement, the matrix representation is chosen for the individuals of the population, in order to keep the information on mesh router nodes positions, mesh client positions, links among routers and links among routers and clients. The definition of the mutation operators is therefore specific to matrix-based encoding of the individuals of the population. Several specific mutation operators were considered in this study, which are move-based and swap-based operators.

- *SingleMutate*: This is a move-based operator. It selects a mesh router node in the grid area and moves it to another cell of the grid area. (see Fig. 6.2(a)).
- *RectangleMutate*: This is a swap-based operator. In this version, the operator selects two "small" rectangles at random in the grid area, and swaps the mesh routers nodes in them. (see Fig. 6.2(b)).
- *SmallMutate*: This is a move-based operator. In this case, the operator chooses randomly a router and moves it a small (*a priori* fixed) number of cells in one of the four directions: up, down, left or right in the grid (see Fig. 6.2(c)). This operator could be used a number of times to achieve the effect of SingleMutate operator.
- *SmallRectangleMutate*: This is a move-based operator. The operator selects first at random a rectangle and then all routers inside the rectangle are moved with a small (*a priori* fixed) numbers of cells in one of the four directions: up, down, left or right in the grid. (see Fig. 6.2(d)).

■Time Efficiency of Mutation and Selection Operators Simple mutation operators, such as bit-flip or simple move are very efficient as only small changes are done in the combinatorial structure. However, mutation based on making larger perturbations of the individual could be time costly given they are applied a total expected number of $p_m \cdot population\_size$, where p_m denotes the



(a) SingleMutate (b) RectangleMutate (c) SmallMutate (d) SmallRectangleMutate

Fig. 6.2 Mutation operators.

mutation probability. This amount is to be multiplied by the time cost of performing one mutation, which in case of rectangle mutate and small rectangle mutate are $O(grid\_side \times grid\_side)$. The computational effort is even larger for the crossover operators as in this case the grid area has to be explored to find the sparsest and densest areas.

6.7 WMN-TS System

TS method was introduced by Glover [108] as a high-level algorithm that uses other specific heuristics to guide the search; the objective is to perform an intelligent exploration of the search space that would eventually allow to avoid getting trapped into local optima. The objective is thus to remedy one of the main issues of local search methods, namely the useless search in neighborhood of local optima without further improvements due to re-visiting solutions or paths of solutions already explored. This is achieved by giving the tabu status to solutions visited in the recent search. TS is also designed to be a flexible method, so that the tabu status of solutions can be waived, in case they have been prohibited for a long while or if they satisfy some aspiration criteria. The classification of some solutions as tabu is achieved through the intelligent use of adaptive memory, which is allowed to evolve and eventually change the status of tabu solutions. The main features of the TS method are that of adaptive memory and responsive exploration. Again, the adaptive memory is the basis to guide the search in taking intelligent decisions. This gives the TS method advantages with regard to other memoryless methods, being these local search methods (HC, SA, etc.) or population based methods (GAs, Memetic Algorithms, etc.). On the other hand, the responsive exploration enables the method to select some solutions which though not so good at the current search iteration might at long run lead to promising areas of good solutions in the search space (see Algorithm 8).

As it can be seen in the template of Algorithm 8, the inner components to be specified for designing a TS algorithm for a concrete problem are the following:

- *Local search*: It consists in exploring the neighborhood of a solution and can be implemented. It should be noted, however, that in the TS method the neighborhood has a dynamic structure since some solutions could be (temporarily) forbiddenduring the search process.

Algorithm 8 : Pseudo-code of WMN-TS

```
begin
Compute an initial solution s;
let ŝ ← s;
Reset the tabu and aspiration conditions;
while not termination-condition do
  Generate a subset N*(s) ⊆ N(s) of solutions such that:
  (none of the tabu conditions is violated) or (the aspiration criteria hold)
  Choose the best s′ ∈ N*(s) with respect to the cost function;
  ŝ ← s′ ;
  if improvement(s′ , ŝ) then
    ŝ ← s′ ;
  end if
  Update the recency and frequency;
  if (intensification condition) then
    Perform intensification procedure;
  end if
  if (diversification condition) then
    Perform diversification procedures;
  end if
end while
return ŝ;
end
```

- *Historical memory*: This is one of the main distinguishing characteristics of TS with respect to other local search and population-based methods. This historical memory is usually composed by a short term memory (or recency), with information on recently visited solutions, and a long term memory (or frequency), storing information gathered during the whole exploration process.
- *Tabu status and aspiration criteria*: Recently visited solutions are given a tabu status meaning that they cannot be considered as candidate solutions during the exploration process. This is called the short term memory. The intended objective of the use of tabu solutions is to avoid cycles in the search. Tabu status of solutions could be cancelled in case some sort of conditions, called aspiration criteria, hold.
- *Intensification and diversification procedures*: Used for appropriately managing the exploration/exploitation tradeoff on the search space.

6.8 WMN-HC System

We proposed and implemented a new simulator that uses Hill Climbing algorithm to solve the problem of node placement in WMNs. We called this simulator WMN-HC. Our system can generate instances of the problem using different distributions of client and mesh routers.

We present here the particularization of the Hill Climbing algorithm (see Algorithm 9) for the mesh router node placement problem in WMNs.

Algorithm 9 : Pseudo-code of WMN-HC.

```
Start: Generate an initial solution s_0;
s = s_0; s* = s_0; f* = f(s_0);
repeat
  Movement Selection: Choose a movement m = select_movement(s);
  Evaluate & Apply Movement:
  if δ(s, m) ≥ 0 then
    s' = appply(m, s);
    s = s';
  end if
  Update Best Solution:
  if f(s') > f(s*) then
    f* = f(s');
    s* = s';
  end if
  Return s*, f*;
until (stopping condition is met)
```

■**Initial solution** The algorithms starts by generating an initial solution either random or by *ad hoc* methods [28].

■**Evaluation of fitness function** An important aspect is the determination of an appropriate objective function and its encoding. In our case, the fitness function follows a hierarchical approach in which the main objective is to maximize the size of giant component in WMN.

■**Neighbor selection and movement types** The neighborhood $N(s)$ of a solution s consists of all solutions that are accessible by a local move from s. We have considered three different types of movements. The first, called *Random*, consists in choosing a router at random in the grid area and placing it in a new position at random. The second move, called *Radius*, chooses the router of the largest radio and places it at the center of the most densely populated area of client mesh nodes (see Algorithm 10). Finally, the third move, called *Swap*, consists in swapping two routers: the one of the smallest radio situated in the most densely populated area of client mesh nodes with that of largest radio situated in the least densely populated area of client mesh nodes. The aim is that largest radio routers should serve to more clients by placing them in more dense areas.

We also considered the possibility to combine the above movements in sequences of movements. The idea is to see if the combination of these movements offers some improvement over the best of them alone. We called this type of movement *Combination*:

$< Rand_1, \ldots, Rand_k; Radius_1, \ldots, Radius_k; Swap_1, \ldots, Swap_k >$,

where k is a user specified parameter.

■**Acceptability criteria** The acceptability criteria for newly generated solution can be done in different ways (simple ascent, steepest ascent, or stochastic). In our case, we have adopted the simple ascent, that is, if s is current solution and m is a movement, the resulting solution s' obtained by applying m to s will be accepted, and hence become current solution, iff the fitness of s' is at least as good as fitness of solution s. In terms of δ function, s' is accepted and becomes current solution if $\delta(s, m) \geq 0$. It should be noted that in this definition we are also accepting solutions that have the same fitness as previous solution. The aim is to give chances to the search to move towards better solutions in solution space. A more strict version would be to accept only solutions that strictly improve the fitness function ($\delta(s, m) > 0$).

Algorithm 10 Radius movement.

1: **Input**: Values H_g and W_g for height and width of a small grid area.
2: **Output**: New configuration of mesh nodes network.
3: Compute the most dense $H_g \times W_g$ area and (x_{dense}, y_{dense}) its central cell point.
4: Compute the position of the router of largest radio coverage $(x_{largest\_cov}, y_{largest\_cov})$.
5: Move router at $(x_{largest\_cov}, y_{largest\_cov})$ to new position (x_{dense}, y_{dense}).
6: Re-establish mesh nodes network connections.

Movement type can be defined in different ways. We considered the swap movement that consists in exchanging the placement of two routers. More precisely, the worst router (that of smallest radio coverage) in the most dense area is exchanged with the best router (that of largest radio coverage) of the sparsest area (see steps in Algorithm 11). The idea is to promote the placement of best routers in most dense areas of the grid area.

Algorithm 11 Swap movement

1: Choose values H_g and W_g for height and width of a small grid area.
2: Choose threshold values for "dense" and "sparse" grid area of size $H_g \times W_g$.
3: Compute the position of most dense $H_g \times W_g$ area.
4: Compute the position (x_{dense}, y_{dense}) of less powerful router within the dense area.
5: Compute the position of most sparse $H_g \times W_g$ area.
6: Compute the position (x_{sparse}, y_{sparse}) of most powerful router within the sparse area.
7: Swap routers in (x_{dense}, y_{dense}) and (x_{sparse}, y_{sparse}) positions.
8: Re-establish mesh nodes network connections.

6.9 WMN-SA System

The SA algorithm [109] is a generalization of the metropolis heuristic. Indeed, SA consists of a sequence of executions of metropolis with a progressive decrement of the temperature starting from a rather high temperature, where almost any move is accepted, to a low temperature, where the search resembles Hill Climbing. In fact, it can be seen as a hill-climber with an internal mechanism to escape local optima (see pseudo-code in Algorithm 12). In SA, the solution s' is accepted as the new current solution if $\delta \leq 0$ holds, where $\delta = f(s') - f(s)$. To allow escaping from a local optimum, the movements that increase the energy function are accepted with a decreasing probability $\exp(-\delta/T)$ if $\delta > 0$, where T is a parameter called the "temperature". The decreasing values of T are controlled by a *cooling schedule*, which specifies the temperature values at each stage of the algorithm, what represents an important decision for its application (a typical option is to use a proportional method, like $T_k = \alpha \cdot T_{k-1}$). SA usually gives better results in practice, but uses to be very slow. The most striking difficulty in applying SA is to choose and tune its parameters such as initial and final temperature, decrement of the temperature (cooling schedule), equilibrium detection, etc.

For further details on initial solution, fitness evaluation and movement types, refer to [110][*1] However, the acceptability criteria of neighboring solutions is now different, as explained next.

■Acceptability Criteria The acceptability criteria for newly generated solution is based on the definition of a threshold value (accepting threshold) as follows. We consider a succession t_k such that $t_k > t_{k+1}$, $t_k > 0$ and t_k tends to 0 as k tends to infinity. Then, for any two solutions s_i and s_j, if $fitness(s_j) - fitness(s_i) < t_k$, then accept solution s_j.

For the SA, t_k values are taken as accepting threshold but the criterion for acceptance is probabilistic:

- If $fitness(s_j) - fitness(s_i) \leq 0$ then s_j is accepted.
- If $fitness(s_j) - fitness(s_i) > 0$ then s_j is accepted with probability $\exp[(fitness(s_j) - fitness(s_i))/t_k]$ (at iteration k the algorithm generates a random number $R \in (0, 1)$ and s_j is accepted if $R < \exp[(fitness(s_j) - fitness(s_i))/t_k]$).

In this case, each neighbor of a solution has a positive probability of replacing the current solution. The t_k values are chosen in way that solutions with large increase in the cost of the solutions are less likely to be accepted (but there is still a positive probability of accepting them).

Algorithm 12 : Pseudo-code of WMN-SA.

```
t := 0
Initialize T
s0 := Initial_Solution()
v0 := Evaluate(s0)
while (stopping condition not met) do
  while t mod MarkovChainLen = 0 do
    t := t+1
    s1 := Generate(s0,T)   //Move
    v1 := Evaluate(s1)
    if Accept(v0,v1,T) then
      s0 := s1
      v0 := v1
    end if
  end while
  T := Update(T)
end while
return s0
```

6.10 Web Interface

The Web application [111] follows a standard Client-Server architecture and is implemented using LAMP (Linux + Apache + MySQL + PHP) technology (see Fig. 6.3). Remote users (clients) submit their requests by completing first the parameter setting. The parameter values to be provided by the user are classified into three groups, as follows.

[*1] Initial solution, fitness evaluation and movement types are the same for Hill Climbing and Simulated Annealing

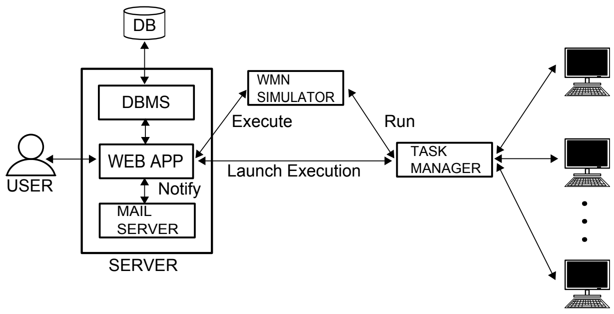
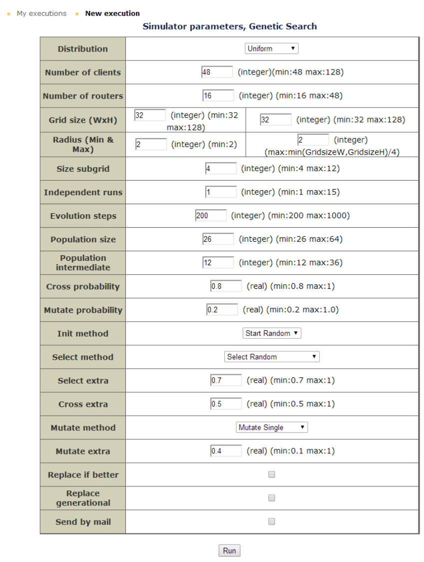
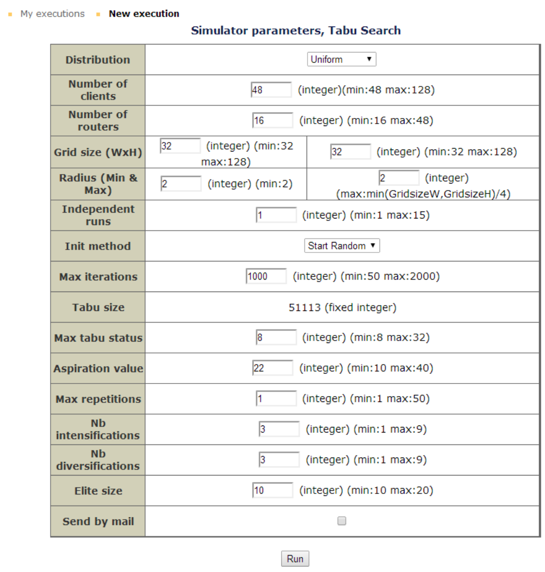


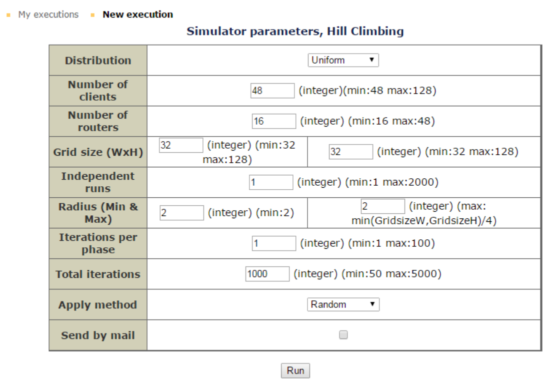
Fig. 6.3 System structure for Web interface



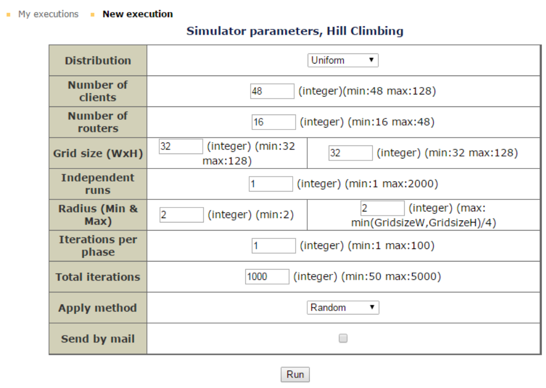
(a) GA



(b) TS



(c) HC



(d) SA

Fig. 6.4 Web interfaces

- *Parameters related to the problem instance*: These include parameter values that determine a problem instance to be solved and consist of number of router nodes, number of mesh client nodes, client mesh distribution, radio coverage interval and size of the deployment area.

- *Parameters of the resolution method*: Each method has its own parameters. In Fig. 6.4 are shown the the GUI of Web Interfaces for the parameter setting of GA, TS, HC and SA.
- *Execution parameters*: These parameters are used for stopping condition of the resolution methods and include number of iterations and number of independent runs. The former is provided as a total number of iterations and depending on the method is also divided per phase (e.g., number of iterations in a exploration). The later is used to run the same configuration for the same problem instance and parameter configuration a certain number of times.

6.11 ns-3

The ns-3 simulator [112] is developed and distributed completely in the C++ programming language, because it better facilitated the inclusion of C-based implementation code. The ns-3 architecture is similar to Linux computers, with internal interface and application interfaces such as network interfaces, device drivers and sockets. The goals of ns-3 are set very high: to create a new network simulator aligned with modern research needs and develop it in an open source community. Users of ns-3 are free to write their simulation scripts as either *C++ main()* programs or *Python* programs. The ns-3's low-level API is oriented towards the power-user but more accessible "helper" APIs are overlaid on top of the low-level API.

In order to achieve scalability of a very large number of simulated network elements, the ns-3 simulation tools also support distributed simulation. The ns-3 support standardized output formats for trace data, such as the pcap format used by network packet analyzing tools such as tcpdump, and a standardized input format such as importing mobility trace files from ns-2.

The ns-3 simulator is equipped with *Pyviz* visualizer, which has been integrated into mainline ns-3, starting with version 3.10. It can be most useful for debugging purposes, i.e. to figure out if mobility models are what you expect, where packets are being dropped. It is mostly written in Python and it works both with Python and pure C++ simulations. The function of ns-3 visualizer is more powerful than network animator (*nam*) of ns-2 simulator.

The ns-3 simulator has models for all network elements that comprise a computer network. For example, network devices represent the physical device that connects a node to the communication channel. This might be a simple Ethernet network interface card or a more complex wireless IEEE 802.11 device.

The ns-3 is intended as an eventual replacement for popular ns-2 simulator. The ns-3's wifi models a wireless network interface controller based on the IEEE 802.11 standard [113]. The ns-3 provides models for these aspects of 802.11:

1. Basic 802.11 DCF with infrastructure and ad hoc modes.
2. 802.11a, 802.11b, 802.11g and 802.11s physical layers.
3. QoS-based EDCA and queueing extensions of 802.11e.
4. Various propagation loss models including Nakagami, Rayleigh, Friis, LogDistance, FixedRss, and so on.

5. Two propagation delay models, a distance-based and random model.
6. Various rate control algorithms including Aarf, Arf, Cara, Onoe, Rraa, ConstantRate, and Minstrel.

■Log-distance Path Loss Model The log-distance path loss model is a radio propagation model that predicts the path loss a signal encounters inside a building or densely populated areas over distance. This propagation model is applicable for indoor propagation modeling. Log-distance propagation loss model [112] is formally expressed as:

$$L = L_0 + 10nlog_{10}(\frac{d}{d_0}) \tag{6.3}$$

where:

- n: the path loss distance exponent,
- d_0: reference distance [m],
- L_0: path loss at reference distance [dB],
- d: distance [m],
- L: path loss [dB].

When the path loss is requested at a distance smaller than the reference distance, the value of Tx power is returned.

Chapter 7

Simulation Results

7.1 Simulation Scenario 1

In Simulation Scenario 1, we analyze the mesh router placement in WMNs using Friedman Test considering different meta-heuristic methods.

7.1.1 Simulation Description and Design

In this work, we consider different radius of communication distances and evaluate the performance of WMN-GA, WMN-TS, WMN-HC and WMN-SA for Uniform (U), Normal (N), Exponential (E) and Weibull (W) distributions. The number of mesh routers for all scenarios is considered 16 and the number of mesh clients 48. The input parameters for WMN-GA, WMN-TS, WMN-HC and WMN-SA are shown in Table 7.1, Table 7.2, Table 7.3 and Table 7.4, respectively.

Table 7.1 Input parameters of WMN-GA.

Parameters	Values
Number of clients	48
Number of routers	16
Grid width	32 [units]
Grid height	32 [units]
Communication Distance (min:max)	2×2:n×n (n=2, 4, 6, 8) [units]
Independent runs	10
Initial Router Placement Method	HotSpot
Number of Generations	200
Population size	32
Selection Method	Linear Ranking
Crossover rate	80 %
Mutate Method	Single
Mutate rate	20 %
Distribution of Clients	N, U, E, W

Let us define the set of mesh routers as R and the set of mesh clients as C. In our system, fitness function is defined as

$$Fitness(R, C) = 0.7 \times SGC(R) + 0.3 \times NCMC(C) \tag{7.1}$$

The Friedman test [114] is a nonparametric statistical test of multiple group measures. It can be used to approve the null hypothesis that the multiple group measures have the same variance to

Table 7.2 Input parameters of WMN-TS.

Parameters	Values
Number of clients	48
Number of routers	16
Grid width	32 [units]
Grid height	32 [units]
Communication Distance (min:max)	2×2:n×n (n=2, 4, 6, 8) [units]
Independent runs	10
Initial Router Placement Method	HotSpot
Max Iterations	2000
Max Tabu Status	9
Aspiration Value	15
Max Repetitions	15
Number of Intensifications	4
Number of Diversifications	4
Elite Size	10
Distribution of Clients	N, U, E, W

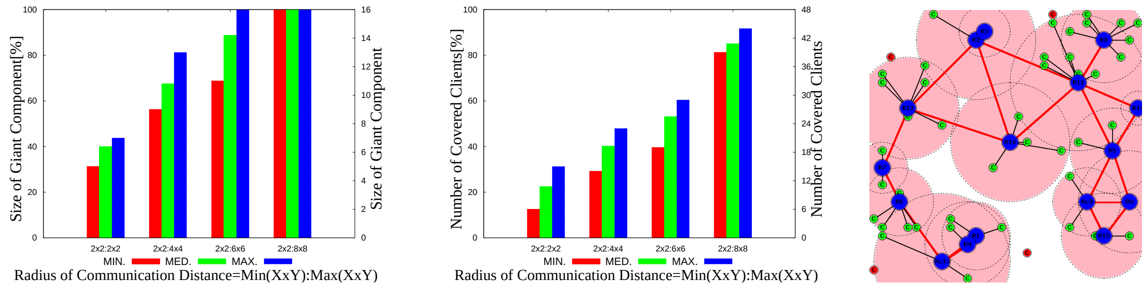
Table 7.3 Input parameters of WMN-HC.

Parameters	Values
Number of clients	48
Number of routers	16
Grid width	32 [units]
Grid height	32 [units]
Communication Distance (min:max)	2×2:n×n (n=2, 4, 6, 8) [units]
Independent runs	10
Initial Router Placement Method	HotSpot
Iteration per Phases	9
Total Iterations	2000
Replacement Method	Combination
Distribution of Clients	N, U, E, W

Table 7.4 Input parameters of WMN-SA.

Parameters	Values
Number of clients	48
Number of routers	16
Grid width	32 [units]
Grid height	32 [units]
Communication Distance (min:max)	2×2:n×n (n=2, 4, 6, 8) [units]
Independent runs	10
Initial Router Placement Method	HotSpot
Iteration per Phases	9
Total Iterations	2000
Temperature	1
Replacement Method	Combination
Distribution of Clients	N, U, E, W

a certain required level of significance. On the other hand, failing to approve the null hypothesis shows that they have different variance values. We analyze the difference in performance between GA, TS, HC and SA using Friedman test in MATLAB. We considered as null hypothesis H_0 that there is difference in the performance between GA, TS, HC and SA. As alternative hypothesis we considered H_1 that there is no difference in the performance of GA, TS, HC and SA. As value of the hypothesis testing we took the maximum value of number of covered mesh clients and size of giant component. The significance level in this testing hypothesis is $\alpha = 0.05$. We reject H_0



(a) SGC vs. radius of communication distance.

(b) NCMC vs. radius of communication distance.

(c) Visualization of nodes after placement for 2×2:8×8

Fig. 7.1 Simulation results of the WMN-GA for Uniform distribution.

for $p > \alpha$ (p-value is the probability of obtaining a test statistic at least as extreme as the one that was actually observed, assuming that the null hypothesis is true). Further, since there is a correspondence between GA, TS, HC and SA, we used Friedman test. The results of p-values for Friedman test for SGC are shown in Table 7.5 and H_1 is rejected because $p < 0.05$. In this case, we adopted H_1. For NCMC, the p-values are shown in Table 7.6, and we adopt H_1 since $p < 0.05$.

Table 7.5 The p-value of SGC of Friedman test.

	GA	TS	HC	SA
GA	N/A	0.0027	0.0027	0.0027
TS	0.0027	N/A	0.3173	0.3173
HC	0.0027	0.3173	N/A	0.3173
SA	0.0027	0.3173	0.3173	N/A

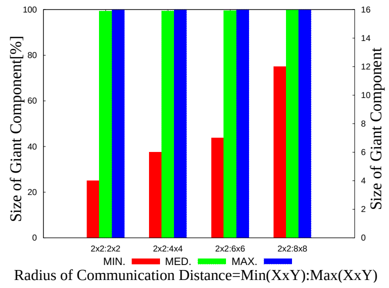
Table 7.6 The p-value of NCMC of Friedman test.

	GA	TS	HC	SA
GA	N/A	0.6171	0.0334	0.0334
TS	0.6171	N/A	0.0196	0.0196
HC	0.0334	0.3173	N/A	0.1797
SA	0.0334	0.0196	0.1797	N/A

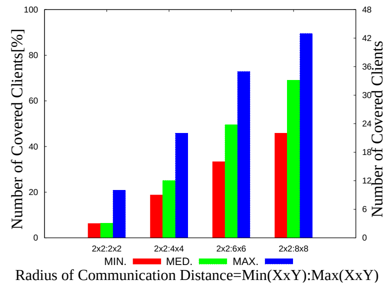
7.1.2 Discussion of Simulation Results

In Fig. 7.1, Fig. 7.2, Fig. 7.3 and Fig. 7.4 are shown simulation results for Uniform distribution using WMN-GA, WMN-TS, WMN-HC and WMN-SA, respectively. We used bar graph representation that shows the minimum, average and maximum value. We also show the visualization of nodes after the placement. If we compare the results, the WMN-HC and WMN-SA perform better than WMN-GA and WMN-TS. However, for radius of communication distance 2×2:2×2, the SGC of WMN-TS is better than other systems.

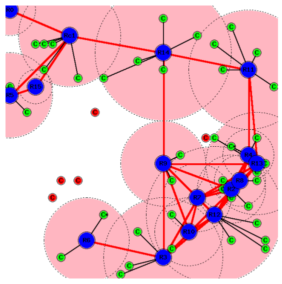
In Fig. 7.5, Fig. 7.6, Fig. 7.7 and Fig. 7.8 are shown simulation results for Normal distribution using WMN-GA, WMN-TS, WMN-HC and WMN-SA, respectively. If we compare the results, the WMN-HC and WMN-SA perform better than WMN-GA and WMN-TS. But, for radius of



(a) SGC vs. radius of communication distance.

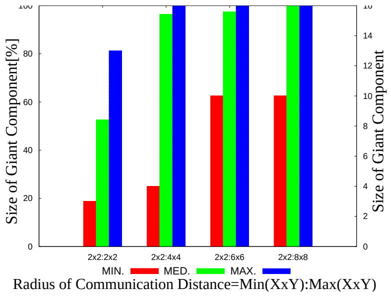


(b) NCMC vs. radius of communication distance.

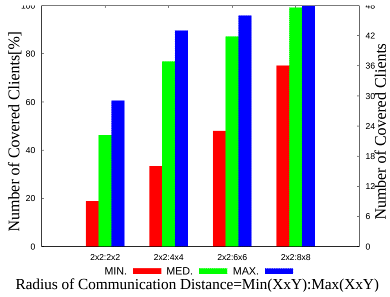


(c) Visualization of nodes after placement for 2×2:8×8

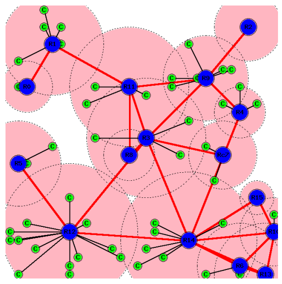
Fig. 7.2 Simulation results of the WMN-TS for Uniform distribution.



(a) SGC vs. radius of communication distance.

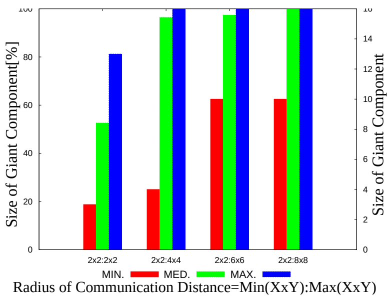


(b) NCMC vs. radius of communication distance.

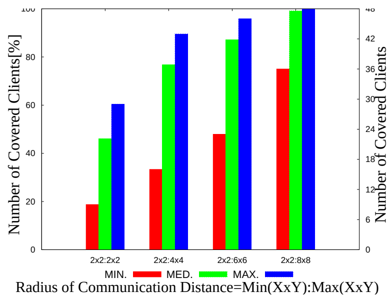


(c) Visualization of nodes after placement for 2×2:8×8

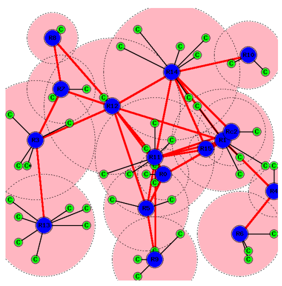
Fig. 7.3 Simulation results of the WMN-HC for Uniform distribution.



(a) SGC vs. radius of communication distance.



(b) NCMC vs. radius of communication distance.

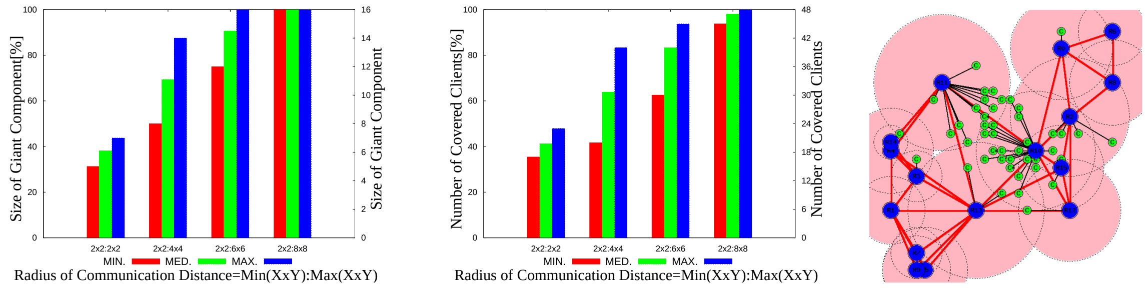


(c) Visualization of nodes after placement for 2×2:8×8

Fig. 7.4 Simulation results of the WMN-SA for Uniform distribution.

communication distance 2×2:8×8, the SGC and NCMC (see Fig. 7.5(a) and Fig. 7.5(b)) of WMN-GA are better than other systems.

In Fig. 7.9, Fig. 7.10, Fig. 7.11 and Fig. 7.12 are shown simulation results for Exponential distribution using WMN-GA, WMN-TS, WMN-HC and WMN-SA, respectively. If we compare

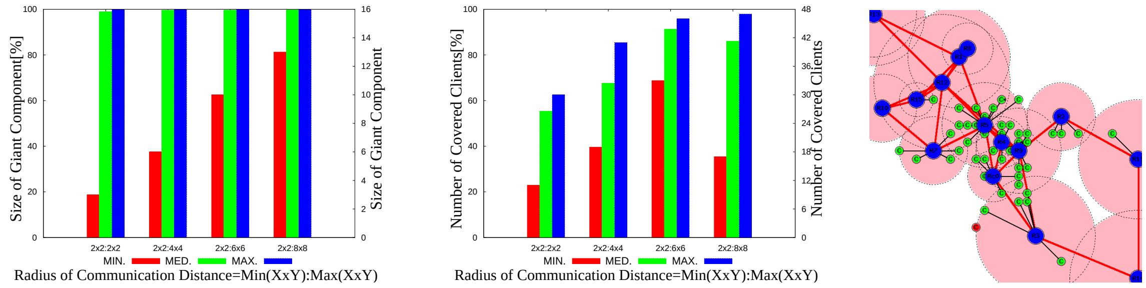


(a) SGC vs. radius of communication distance.

(b) NCMC vs. radius of communication distance.

(c) Visualization of nodes after placement for 2×2:8×8

Fig. 7.5 Simulation results of the WMN-GA for Normal distribution.

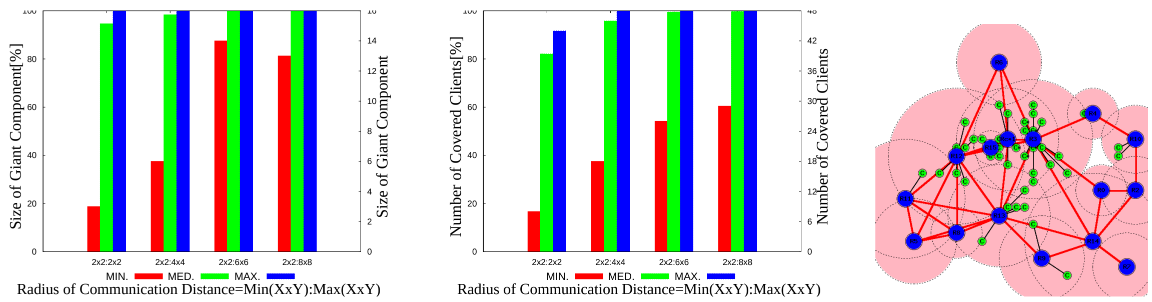


(a) SGC vs. radius of communication distance.

(b) NCMC vs. radius of communication distance.

(c) Visualization of nodes after placement for 2×2:8×8

Fig. 7.6 Simulation results of the WMN-TS for Normal distribution.



(a) SGC vs. radius of communication distance.

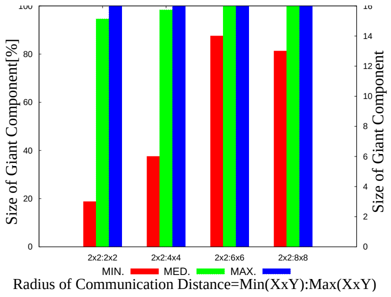
(b) NCMC vs. radius of communication distance.

(c) Visualization of nodes after placement for 2×2:8×8

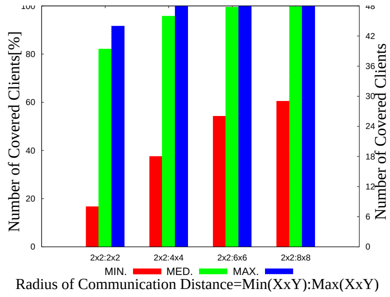
Fig. 7.7 Simulation results of the WMN-HC for Normal distribution.

the results, the WMN-TS, WMN-HC and WMN-SA perform better than WMN-GA for all radius of communication distances.

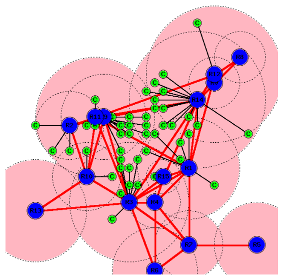
In Fig. 7.13, Fig. 7.14, Fig. 7.15 and Fig. 7.16 are shown simulation results for Weibull distribution using WMN-GA, WMN-TS, WMN-HC and WMN-SA, respectively. If we compare



(a) SGC vs. radius of communication distance.

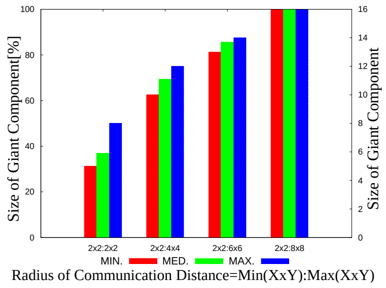


(b) NCMC vs. radius of communication distance.

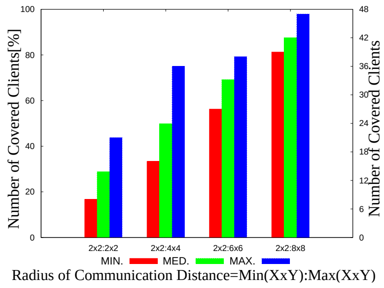


(c) Visualization of nodes after placement for 2×2:8×8

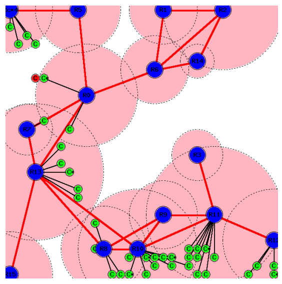
Fig. 7.8 Simulation results of the WMN-SA for Normal distribution.



(a) SGC vs. radius of communication distance.

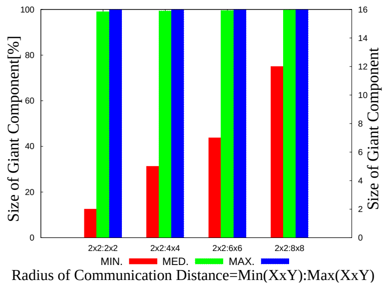


(b) NCMC vs. radius of communication distance.

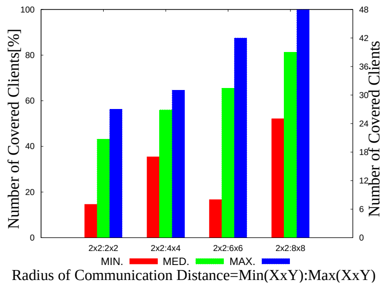


(c) Visualization of nodes after placement for 2×2:8×8

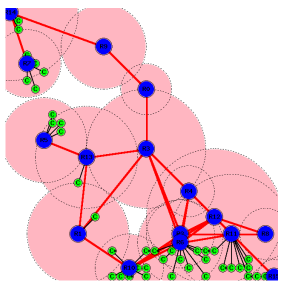
Fig. 7.9 Simulation results of the WMN-GA for Exponential distribution.



(a) SGC vs. Radius of communication distance.



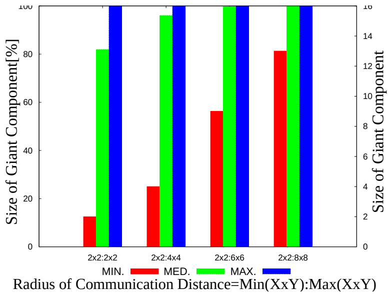
(b) NCMC vs. Radius of communication distance.



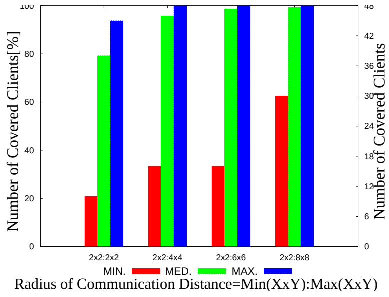
(c) Visualization of nodes after placement for 2×2:8×8

Fig. 7.10 Simulation results of the WMN-TS for Exponential distribution.

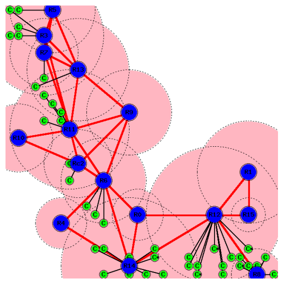
the results, the WMN-TS has a good performance for radius of communication distance less than 2×2:6×6, but for 2×2:8×8 the WMN-GA, WMN-HC and WMN-SA perform better.



(a) SGC vs. radius of communication distance.

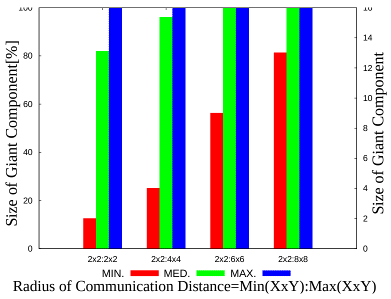


(b) NCMC vs. radius of communication distance.

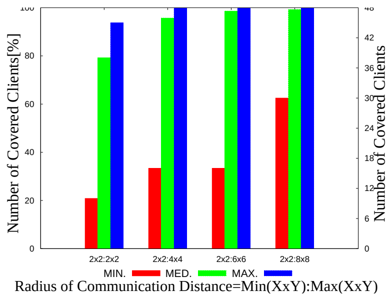


(c) Visualization of nodes after placement for 2×2:8×8

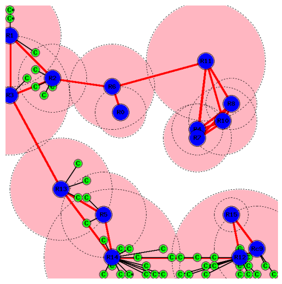
Fig. 7.11 Simulation results of the WMN-HC for Exponential distribution.



(a) SGC vs. radius of communication distance.

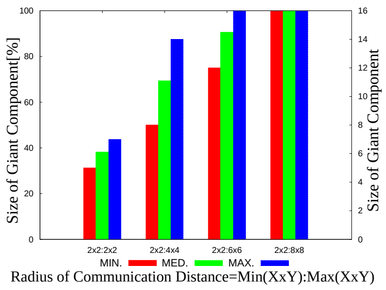


(b) NCMC vs. radius of communication distance.

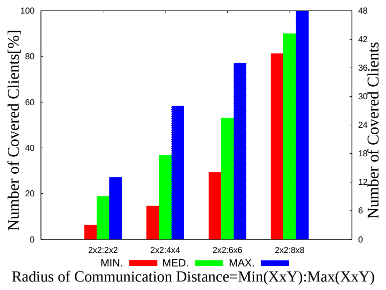


(c) Visualization of nodes after placement for 2×2:8×8

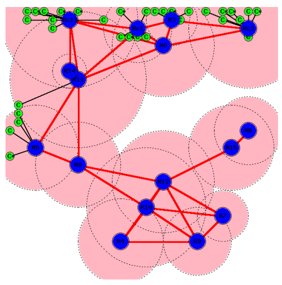
Fig. 7.12 Simulation results of the WMN-SA for Exponential distribution.



(a) SGC vs. radius of communication distance.



(b) NCMC vs. radius of communication distance.

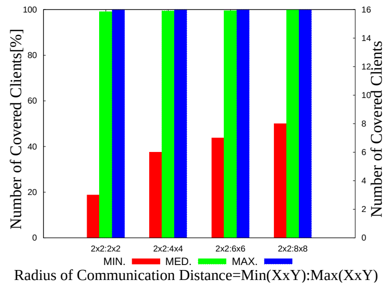


(c) Visualization of nodes after placement for 2×2:8×8

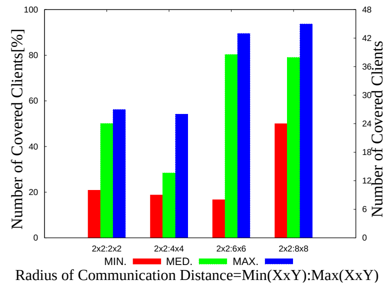
Fig. 7.13 Simulation results of the WMN-GA for Weibull distribution.

7.2 Simulation Scenario 2

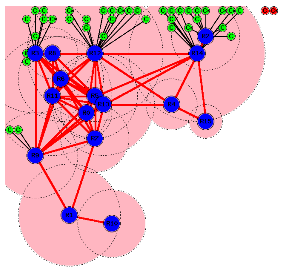
In Simulation Scenario 2, we analyze the WMN-GA system data considering the number of mesh routers.



(a) SGC vs. radius of communication distance.

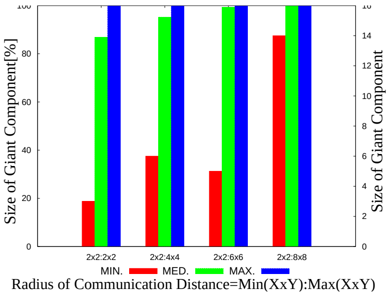


(b) NCMC vs. radius of communication distance.

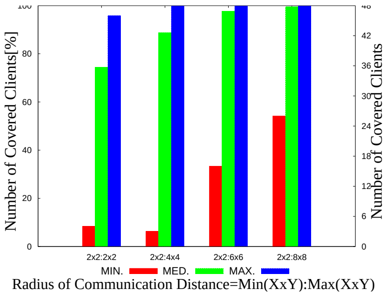


(c) Visualization of nodes after placement for 2×2:8×8

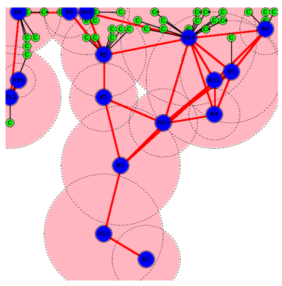
Fig. 7.14 Simulation results of the WMN-TS for Weibull distribution.



(a) SGC vs. radius of communication distance.

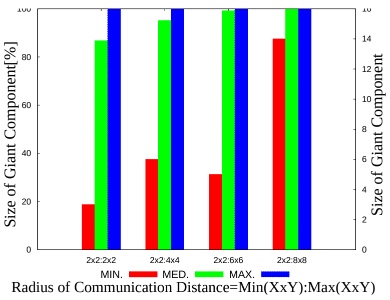


(b) NCMC vs. radius of communication distance.

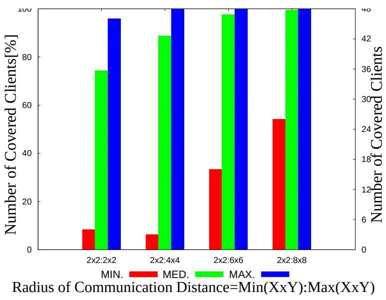


(c) Visualization of nodes after placement for 2×2:8×8

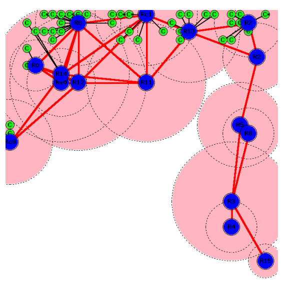
Fig. 7.15 Simulation results of the WMN-HC for Weibull distribution.



(a) SGC vs. radius of communication distance.



(b) NCMC vs. radius of communication distance.

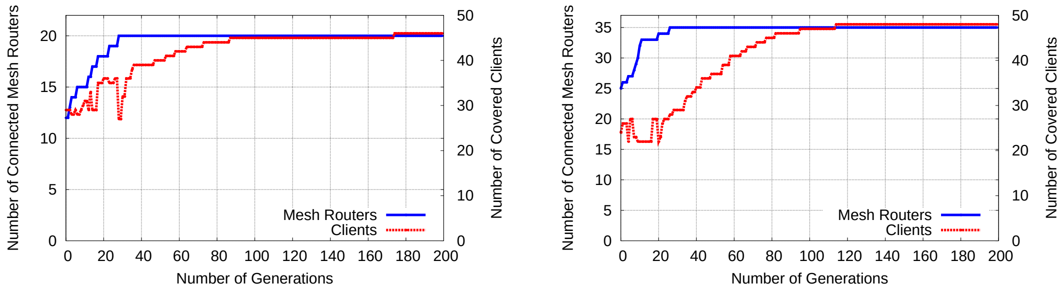


(c) Visualization of nodes after placement for 2×2:8×8

Fig. 7.16 Simulation results of the WMN-SA for Weibull distribution.

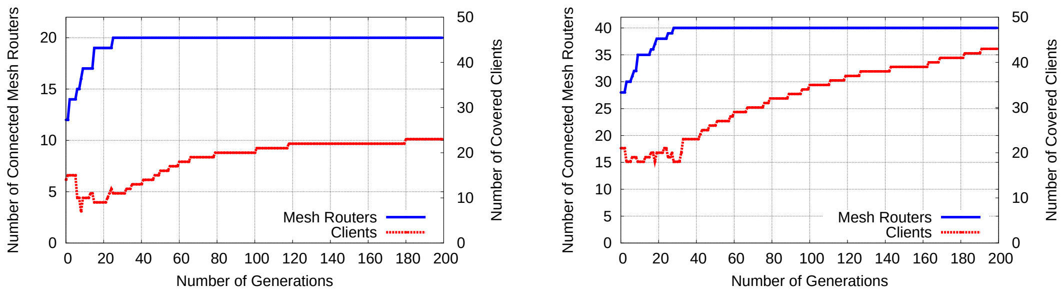
7.2.1 Positioning of Mesh Routers by WMN-GA

We use WMN-GA system for node placement problem in WMNs. A bi-objective optimization is used to solve this problem by first maximizing the SGC routers in the network and then the client coverage. The input parameters of WMN-GA system are shown in Table 7.7.



(a) Number of mesh routers: 20 (b) Number of mesh routers: 35

Fig. 7.17 SGC and NCMC vs. number of generations for Normal distribution.



(a) Number of mesh routers: 20 (b) Number of mesh routers: 40

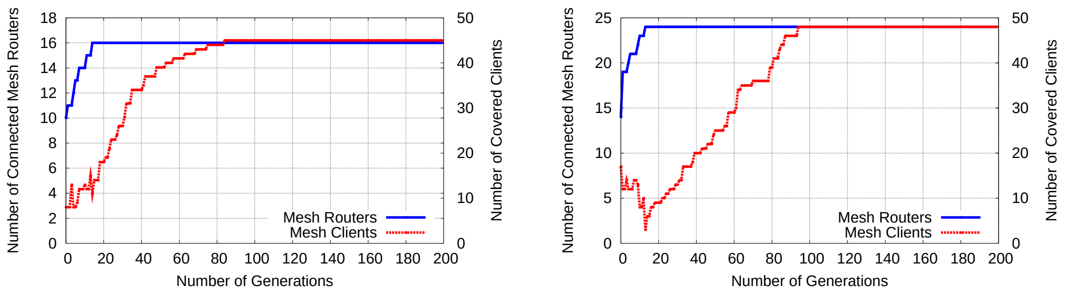
Fig. 7.18 SGC and NCMC vs. number of generations for Uniform distribution.

In Fig. 7.17, Fig. 7.18, Fig. 7.19 and Fig. 7.20 are shown the simulation results of SGC vs. number of generations for four distributions, respectively. After few generations, all routers are connected with each other for both distributions. Then, we optimize the position of routers in order to cover as many mesh clients as possible.

Table 7.7 Input parameters of WMN-GA.

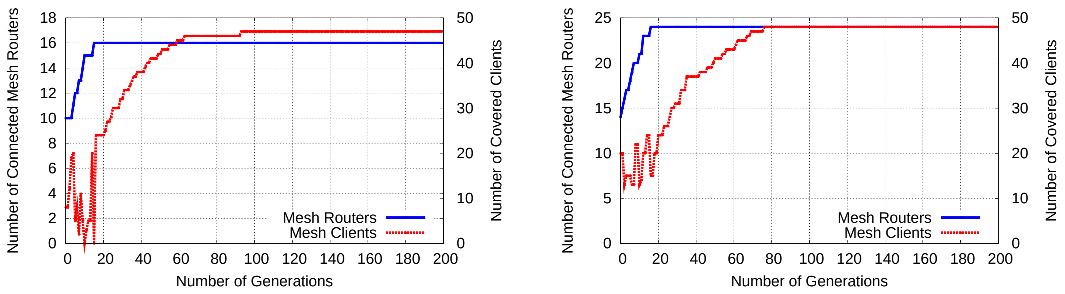
Parameters	Values
Number of clients	48
Number of routers	16, 20, 24, 28, 30, 35 and 40
Grid width	32 units
Grid height	32 units
Independent runs	10
Number of Generations	200
Population size	4096
Selection Method	Linear Ranking
Crossover rate	80 %
Mutate Method	Single
Mutate rate	20 %
Distribution of Mesh Clients	Normal, Uniform, Exponential, Weibull

In Table 7.8 and Table 7.9 are shown the results of SGC and NCMC for 200 generations for each distribution. In this case, by increasing the number of mesh routers, the number of covered clients is increased. We used WMN-GA with grid size (32 [units]×32 [units]), different number of mesh



(a) Number of mesh routers: 16 (b) Number of mesh routers: 24

Fig. 7.19 SGC and NCMC vs. number of generations for Exponential distribution.



(a) Number of mesh routers: 16 (b) Number of mesh routers: 24

Fig. 7.20 SGC and NCMC vs. number of generations for Weibull distribution.

routers and 48 mesh clients to allocate the position of mesh routers. The network topologies for four distributions are shown in Fig. 7.21, Fig. 7.22, Fig. 7.23 and Fig. 7.24, respectively.

Table 7.8 Evaluation of WMN-GA for Normal and Uniform distributions.

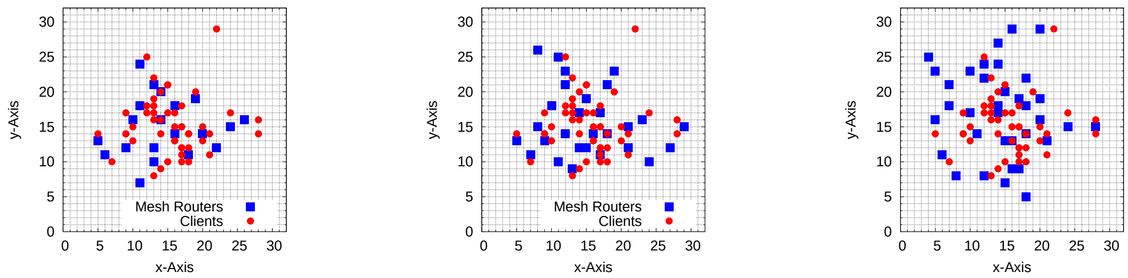
Number of	Normal Distribution		Uniform Distribution	
mesh routers	SGC	SGC	SGC	SGC
20	20	46	20	23
28	28	47	28	36
35	35	48	35	37

Table 7.9 Evaluation of WMN-GA for Exponential and Weibull distributions.

Number of	Exponential Distribution		Weibull Distribution	
mesh routers	NCMC	NCMC	NCMC	NCMC
20	20	48	20	48
28	28	48	28	48
35	35	48	35	48

7.2.2 Simulation Description

The topologies of our WMN are generated using WMN-GA system (see Figs. 7.21, 7.22, 7.23 and 7.24). We took in consideration the connectivity between mesh routers and conduct simulations

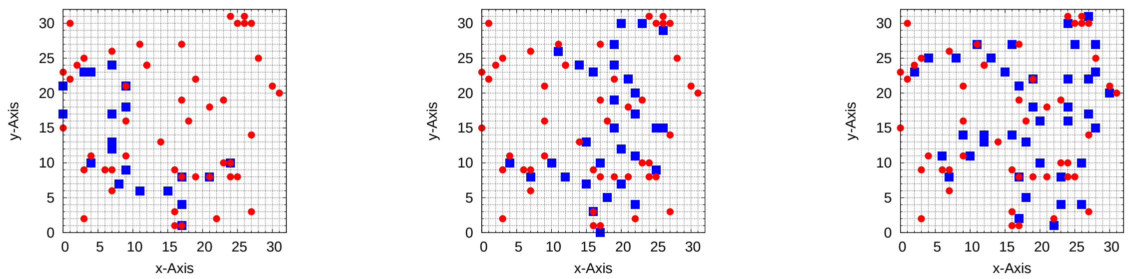


(a) Number of mesh routers: 20 (20, 46)

(b) Number of mesh routers: 28 (28, 47)

(c) Number of mesh routers: 35 (35, 48)

Fig. 7.21 (m, n): m is SGC, n is NCMC for Normal distribution.

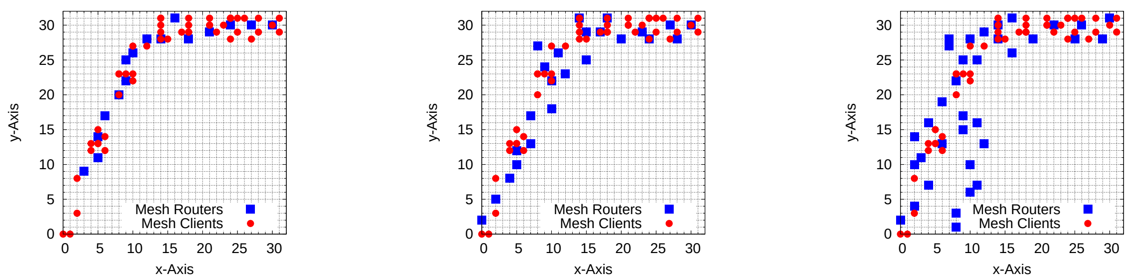


(a) Number of mesh routers: 20 (20, 23)

(b) Number of mesh routers: 30 (30, 34)

(c) Number of mesh routers: 40 (40, 43)

Fig. 7.22 (m, n): m is SGC, n is NCMC for Uniform distribution.



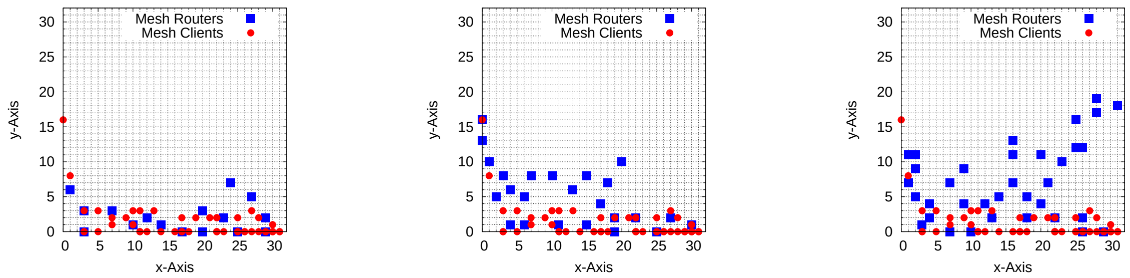
(a) Number of mesh routers: 16 (16, 45)

(b) Number of mesh routers: 24 (24, 48)

(c) Number of mesh routers: 35 (35, 48)

Fig. 7.23 (m, n): m is SGC, n is NCMC for Exponential distribution.

using ns-3. The simulations are done for different number of mesh routers and 48 mesh clients for 4 different distributions (Normal, Uniform, Exponential and Weibull). The area size is considered 640 [m]×640 [m] (or 32 [units]×32 [units]). We used HWMP routing protocol and sent multiple CBR flows over UDP. The pairs source-destination are the same for all simulation scenarios. We made simulations for 10 and 20 connections considering Log-distance path loss model and constant speed delay model. Other simulation parameters are shown in Table 7.10.



(a) Number of mesh routers: 16 (16, 47)

(b) Number of mesh routers: 24 (24, 48)

(c) Number of mesh routers: 35 (35, 48)

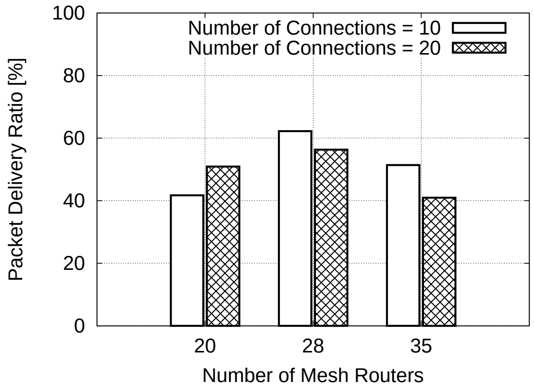
Fig. 7.24 (m, n): m is SGC, n is NCMC for Weibull distribution.

Table 7.10 Simulation parameters.

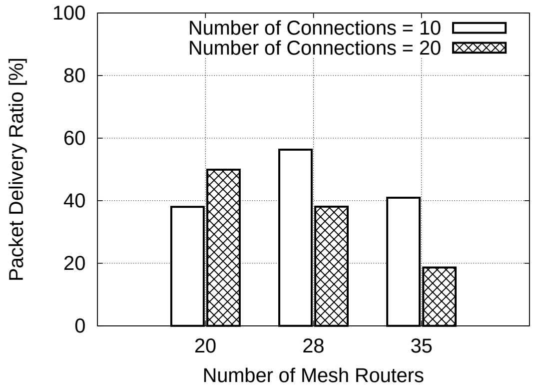
Parameters	Values
Area Size	640 [m]×640 [m]
MAC	IEEE 802.11s
Propagation Loss Model	Log-distance Path Loss Model
Propagation Delay Model	Constant Speed Model
Number of Mesh routers	20, 28, 35
Number of Mesh clients	48
Number of Connections	10, 20
Transport Protocol	UDP
Application Type	CBR
Packet Size	1024 bytes
Data Rate	1.4 [Mbps]
Source Node ID	Random
Destination Node ID	Random
Simulation Time	650 [sec]

7.2.3 Discussion of Simulation Results

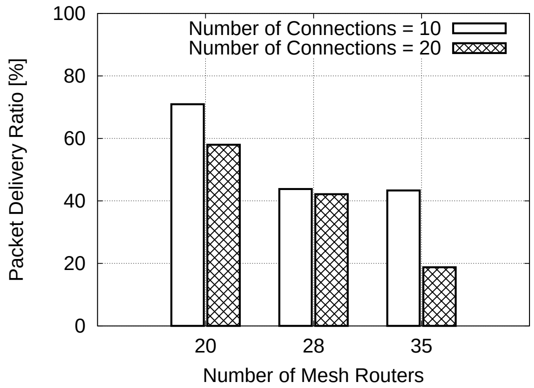
We used PDR, throughput and delay metrics for performance evaluation. In Fig. 7.25(a), we show the simulation results of PDR vs. number of mesh routers for Normal distribution for 10 and 20 connections. When there are 28 mesh routers in the network, the performance of PDR is higher. When the number of mesh routers is increased the number of hops increases and PDR decreases. In Fig. 7.25(b), we show the simulation results of PDR for Uniform distribution. Also in this case when the number of mesh routers is increased the number of hops increases and PDR decreases. When there are 20 mesh routers in the network with 20 connections, the PDR is higher compared with 10 connections. This happens because there are more communicating pairs near each other. In Fig. 7.25(c), we show the simulation results of PDR for Exponential distribution. When there are 20 mesh routers in the network, the performance of PDR is higher. When the number of mesh routers is increased, the number of hops increases and PDR decreases. In Fig. 7.25(d), we show the simulation results of PDR for Weibull distribution. When there are 35 mesh routers in the network with 10 connections, the PDR is higher compared with other cases.



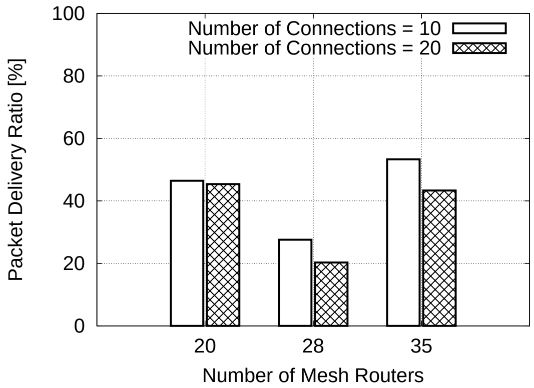
(a) Normal distribution



(b) Uniform distribution



(c) Exponential distribution



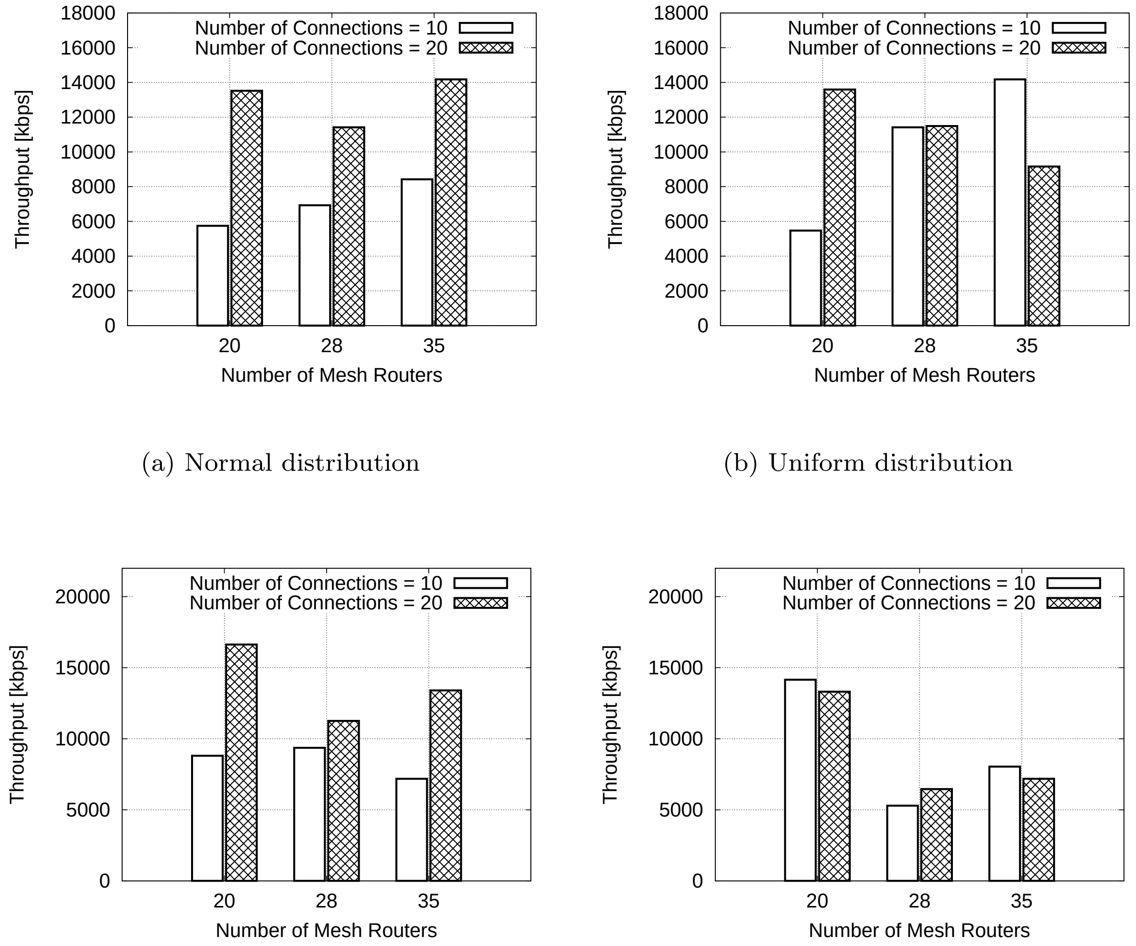
(d) Weibull distribution

Fig. 7.25 Results of mean PDR.

In Fig. 7.26(a), we show the simulation results of throughput vs. number of mesh routers for Normal distribution for 10 and 20 connections. Based on the number of connections, the total data rate that is transmitted in the network changes. The theoretical throughput is calculated by the following equation:

$$\text{Theoretical throughput} = \text{Transmission rate} \times \text{numCon} \tag{7.2}$$

When the number of connections is 10, there is an improvement of throughput when the number of mesh routers increases. In the case of 20 connections, the network load is high and the throughput is different for different number of mesh routers. In the case of Normal distribution, all mesh routers are concentrated in the center of grid and the communication becomes easy. On the other hand, for Uniform distribution the mesh routers are more scattered, the creating of links is more difficult and the communication can be done only with multiple hops. In Fig. 7.26(b) for big number of mesh routers the total data rate for 20 connections is very high (24 [Mbps]), many packets are dropped because of congestion and the throughput is decreased. In Fig. 7.26(c), we show the simulation results of throughput for Exponential distribution. The throughput is decreased with the increase of number of mesh routers. This happens because the number of hops is increased. On the other hand, for Weibull distribution (see Fig. 7.26(d)) when there are 20 connections, with



(a) Normal distribution (b) Uniform distribution

(c) Exponential distribution (d) Weibull distribution

Fig. 7.26 Results of mean throughput.

the increase of the number of mesh routers the throughput is decreased much more than the case of Exponential distribution. This is because many packets are dropped.

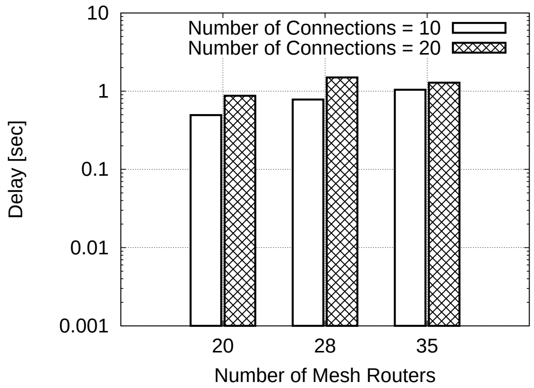
When the number of nodes in the network is increased, there are many intermediate nodes and the distance between them is big, which causes the increase of the delay (see Fig. 7.27(a) and Fig. 7.27(b)). However, for Exponential and Weibull distributions, because these distributions cover a certain area of the grid (hotspot case), the delay is almost the same (See Fig.7.27(c) and Fig. 7.27(d)).

7.3 Simulation Scenario 3

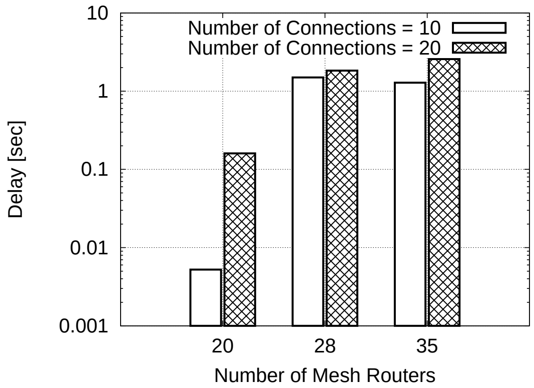
In Simulation Scenario 3, we analyze the WMN-GA system data considering different transmission rates.

7.3.1 Simulation Settings

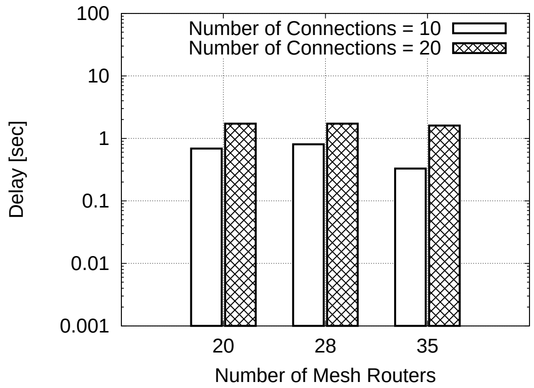
The input parameters of WMN-GA system are shown in Table 7.11. We used a 640 [m]×640 [m] area, with 16 routers and 48 clients for calculating the positions of routers.



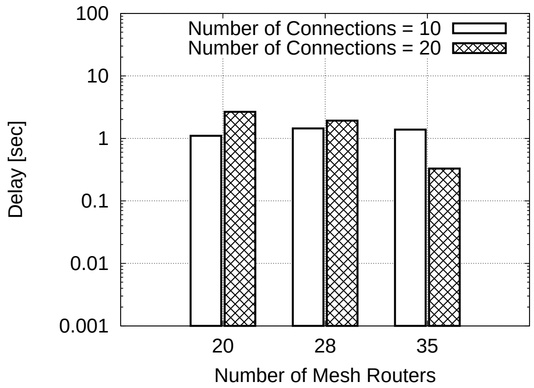
(a) Normal distribution



(b) Uniform distribution



(c) Exponential distribution



(d) Weibull distribution

Fig. 7.27 Results of mean delay.

In Fig. 7.28 and Fig. 7.29 are shown the simulation results of SGC vs. number of generations for Exponential and Weibull distributions, respectively. After few generations, all 16 routers are connected with each other.

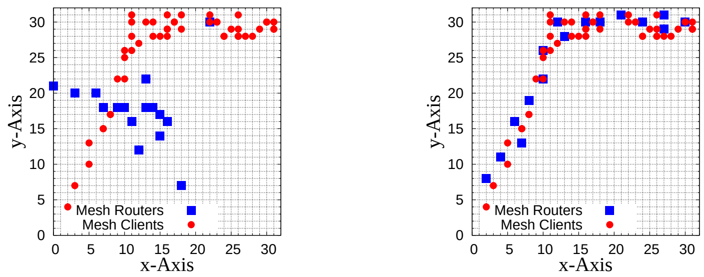
Then we optimize the position of routers in order to cover as many mesh clients as possible. We consider Exponential and Weibull distributions of mesh clients, which are similar with mesh clients concentrated in hot-spot environment.

By increasing the number of generations, the number of covered clients is increased. After 200 generations, more than 47 mesh clients are covered.

The simulations in ns-3 are done for 5 and 200 generations (see Fig. 7.28 and 7.29). The area size is considered 640 [m]$\times$640 [m] (or 32 [units] $\times$32 [units]) and the number of mesh routers is 16. We used HWMP routing protocol and sent multiple CBR flows over UDP. The pairs source-destination are the same for all the simulation scenarios. We made simulations for different number of connections (10, 20 and 30 connections). Log-distance path loss model and constant speed delay model are used for the simulation and other parameters are shown in Table 7.12.

Table 7.11 Input parameters of WMN-GA.

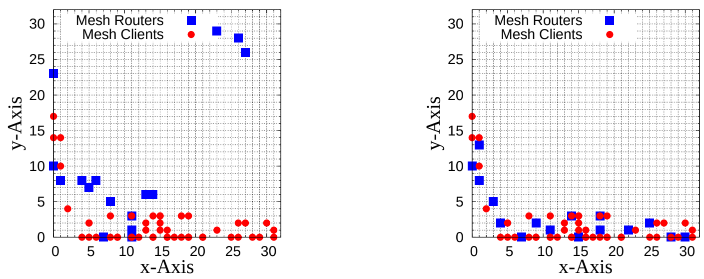
Parameters	Values
Number of clients	48
Number of routers	16
Grid width	32 units
Grid height	32 units
Independent runs	10
Number of Generations	200
Population size	4096
Selection Method	Linear Ranking
Crossover rate	80 %
Mutate Method	Single
Mutate rate	20 %
Distribution of Clients	Exponential and Weibull



(a) Number of generations: 5 (12, 17)

(b) Number of generations: 200 (16, 47)

Fig. 7.28 (m, n): m is SGC, n is NCMC (Exponential distribution).



(a) Number of generations: 5 (12, 12)

(b) Number of generations: 200 (16, 47)

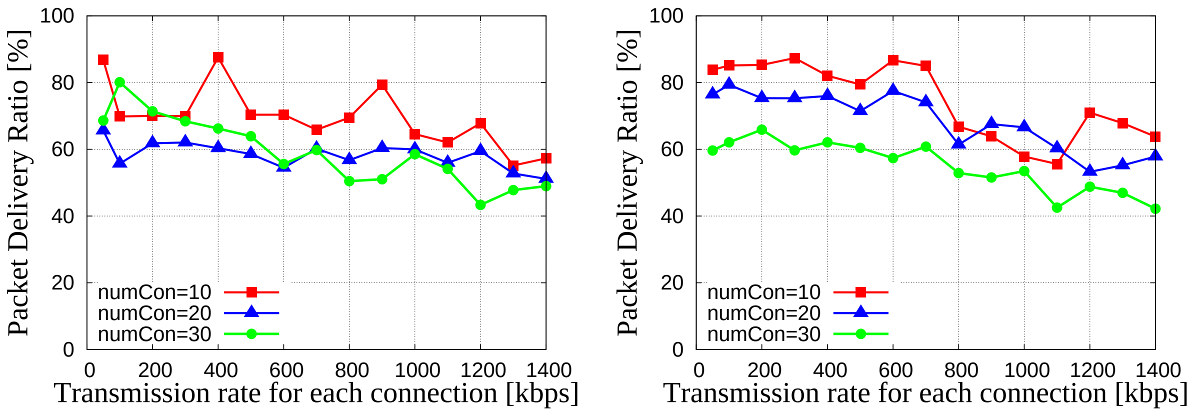
Fig. 7.29 (m, n): m is SGC, n is NCMC (Weibull distribution).

7.3.2 Discussion of Simulation Results

We used PDR, throughput and delay metrics for performance evaluation. In Fig. 7.30, we show the simulation results of PDR vs. transmission rate for 200 generations and number of connections 10, 20 and 30. From the results, we can see that when the number of connections is 10 the PDR is

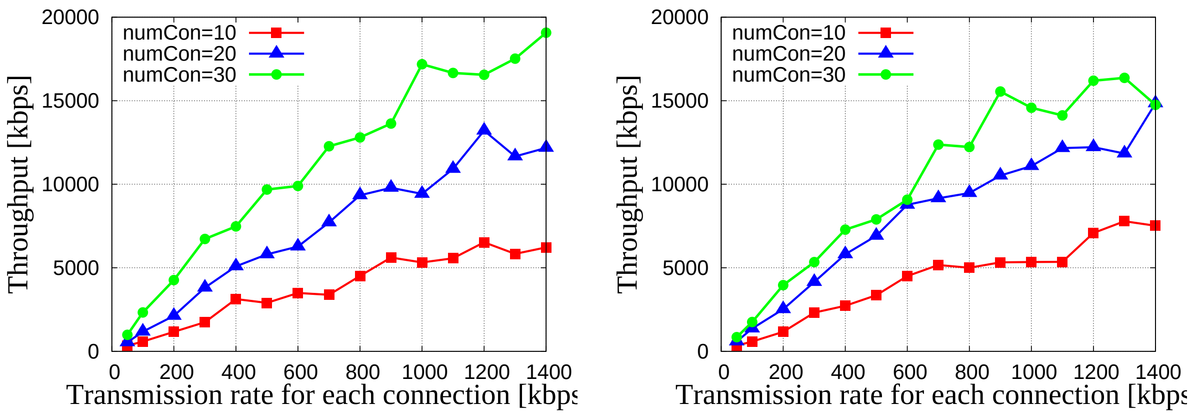
Table 7.12 Simulation parameters.

Parameters	Values
Area Size	640 $[m] \times 640$ $[m]$
MAC	IEEE 802.11s
Propagation Loss Model	Log-distance Path Loss Model
Propagation Delay Model	Constant Speed Model
Number of Mesh routers	16
Number of Mesh clients	48
Number of Connections	10, 20, 30
Transport Protocol	UDP
Application Type	CBR
Packet Size	1024 $[Bytes]$
Source Node ID	Random
Destination Node ID	Random
Simulation Time	650 $[sec]$



(a) Exponential distribution (b) Weibull distribution

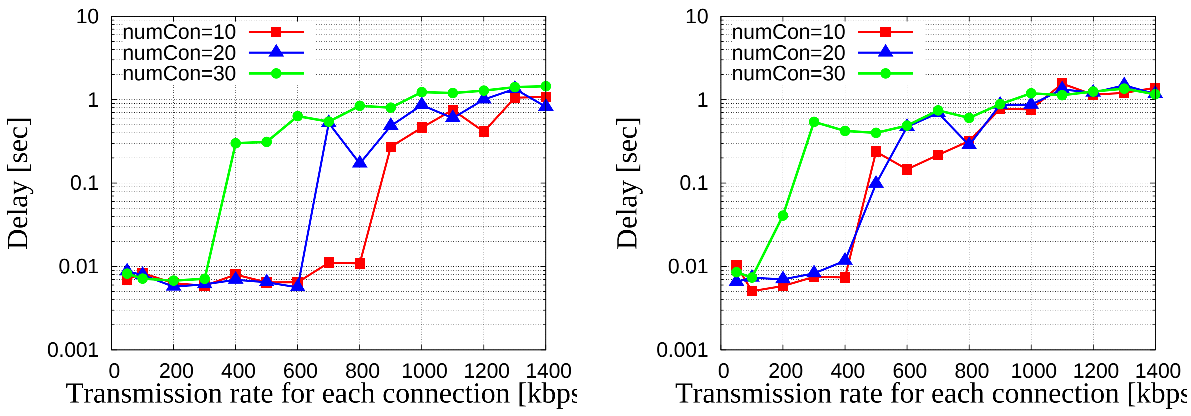
Fig. 7.30 Results of average PDR for no. of generations 200.



(a) Exponential distribution (b) Weibull distribution

Fig. 7.31 Results of average throughput for no. of generations 200.

higher than other cases. With the increasing of the transmission rate, the PDR is decreased. For number of connections 10, 20 and 30, the PDR is less than 60 [%] when the transmission rate is more than 1200 [kbps] for Exponential distribution. The PDR for Weibull distribution is higher than for Exponential distribution. When transmission rate is higher, the PDR decreases because



(a) Exponential distribution

(b) Weibull distribution

Fig. 7.32 Results of average delay for no. of generations 200.

of packet collision and congestion. In this case, the queues of each mesh router get full and many packets are dropped, so the efficiency of the network is decreased.

In Fig. 7.31, we show the simulation results of throughput vs. transmission rate for 200 generations and number of connections 10, 20 and 30. For different number of connections, the throughput is increased linearly with the increasing of the transmission rate. The throughput of Exponential distribution is higher than Weibull distribution.

The simulation results of delay vs. transmission rate for 200 generations are shown in Fig. 7.32. With increasing of the number of connections and transmission rate, the delay is increased. For 10 connections, the delay is very small until the transmission rate is 800 [kbps] for Exponential distribution. The delay of Exponential distribution is smaller than Weibull distribution.

7.4 Simulation Scenario 4

In Simulation Scenario 4, we analyze the WMN-GA system for different WMN architectures considering DCF and EDCA.

7.4.1 Simulation Settings

The input parameters of WMN-GA system are shown in Table 7.13. In Fig. 7.33, we show the location of mesh routers and clients for first generations and the optimized topologies generated by WMN-GA system for normal distribution.

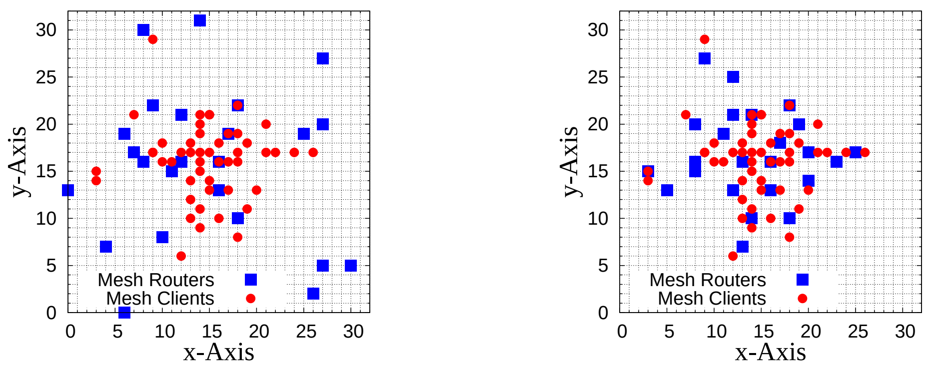
In Fig. 7.34 are shown the simulation results of Size of Giant Component (SGC) and Number of Covered Mesh Clients (NCMC) vs. number of generations. After few generations, all routers are connected with each other.

Then, we optimize the position of routers in order to cover as many mesh clients as possible. We consider normal distribution of mesh clients. The simulation results of SGC and NCMC are shown in Table 7.14.

Table 7.13 Input parameters of WMN-GA system.

Parameters	Values
Number of clients	48
Number of routers	16, 24, 32
Grid width	32 [units]
Grid height	32 [units]
Independent runs	10
Number of generations	200
Population size	64
Selection method	Linear Ranking
Crossover rate	80 [%]
Mutate method	Single
Mutate rate	20 [%]
Distribution of clients	Normal

We conduct simulations using ns-3 simulator. The simulations in ns-3 are done for number of generations 1 and 200. The area size is considered 640 [m]×640 [m] (or 32 [units]×32 [units]) and the number of mesh routers is from 16 to 32. We used DCF, EDCA and OLSR routing protocol and sent multiple CBR flows over UDP. The pairs source-destination are the same for all simulation scenarios. Log-distance path loss model and constant speed delay model are used for the simulation and other parameters are shown in Table 7.15.



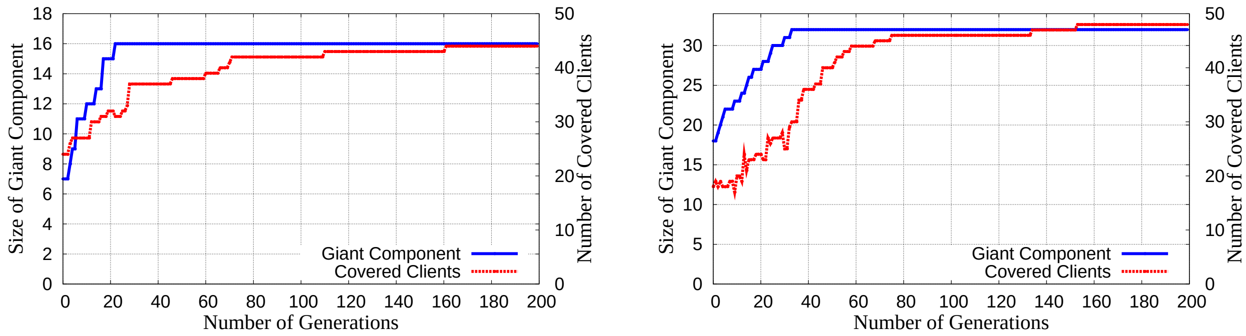
(a) Number of generations: 1 (8, 12)

(b) Number of generations: 200 (24, 47)

Fig. 7.33 (m, n): m is SGC, n is NCMC for normal distribution.

Table 7.14 Evaluation of WMN-GA system.

Number of mesh routers	Normal Distribution SGC	Normal Distribution NCMC
16	16	44
20	20	46
24	24	47
28	28	48
32	32	48



(a) Number of mesh routers: 16

(b) Number of mesh routers: 32

Fig. 7.34 SGC and NCMC vs. number of generations for normal distribution.

Table 7.15 Simulation parameters for ns-3.

Parameters	Values
Area Size	640[m]×640[m]
Distributions of mesh clients	Normal
Number of mesh routers	16
Number of mesh clients	48
PHY protocol	IEEE 802.11b
Propagation loss model	Log-distance Path Loss Model
Propagation delay model	Constant Speed Model
MAC protocols	DCF, EDCA
Routing protocol	OLSR
Transport protocol	UDP
Application type	CBR
Packet size	1024 [Bytes]
Number of source nodes	10, 20, 30
Number of destination node	1
Transmission current	17.4 [mA]
Receiving current	19.7 [mA]
Simulation time	600 [sec]

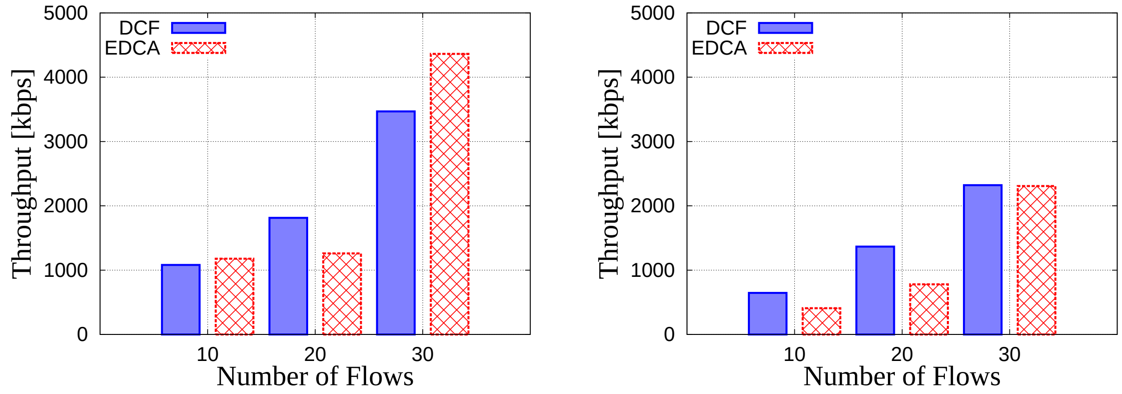
7.4.2 Simulation Results

We used the throughput, delay, jitter and fairness index metrics to evaluate the performance of WMNs for two architectures and considering MAC protocols and normal distributions.

In Fig. 7.35, we show the simulation results of throughput. The throughput of I/B WMN is higher than Hybrid WMN architecture.

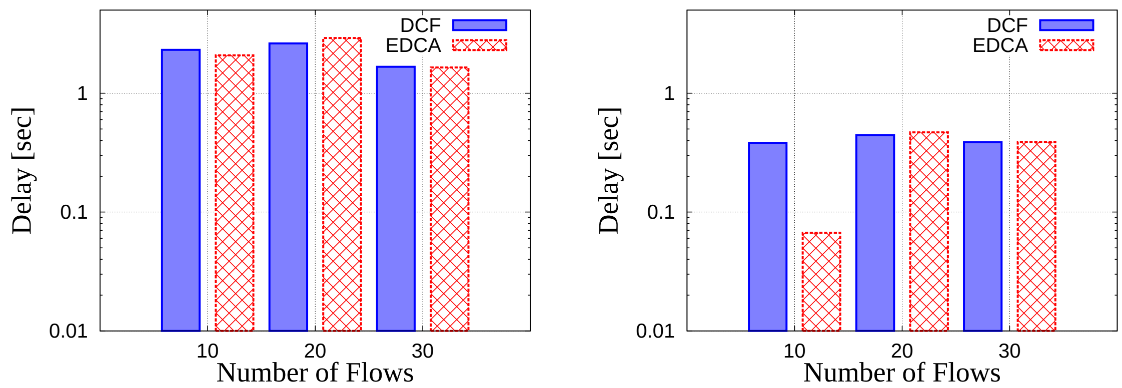
In Fig. 7.36 and Fig. 7.37, the delay and jitter of Hybrid WMN is a lower compared with I/B WMN.

In Fig. 7.38, we show the fairness index. The fairness index of 10 and 20 flows is higher than 30 flows for both WMN architectures. In I/B architecture the fairness index of DCF is higher than EDCA. However, for Hybrid WMN, the fairness index of EDCA is higher than DCF.



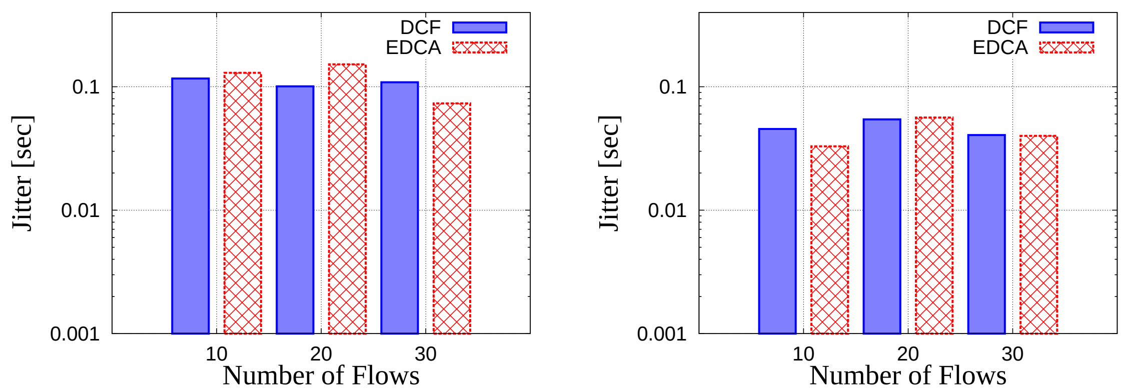
(a) I/B WMN (b) Hybrid WMN

Fig. 7.35 Results of average throughput.



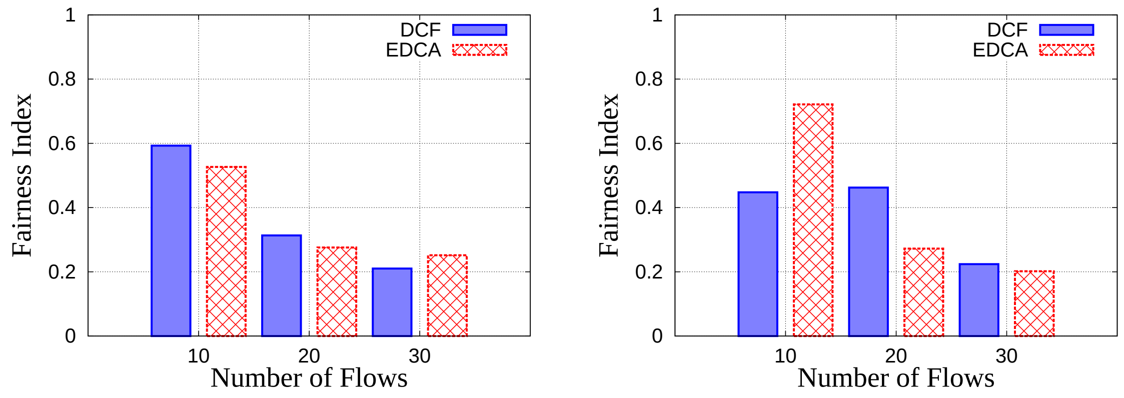
(a) I/B WMN (b) Hybrid WMN

Fig. 7.36 Results of average delay.



(a) I/B WMN (b) Hybrid WMN

Fig. 7.37 Results of average jitter.



(a) I/B WMN (b) Hybrid WMN

Fig. 7.38 Results of fairness index.

7.5 Simulation Scenario 5

In Simulation Scenario 5, we analyze the WMN-GA system data considering different routing protocols and architectures.

7.5.1 Simulation Settings

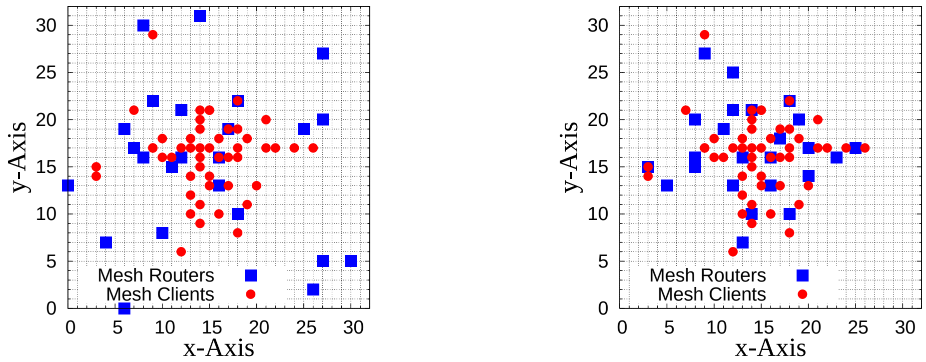
The input parameters of WMN-GA system are shown in Table 7.16. In Fig. 7.39, Fig. 7.40, Fig.

Table 7.16 Input parameters of WMN-GA system.

Parameters	Values
Number of clients	48
Number of routers	16, 20, 24, 28, 32
Grid width	32 [units]
Grid height	32 [units]
Independent runs	10
Number of generations	200
Population size	64
Selection method	Linear Ranking
Crossover rate	80 [%]
Mutate method	Single
Mutate rate	20 [%]
Distribution of clients	N, U, E, W

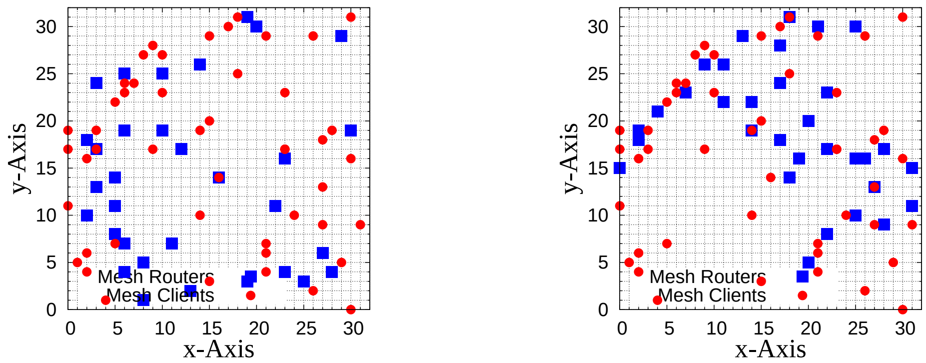
7.41 and Fig. 7.42, we show the location of mesh routers and mesh clients for first generations and the optimized topologies generated by WMN-GA system for Normal (N), Uniform (U), Exponential (E) and Weibull (W) distributions, respectively.

In Fig. 7.43, Fig. 7.44, Fig. 7.45 and Fig. 7.46 are shown the simulation results of Size of Giant Component (SGC) vs. number of generations. After few generations, all mesh routers are connected with each other.



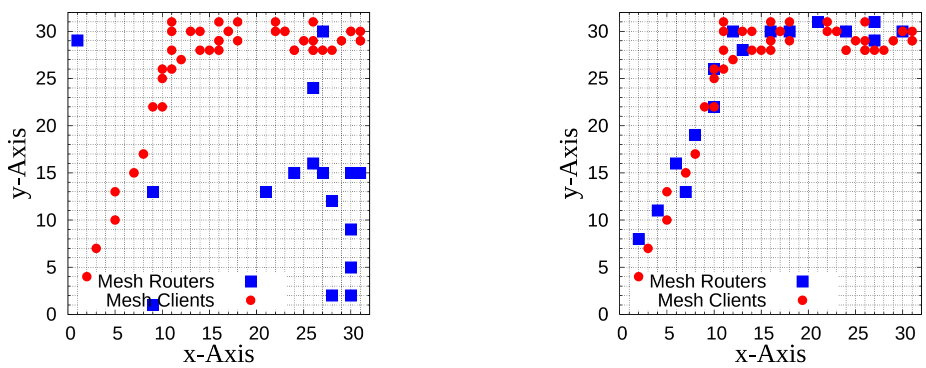
(a) Number of generations: 1 (8, 12) (b) Number of generations: 200 (24, 47)

Fig. 7.39 (m, n): m is SGC, n is NCMC of normal distribution.



(a) Number of generations: 1 (15, 18) (b) Number of generations: 200 (32, 35)

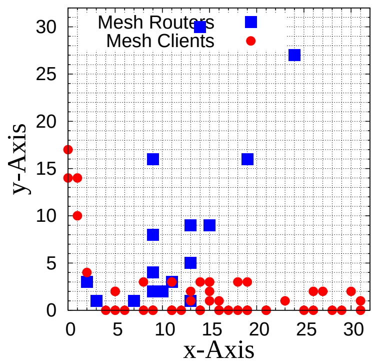
Fig. 7.40 (m, n): m is SGC n is NCMC of uniform distribution.



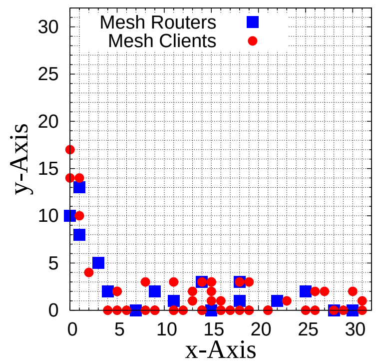
(a) Number of generations: 1 (12, 11) (b) Number of generations: 200 (16, 47)

Fig. 7.41 (m, n): m is SGC, n is NCMC (Exponential distribution).

Then, we optimize the position of mesh routers in order to cover as many mesh clients as possible. We consider Normal and Uniform distributions of mesh clients, which are similar with

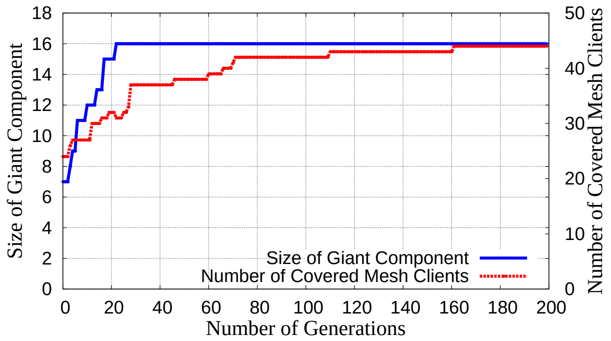


(a) Number of generations: 1 (12, 17)

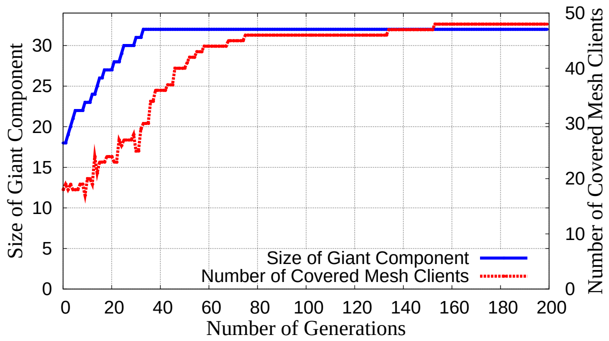


(b) Number of generations: 200 (16, 47)

Fig. 7.42 (m, n): m is SGC, n is NCMC (Weibull distribution).

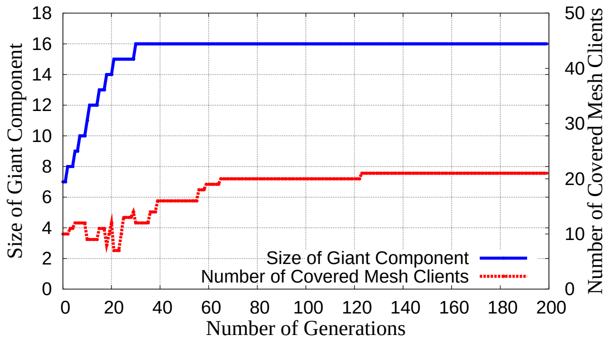


(a) Number of mesh routers: 16

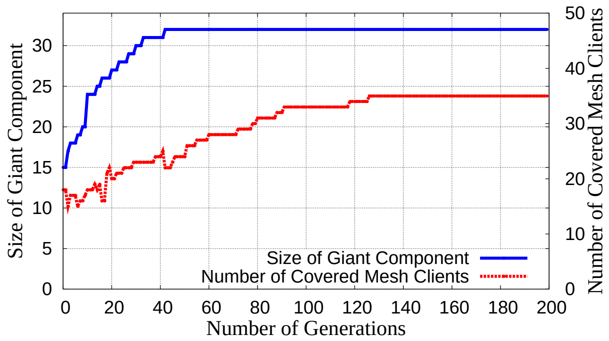


(b) Number of mesh routers: 32

Fig. 7.43 SGC and NCM vs. number of generations for Normal Distribution.



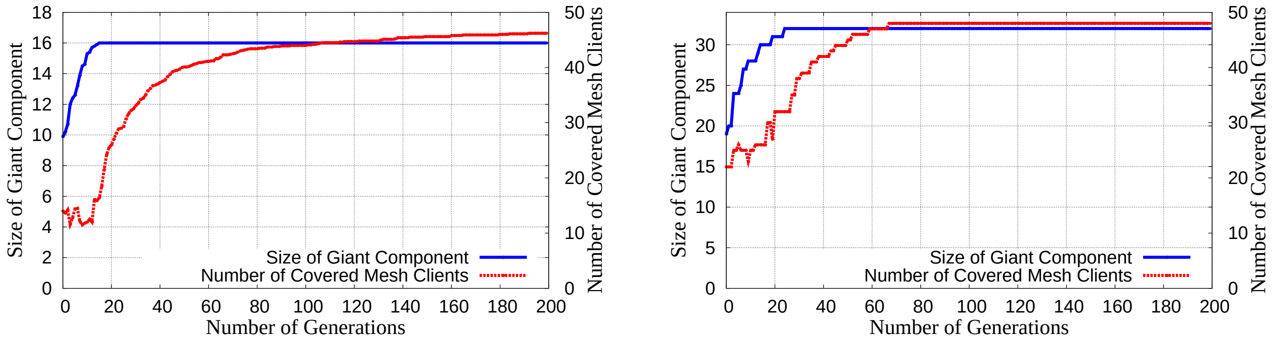
(a) Number of mesh routers: 16



(b) Number of mesh routers: 32

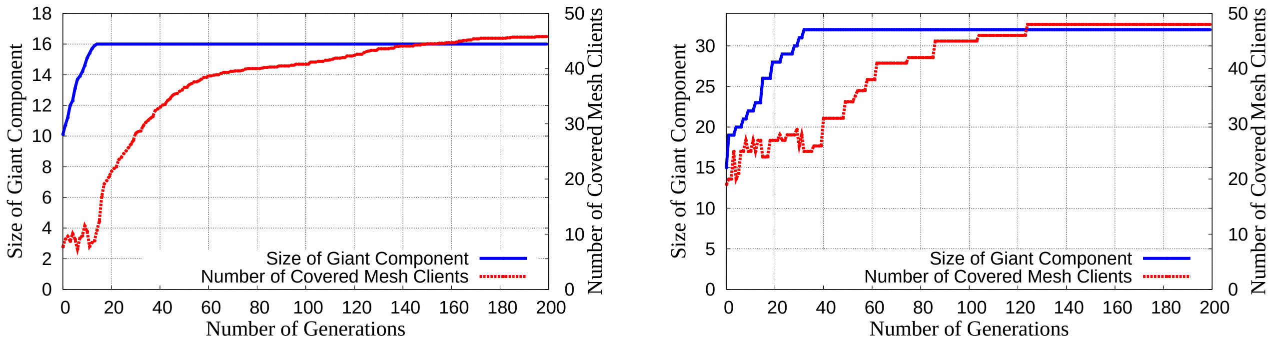
Fig. 7.44 SGC and NCM vs. number of generations for Uniform Distribution.

nodes concentrated in event-site environment, while Exponential and Weibull distributions of mesh clients, which are similar with mesh clients concentrated in hot-spot environment. The simulation results of SGC and Number of Covered Mesh Clients (NCMC) are shown in Table. 7.17.



(a) Number of mesh routers: 16 (b) Number of mesh routers: 32

Fig. 7.45 SGC and NCM vs. number of generations for Exponential Distribution.



(a) Number of mesh routers: 16 (b) Number of mesh routers: 32

Fig. 7.46 SGC and NCM vs. number of generations for Weibull Distribution.

Table 7.17 Evaluation results of WMN-GA system.

Number of	Normal Distribution		Uniform Distribution		Exponential Distribution		Weibull Distribution	
mesh routers	SGM	NCN	SGC	NCM	SGM	NCN	SGC	NCM
16	16	44	16	21	16	47	16	47
20	20	46	20	22	20	48	20	48
24	24	47	24	27	24	48	24	48
28	28	48	28	33	28	48	28	48
32	32	48	32	35	32	48	32	48

We conduct simulations using ns-3 simulator. The simulations in ns-3 are done for number of generations 1 and 200. The area size is considered 640 [m]×640 [m] (or 32 units×32 units) and the number of mesh routers is from 16 to 32. We used HWMP and OLSR routing protocols and sent multiple CBR flows over UDP. The pairs source-destination are the same for all simulation scenarios. Log-distance path loss model and constant speed delay model are used for the simulation and other parameters are shown in Table 7.18.

Table 7.18 Simulation parameters for ns-3.

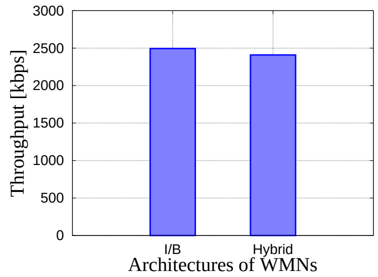
Parameters	Values
`Node settings`	
Area size	640[m]×640[m]
Number of mesh routers	8, 16, 32, 64, 128
Distributions of mesh clients	N, U, E, W
Number of mesh clients	24, 48, 96, 192, 384
`Network architecture settings`	
Propagation loss model	Log-distance Path Loss Model
Propagation delay model	Constant Speed Model
PHY	IEEE 802.11a
MAC	IEEE 802.11s
Queue management algorithm	ECDA, DCF
Rate control algorithm	OFDM 6 [Mbps]
Routing protocol	HWMP, OLSR
Transport protocol	UDP, TCP
Types of network architecture	I/B WMN, Hybrid WMN
`Flow settings`	
Application type	CBR
Transmission rate	128, 256, 512, 1024, 2048 [kbps]
Packet size	128, 256, 512, 1024, 2048 [Bytes]
Number of source nodes	1, 10, 20, 30, 40
Number of destination nodes	1
`Energy settings`	
Transmission energy	17.4 [mA]
Receiving energy	19.7 [mA]
Simulation time	600 [sec]

7.5.2 Discussion of Simulation Scenario

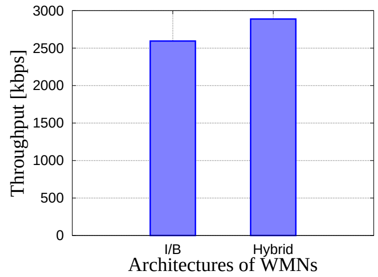
We used the throughput, delay and energy metrics to evaluate the performance of WMNs using HWMP and OLSR protocols for Normal, Uniform, Exponential and Weibull distributions and I/B WMN and Hybrid WMN architectures. In Fig. 7.47, we show the simulation results of HWM protocol throughput for Normal, Uniform, Exponential and Weibull distributions, respectively. For Normal and Exponential distributions, the throughput of I/B WMN is a little bit higher than Hybrid architecture. For Weibull distribution the throughput is almost the same for both WMN architectures. However, for Uniform distribution the throughput of Hybrid WMN is higher than I/B WMN. This is because for Normal and Exponential distributions, the mesh routers are concentrated in the grid area, thus there are many collisions and the network becomes congested.

In Fig. 7.48, we show the delay for four distributions considering HWM protocol. For Normal and Weibull distributions (see Fig. 7.48(a) and Fig. 7.48(d)), the delay is almost the same. However, for Uniform distribution the delay of Hybrid WMN is lower than I/B WMN as shown in Fig. 7.48(b). This is because in Hybrid WMN also the mesh client communicate between each other. But, for Exponential distribution (see Fig. 7.48(c)) the delay of I/B WMN is low compared with Hybrid WMN.

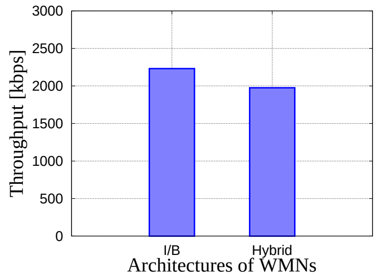
In Fig. 7.49, we show the remaining energy of HWM protocol for both WMN architectures and four distributions, respectively. For Normal and Exponential distributions, the energy decreases sharply, because the mesh routers are concentrated in the grid area and many packets collide with each other. For Weibull distribution, the energy decrease almost the same for both WMN



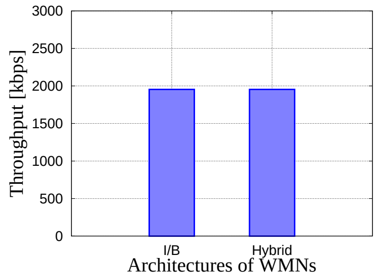
(a) Normal distribution



(b) Uniform distribution

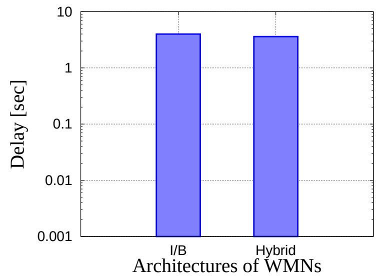


(c) Exponential distribution

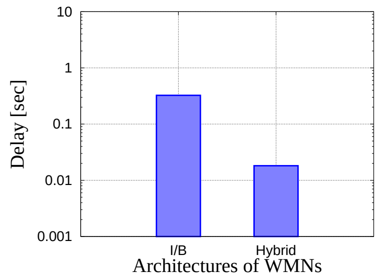


(d) Weibull distribution

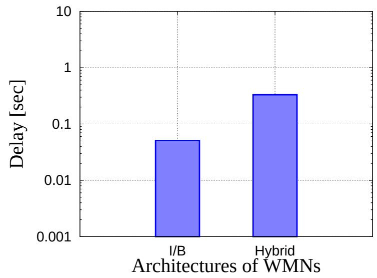
Fig. 7.47 Results of average throughput for WMNs using HWMP of different distributions.



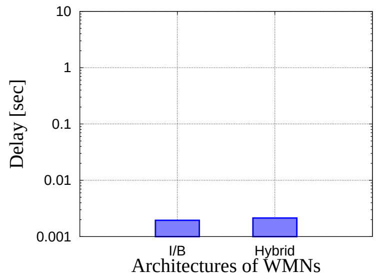
(a) Normal distribution



(b) Uniform distribution



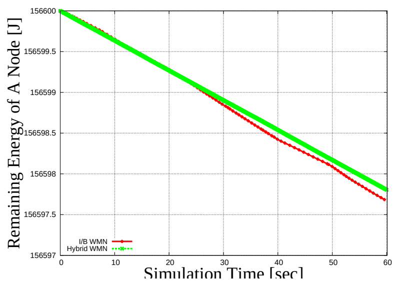
(c) Exponential distribution



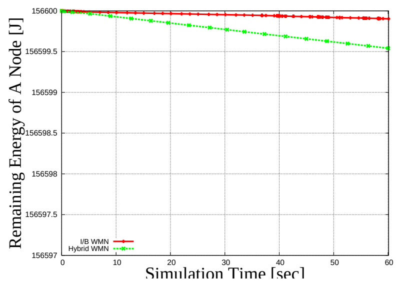
(d) Weibull distribution

Fig. 7.48 Results of average delay for WMNs using HWMP of different distributions.

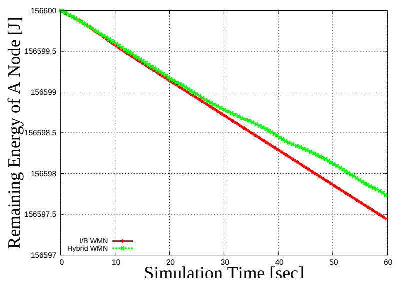
architectures. For Uniform distribution, the remaining energy of I/B WMN is higher than Hybrid



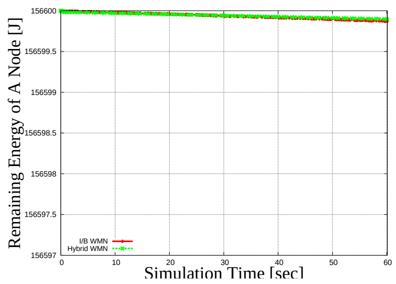
(a) Normal distribution



(b) Uniform distribution



(c) Exponential distribution



(d) Weibull distribution

Fig. 7.49 Results of remaining energies for WMNs using HWMP of different distributions.

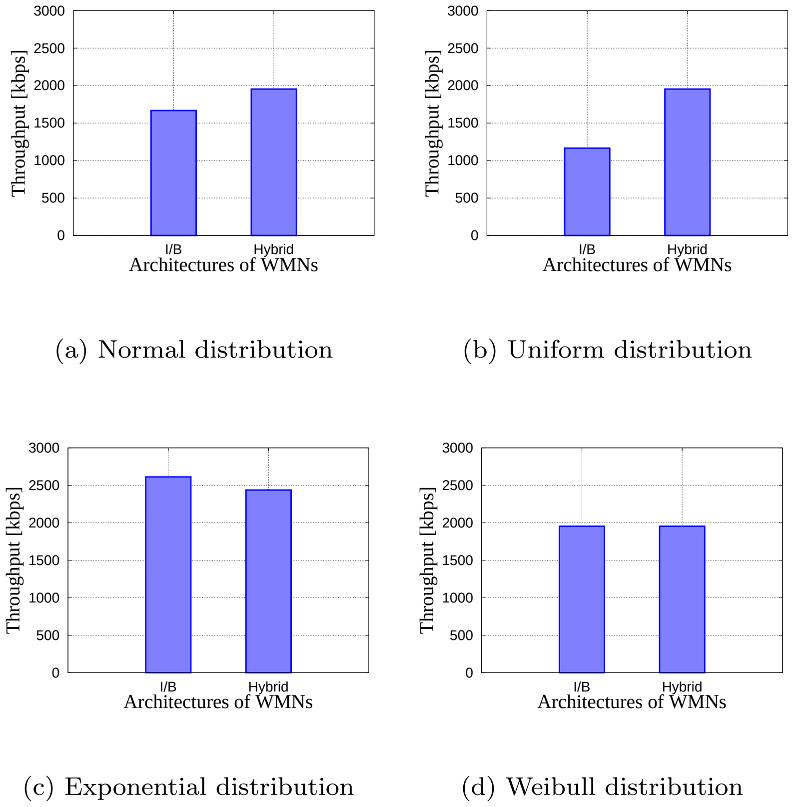
WMN. This is because in Hybrid WMN, there are three communications: mesh client to mesh client, mesh router to mesh router and mesh client to mesh router.

From the simulation results, we conclude that for HWM protocol the throughput of Uniform distribution is higher than other distributions. But, the delay and remaining energy is better for Weibull distribution.

In Fig. 7.50, we show the simulation results of OLSR protocol throughput for Normal, Uniform, Exponential and Weibull distributions, respectively. For Normal and Uniform distributions, the throughput of Hybrid WMN is higher than I/B WMN architecture. But, for Exponential distribution, the throughput of I/B WMN is higher than Hybrid WMN architecture. For Weibull distribution, the throughput is almost the same for both WMN architectures.

In Fig. 7.51, we show the delay for four distributions considering OLSR protocol. In Fig. 7.51(a) and Fig. 7.51(b), the delay of Hybrid WMN is a lower compared with I/B WMN. For Exponential and Weibull distributions (see Fig. 7.51(c) and Fig. 7.51(d)), the delay is almost the same for both distributions. However, the delay of Weibull distribution is lower than other distributions.

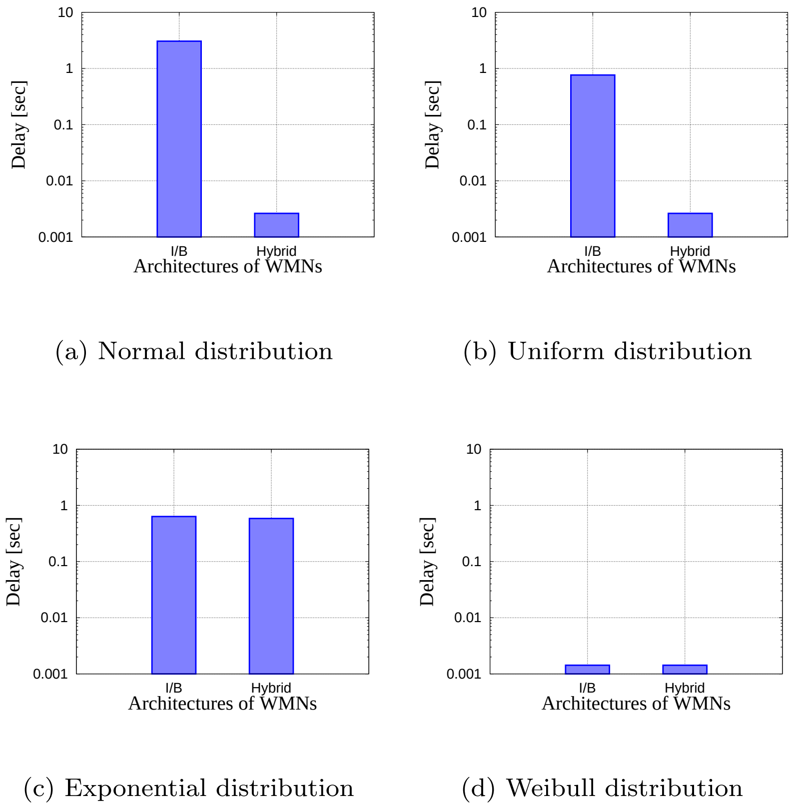
In Fig. 7.52, we show the remaining energy of OLSR protocol for Normal, Uniform, Exponential and Weibull distributions. For Normal distribution, the remaining energy of Hybrid WMN is higher than I/B WMN. While for Exponential distribution, the remaining energy of I/B WMN is higher than Hybrid WMN. For Uniform and Weibull distributions, the energy decrease is almost the same for both WMN architectures. Also, for Uniform and Weibull distributions, the remaining energy is higher compared with Normal and Exponential distributions.



(a) Normal distribution (b) Uniform distribution

(c) Exponential distribution (d) Weibull distribution

Fig. 7.50 Results of average throughput for WMNs using OLSR of different distributions.

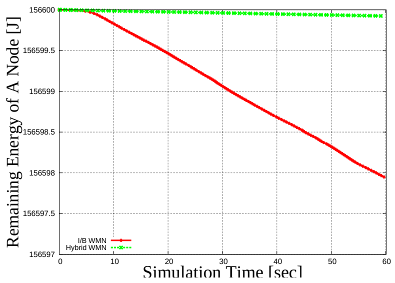


(a) Normal distribution (b) Uniform distribution

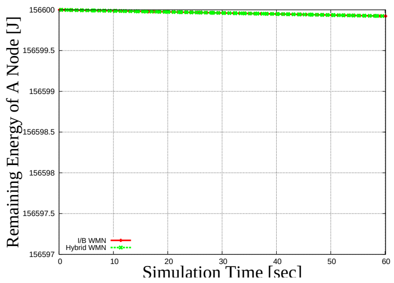
(c) Exponential distribution (d) Weibull distribution

Fig. 7.51 Results of average delay for WMNs using OLSR of different distributions.

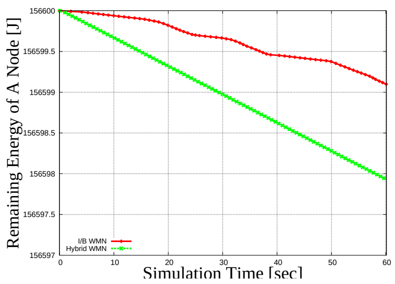
From the simulation results, we conclude that the throughput of Exponential distribution for OLSR protocol is better than other distributions. But, the delay and remaining energy is better for Weibull distribution.



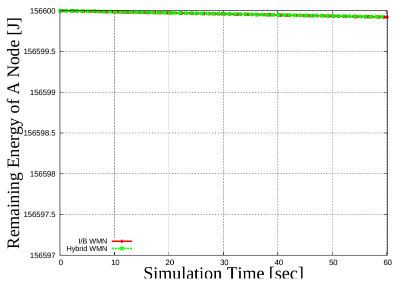
(a) Normal distribution



(b) Uniform distribution



(c) Exponential distribution



(d) Weibull distribution

Fig. 7.52 Results of remaining energies for WMNs using OLSR of different distributions.

Chapter 8

Testbed Implementation

8.1 Testbed Design and Implementation

8.1.1 Description

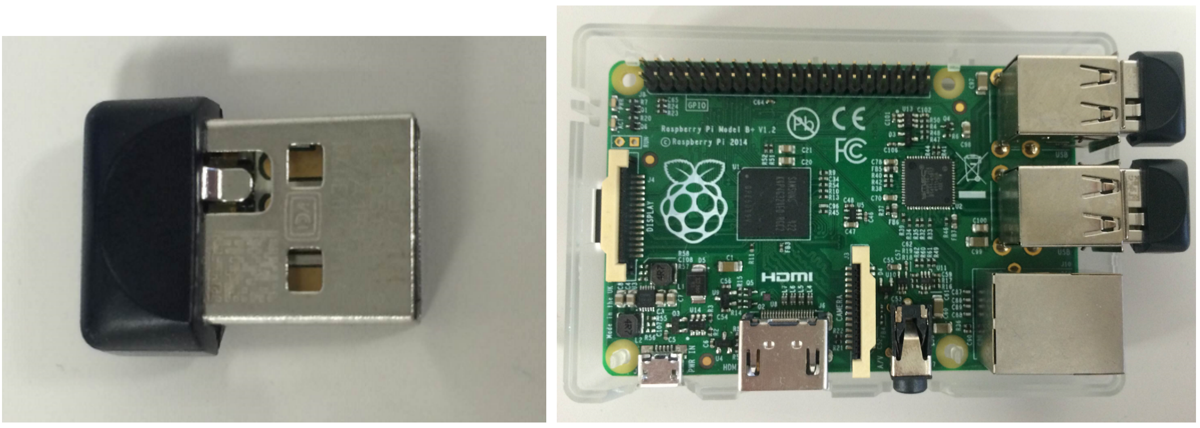
The testbed consists of five Raspberry Pi B+ [115]. The Raspberry Pi is a credit card-sized single-board computer developed by the Raspberry Pi Foundation. The machines operate on Raspbian version Debian 7.8 with kernel 3.18.11 [116] and OpenWRT version Chaos Calmer (r45558) with kernel 3.18.11 [117] suitably modified in order to support the wireless cards. The OpenWrt is an embedded operating system based on the Linux kernel and primarily used on embedded devices to route network traffic. The main components are the Linux kernel, util-linux, uClibc and BusyBox. All components have been optimized for size, to be small enough for fitting into the limited storage and memory available in home routers.

The BUFFALO wireless network devices (Model: WLI-UC-GNM) are usb-based cards (see Fig. 8.1(a) and Fig. 8.1(b)). We verified that the external antenna improves the quality of the first hop link, which is the link connecting the mesh network. The driver can be downloaded from the web site [118–120].

As MAC protocol, we used IEEE 802.11n. The transmission power was set in order to guarantee a coverage radius equal to the maximum allowed geographical distance in the network. In regard to the interference, it is worth noting that, during our tests, almost all the IEEE 802.11 spectrum had been used by other Access Points (APs) disseminated within the campus. In general, the interference from other access points is a non-controllable parameter.

In our testbed, we have two systematic traffic sources we could not eliminate: the control traffic and the other wireless APs interspersed within the campus. The control traffic is due to the `ssh` program, which is used to remotely start and control the measurement software on the source node. The other traffic source brings interferences occupying the available bandwidth, which is typical in an academic scenario.

To generate the traffic between nodes, we used Iperf, which is an network testing tool [121]. The Iperf was originally developed by NLANR/DAST as a modern alternative for measuring TCP and UDP bandwidth performance. The Iperf allows the tuning of various parameters and UDP characteristics. The Iperf reports bandwidth, jitter and datagram loss.



(a) WLI-UC-GNM (b) Attached WLI-UC-GNM to Raspberry Pi B+

Fig. 8.1 Snapshot of the testbed.

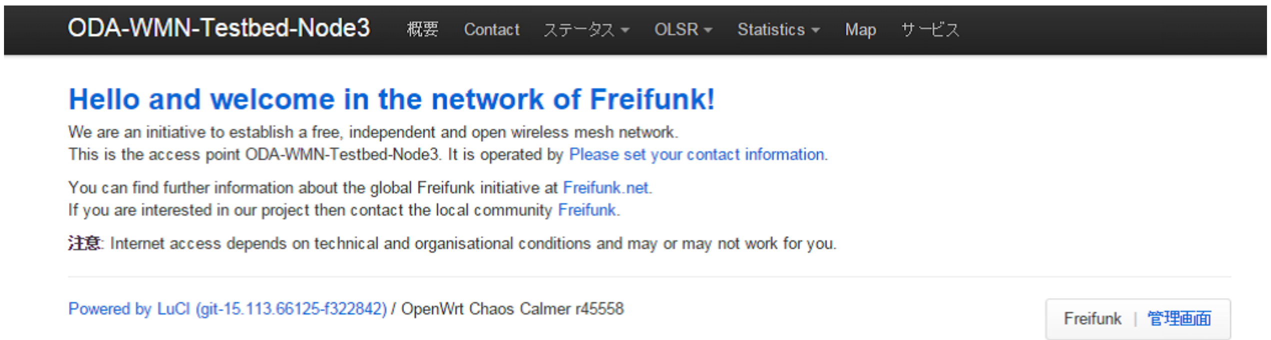


Fig. 8.2 Testbed interface.

8.1.2 Testbed Interface

Upon the first implementation of the testbed, all the parameters settings and editing were done by using command lines of bash shell (terminal), which resulted in many misprints and the experiments were repeated many times. In order to make the experiments easier, we implemented a Graphical User Interface (GUI) interface. OpenWrt is configured using a command-line interface (ash shell), or a web interface (LuCI) [122] (see Fig. 8.2). There are many software packages available for installation via the opkg package management system.

We implemented many parameters in the interface such as transmission duration, number of trials, source address, destination address, packet rate, packet size and topology setting function. We can save the data for these parameters in a text file and can manage in a better way the experimental conditions. Moreover, we implemented collection function of experimental data in order to make easier the experimenter's work.

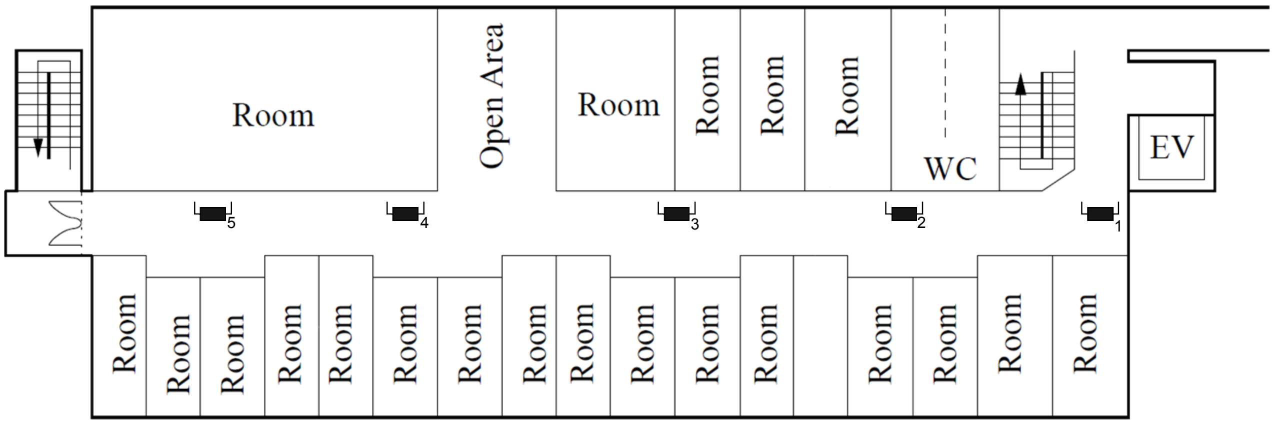


Fig. 8.3 Experimental environment for scenario 1.

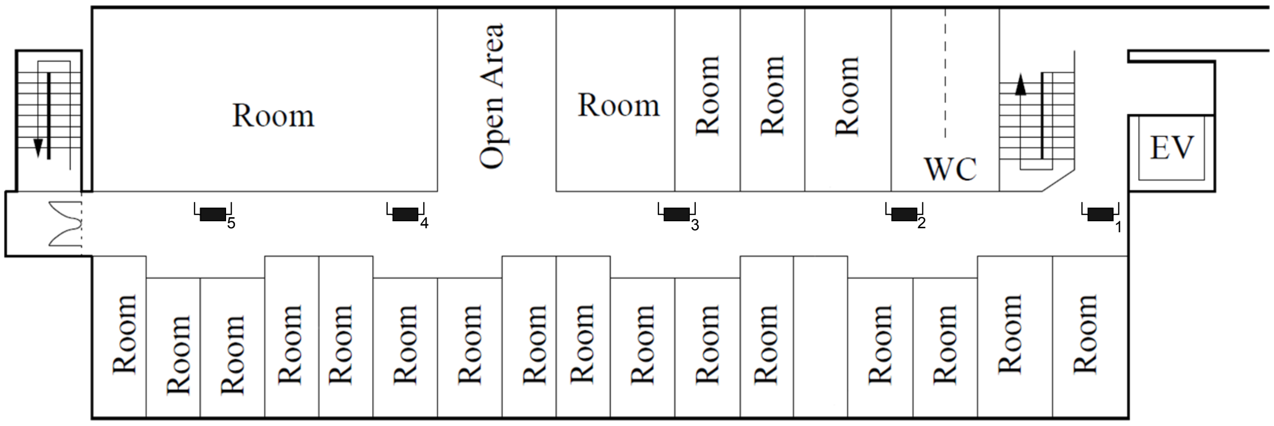
8.1.3 Testbed Environment

Our testbed provides an experimental platform for evaluating protocols and algorithms using realistic parameters. In this testbed, we can implement different topology scenarios and analyze different routing protocols considering different metrics. In this work, we take the following considerations.

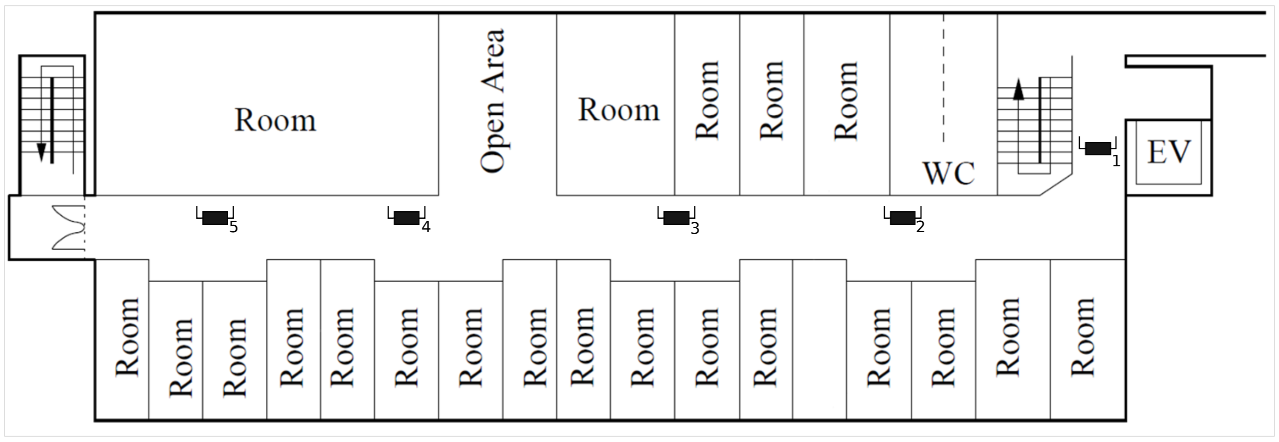
- The experiments are conducted in indoor environment inside our university campus (see Fig. 8.3, Fig. 8.4, Fig. 8.5 and Fig. 8.6).
- We analysed our network for many experimental scenarios, such as static scenarios, where all nodes are static and moving scenarios, where source and destination nodes, respectively, are mobile.
- In moving scenarios, the mobile nodes move at regular speed and when they arrive at turning points, they stop for about three seconds.
- We discuss the effect of multihop and mobility using OLSR routing protocol.

8.2 Experimental Environment

In Experimental Scenario 1, we consider an indoor environment, different OSs and LoS scenario (see Fig. 8.3). In Experimental Scenario 2, we consider an indoor environment and LoS and NLoS scenarios (see Fig. 8.4). In Experimental Scenario 3, we consider mobile mesh node scenario in indoor environment (see Fig. 8.5). In Experimental Scenario 4, we consider an indoor environment, distributed concurrent processing and LoS Scenario (see Fig. 8.6).



(a) LoS scenario



(b) NLoS scenario

Fig. 8.4 Experimental environment for scenario 2.

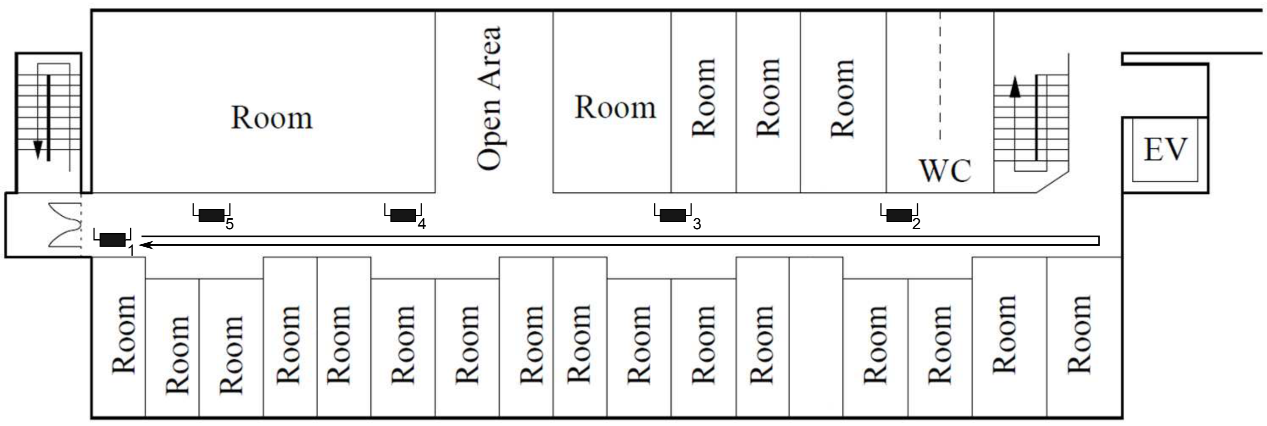


Fig. 8.5 Experimental environment for scenario 3.

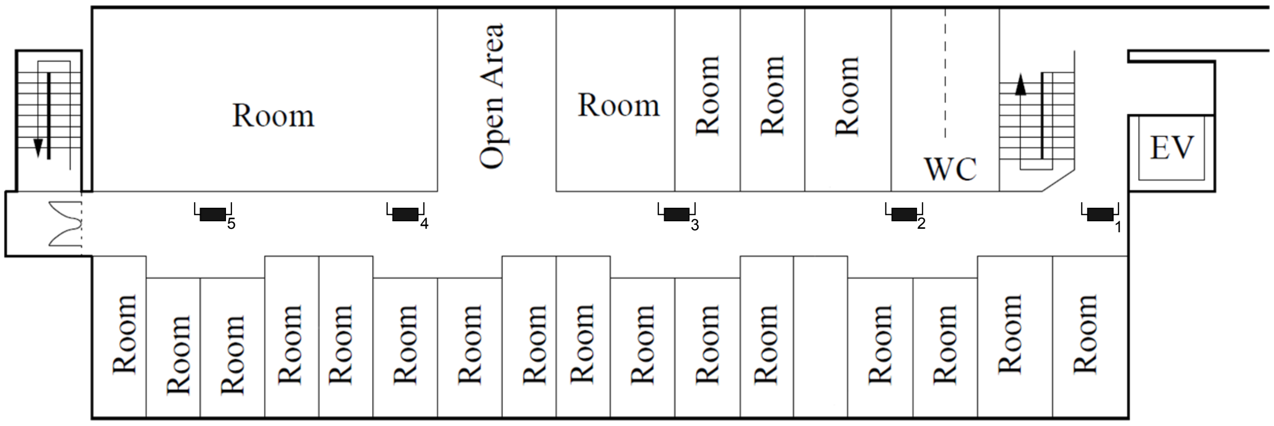


Fig. 8.6 Experimental environment for scenario 4.

Chapter 9

Experimental Results

9.1 Experimental Scenario 1

In Experimental Scenario 1, we analyze experimental results of a WMN testbed in indoor environment considering different OSs and LoS scenario.

9.1.1 Scenario Description

Our testbed is composed of five Raspberry Pi [115]. The Raspberry Pi is a credit card-sized single-board computer developed by the Raspberry Pi Foundation. The operating systems mounted on these machines are Raspbian version Debian 7.8 with kernel 3.18.11 [116] and OpenWRT version Chaos Calmer (r45558) with kernel 3.18.11 [117].

Table 9.1 Experimental parameters for scenario 1.

Functions	Values
OS	Raspbian, OpenWRT
Number of trials	100
Duration	80 [sec]
Number of mesh nodes	5
MAC	IEEE 802.11n
Routing protocol	OLSR
OLSRd	OLSRd 0.6.7.1
Transport protocol	UDP
Flow type	CBR
Bit rate	512 [Kbps]

Table 9.2 OLSRd parameters.

Functions	Values
Hello interval time	5.0 [sec]
Hello validity	40.0 [sec]
TC interval time	2.0 [sec]
TC validity	256.0 [sec]
MID interval time	18.0 [sec]
MID validity	324.0 [sec]
HNA interval time	18.0 [sec]
HNA validity	108.0 [sec]

(a) Node 1

(b) Node 2

(c) Node 3

(d) Node 4

(e) Node 5

Fig. 9.1 Snapshot of nodes in the testbed for scenario 1.

In our WMN testbed (see Fig. 9.1), we have two systematic background or interference traffic we could not eliminate: the control traffic and the other wireless APs interspersed within the campus. The experimental parameters for the LoS scenario are shown in Table 9.1 and Table 9.2.

9.1.2 Experimental Results

9.1.2.1 Experimental Settings and Parameters

The experimental parameters are shown in Table 9.1 and Table 9.2. We collected data for five metrics: throughput, PDR, one-way delay, jitter and hop count. These data are collected by using the Iperf, which is an network testing tool [121].

9.1.2.2 Experimental Measurements

For evaluation, we used a single flow from node 1 to node 5. In Fig. 9.2, we show the experimental results for LoS scenario. Source and destination nodes are always 1 and 5, respectively. In Fig. 9.2(a), we show the average throughput from node 1 to node 5. The average throughput value of Raspbian and OpenWRT is 449 and 494 [Kbps], respectively. Fig. 9.2(b) shows that the PDR of Raspbian and OpenWRT is 88 and 96 [%], respectively. The average hop count of Raspbian and OpenWRT shown in Fig. 9.2(c) indicates that the communication is done for 1 hop, respectively. In Fig. 9.2(d), we show the average delay of Raspbian and OpenWRT. We can see that the delay is lower than 0.08 and 0.009 [sec], respectively. In Fig. 9.2(e), we show the average jitter of Raspbian and OpenWRT. We can see that the jitter is lower than 0.08 and 0.011 [sec], respectively.

The experimental results show that the testbed mounted in OpenWRT has better results than Raspbian.

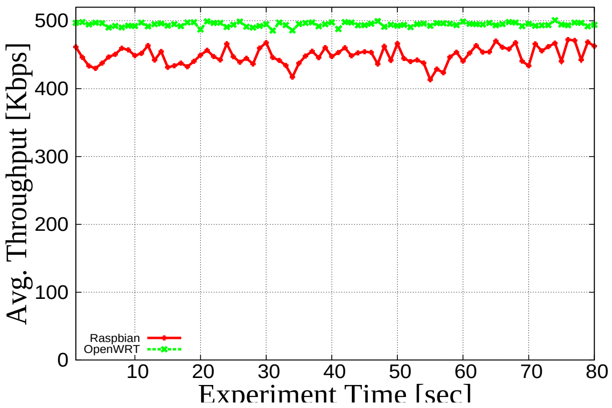
9.2 Experimental Scenario 2

In Experimental Scenario 2, we analyze experimental results of a WMN testbed in indoor environment considering different LoS and NLoS scenarios.

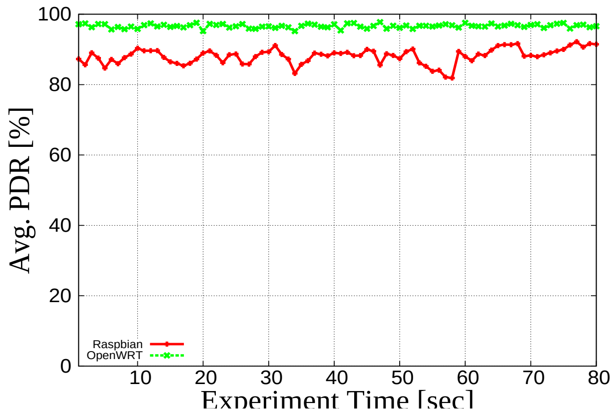
9.2.1 Scenario Description

We implemented this scenario in the indoor environment of our five-floor academic building. Our testbed is composed of five Raspberry Pi. The operating system mounted on these machines is OpenWrt version Chaos Calmer (r45558) with kernel 3.14.18 [117].

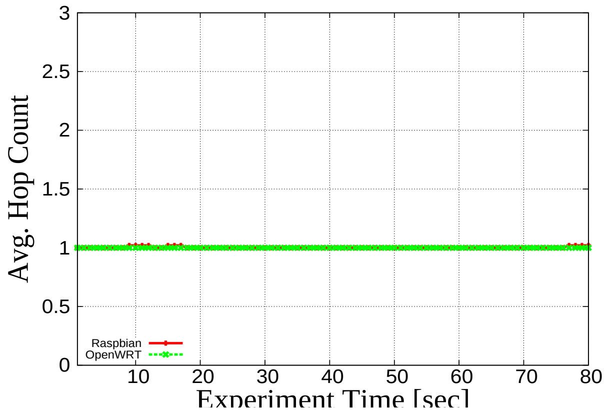
In Fig. 8.4(a), we show the positions of the static nodes in LoS scenario. In Fig. 8.4(b), only the source node is NLoS. In our WMN testbed (see Fig. 9.3), we have two systematic background or interference traffic we could not eliminate: the control traffic and the other wireless APs interspersed within the campus. The experimental parameters for the LoS and NLoS scenarios are shown in Table 9.3. In order to make the experiments easier, we implemented a testbed interface. For the Web User Interface (WUI) we used LuCI.



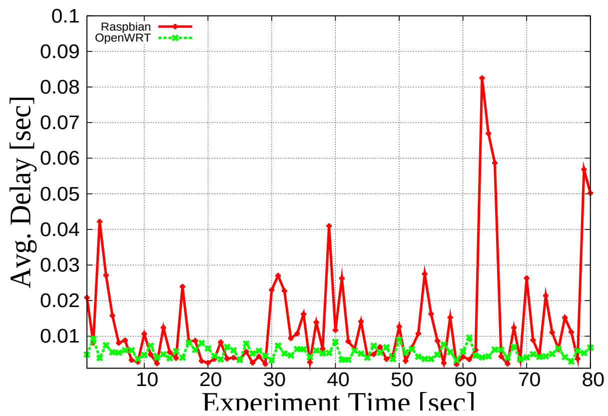
(a) Avg. Throughput



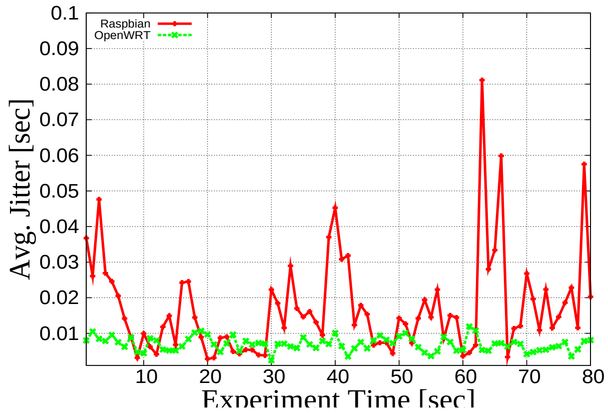
(b) Avg. PDR



(c) Avg. Hop Count



(d) Avg. Delay



(e) Avg. Jitter

Fig. 9.2 Experimental results for 1 → 5.

9.2.2 Experimental Results

9.2.2.1 Experimental Settings and Parameters

The experimental parameters are shown in Table 9.3. We collected data for five metrics: throughput, PDR, hop count, one-way delay and jitter. These data are collected by using the Iperf, which is an network testing tool [121].

(a) Node 1 (LoS)

(b) Node 1 (NLoS)

(c) Node 2

(d) Node 3

(e) Node 4

(f) Node 5

Fig. 9.3 Snapshot of nodes in the testbed for scenario 2.

9.2.2.2 Experimental Measurements

For evaluation, we used a single flow from node 1 to node 5. In Fig. 9.4, we show the experimental results. Source and destination nodes are always 1 and 5, respectively. In Fig. 9.4(a), we show the average throughput from node 1 to node 5. The average throughput value of LoS and NLoS is 494 and 473 [Kbps], respectively. Fig. 9.4(b) shows that the PDR of LoS and NLoS is 96 and 92

Table 9.3 Experimental parameters for scenario 2.

Functions	Values
Number of trials	100
Duration	80 [sec]
Number of mesh nodes	5
MAC	IEEE 802.11n
Routing protocol	OLSRd
OLSRd	OLSRd 0.6.7.1
Transport protocol	UDP
Flow type	CBR
Bit rate	512 [Kbps]

[%], respectively. The average hop count shown in Fig. 9.4(c) indicates that the communication of LoS and NLoS is done for 1 and 3 hops, respectively. In Fig. 9.4(d), we show the average delay of LoS and NLoS. We can see that the delay of LoS and NLoS is lower than 0.009 and 0.05 [sec], respectively. In Fig. 9.4(e), we show the average jitter of LoS and NLoS. We can see that the jitter of LoS and NLoS is lower than 0.011 and 0.06 [sec], respectively.

9.3 Experimental Scenario 3

In Experimental Scenario 3, we analyze experimental results of a WMN testbed in indoor environment considering mobile mesh node scenario.

9.3.1 Scenario Description

Our testbed is composed of five Raspberry Pi [115]. The operating system mounted on these machines is OpenWrt version Chaos Calmer (r45558) with kernel 3.14.18 [117].

In our WMN testbed (see Fig. 9.5), we have two systematic background or interference. traffic we could not eliminate: the control traffic and the other wireless APs interspersed within the campus. The experimental parameters for the LoS scenario are shown in Table 9.4. In order to make the experiments easier, we implemented a testbed interface. For the Web User Interface (WUI) we used LuCI.

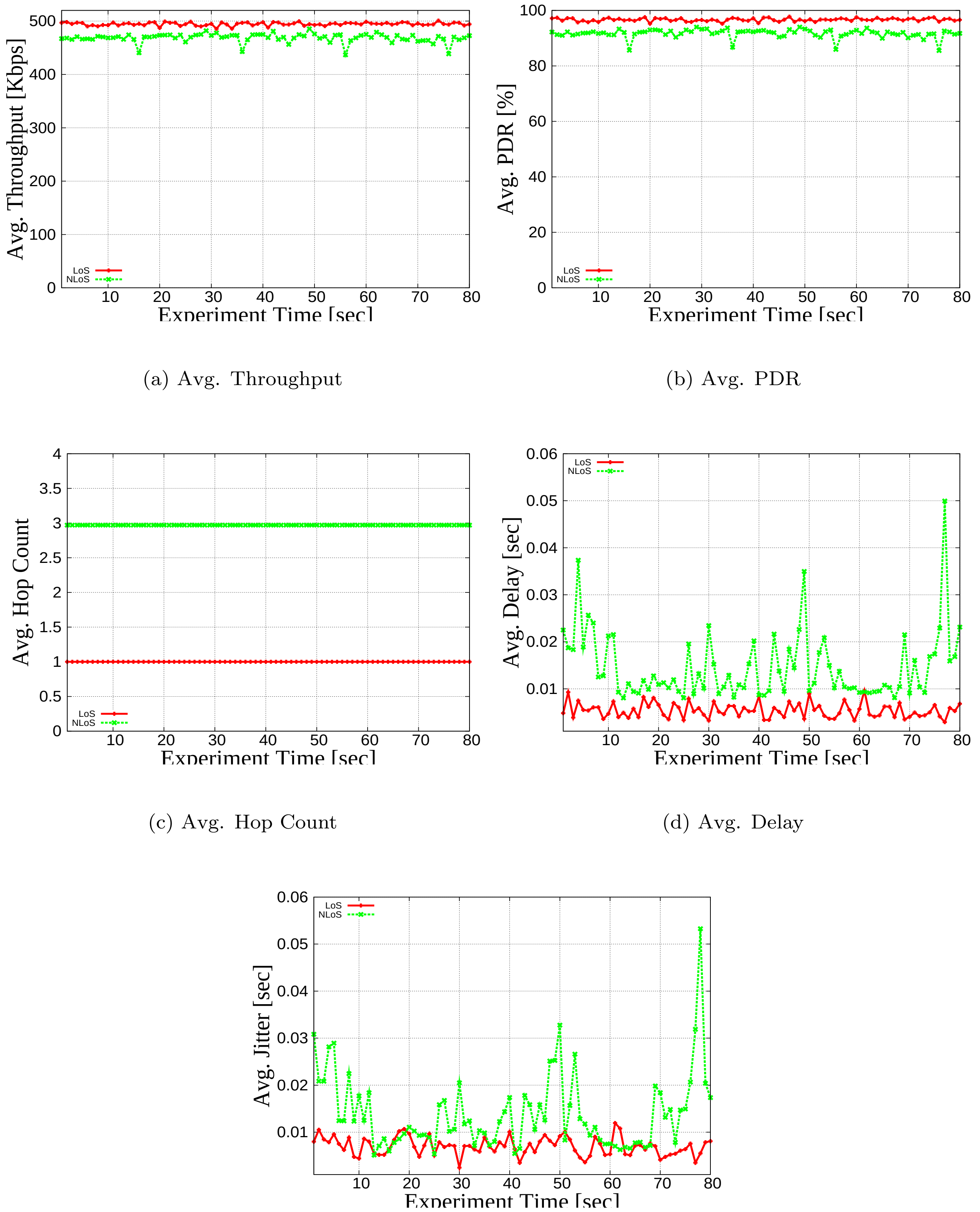
9.3.2 Experimental Results

9.3.2.1 Experimental Settings and Parameters

The experimental parameters are shown in Table 9.4. We collected data for five metrics: throughput, PDR, hop count, one-way delay and jitter. These data are collected by using the Iperf, which is an network testing tool [121].

9.3.2.2 Experimental Measurements

For evaluation, we used a single flow from node 1 to node 5. In Fig. 9.6, we show the experimental results for LoS scenario. Source and destination nodes are always 1 and 5, respectively. In Fig. 9.6(a), we show the average throughput from node 1 to node 5. The average throughput value is 457 [Kbps]. Fig. 9.6(b) shows that the PDR is 90 [%]. The average hop count shown in



(a) Avg. Throughput

(b) Avg. PDR

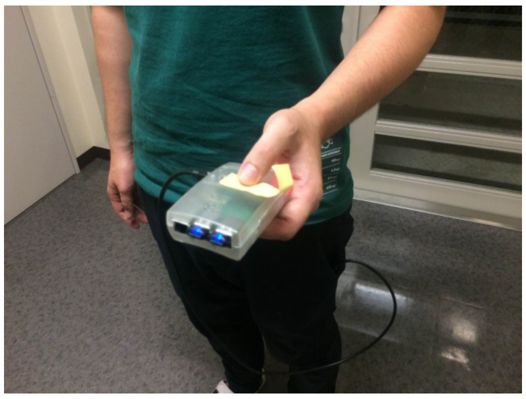
(c) Avg. Hop Count

(d) Avg. Delay

(e) Avg. Jitter

Fig. 9.4 Experimental results for 1 $\rightarrow$ 5.

Fig. 9.6(c) indicates that the communication is done for 2 hops. In Fig. 9.6(d) and Fig. 9.6(e), we show the average delay and jitter. We can see that the delay and jitter is lower than 0.05 and 0.06 [sec], respectively.



(a) Node 1

(b) Node 2

(c) Node 3

(d) Node 4

(e) Node 5

Fig. 9.5 Snapshot of nodes in the testbed for scenario 3.

9.4 Experimental Scenario 4

In Experimental Scenario 4, we analyze experimental results of a WMN testbed in indoor environment considering distributed concurrent processing and LoS Scenario.

Table 9.4 Experimental parameters for scenario 3.

Functions	Values
Number of trials	100
Duration	80 [sec]
Number of mesh nodes	5
MAC	IEEE 802.11n
Routing protocol	OLSR
OLSRd	OLSRd 0.6.8
Transport protocol	UDP
Flow type	CBR
Bit rate	512 [Kbps]

9.4.1 WMN Testbed for Distributed Concurrent Processing

9.4.1.1 Open MPI

Open MPI consists of three abstraction layers which are combined to provide a full featured MPI implementation [123]. Below the user application is the Open MPI layer that presents the application with the expected MPI standard interface. Below that is the Open Run-Time Environment (ORTE) layer that provides a uniform parallel run-time interface regardless of system capabilities. Next is the Open Portable Access Layer (OPAL) that abstracts the peculiarities of a specific system away to provide a consistent interface aiding portability. Below OPAL is the operating system and other standard services running on the local machine.

Open MPI uses the Modular Component Architecture (MCA) to define internal APIs called Frameworks for particular services such as process launch. Each framework contains one or more Components which are specific implementations for a framework (e.g., SLURM and RSH components of the process launch framework). Components can be dynamically selected at runtime.

9.4.1.2 NAS Parallel Benchmarks

The well-documented NAS Parallel Benchmarks (NPB) [124–127] is a suite of parallel workloads designed to evaluate performance of various hardware and software components of a parallel computing system [128]. These benchmarks span different problem sizes, called classes in NPB terminology, and in this paper we use classes B, which are standard for the analysis of single-node systems.

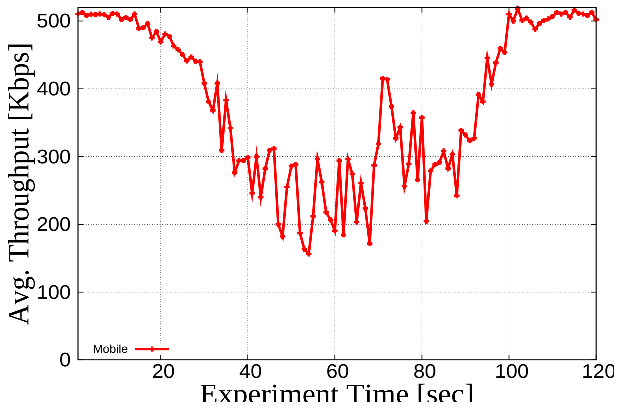
Most of the NAS benchmarks are computational kernels. EP evaluates an integral by means of pseudorandom trials.

9.4.2 Scenario Description

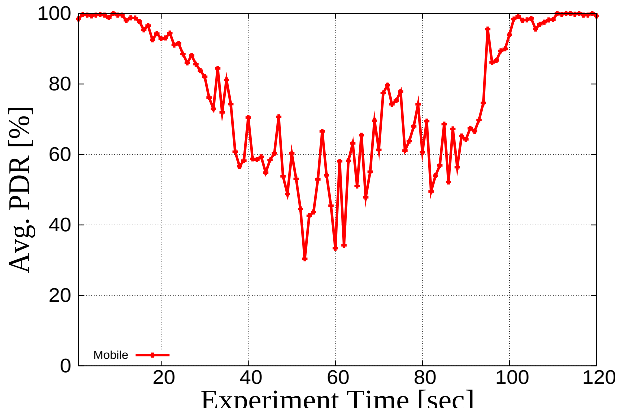
Our testbed is composed of five Raspberry Pi B+ [115]. The operating systems mounted on these machines are Raspbian version Debian 7.8 with kernel 3.18.11 [116].

In Fig. 9.7 are shown snapshots of nodes in the testbed.

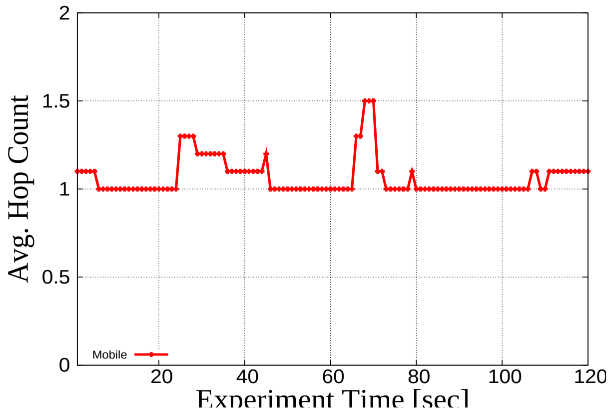
The experimental parameters for the LoS scenario are shown in Table 9.5. In order to make the experiments easier, we implemented a testbed interface.



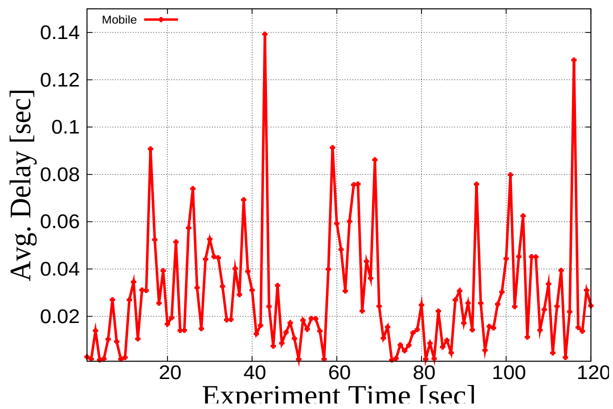
(a) Avg. Throughput



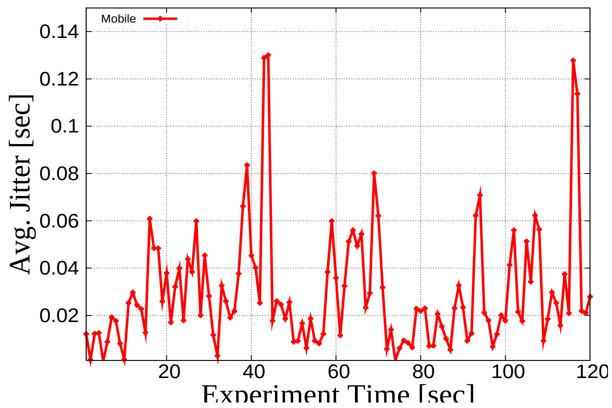
(b) Avg. PDR



(c) Avg. Hop Count



(d) Avg. Delay



(e) Avg. Jitter

Fig. 9.6 Experimental results for 1 → 5.

9.4.3 Experimental Results

9.4.3.1 Experimental Settings and Parameters

We collected data for four metrics: throughput, PDR, hop count, delay, jitter and processing time. These data are collected by using the Iperf [121] and NAS Parallel Benchmark. The Iperf was

(a) Node 1

(b) Node 2

(c) Node 3

(d) Node 4

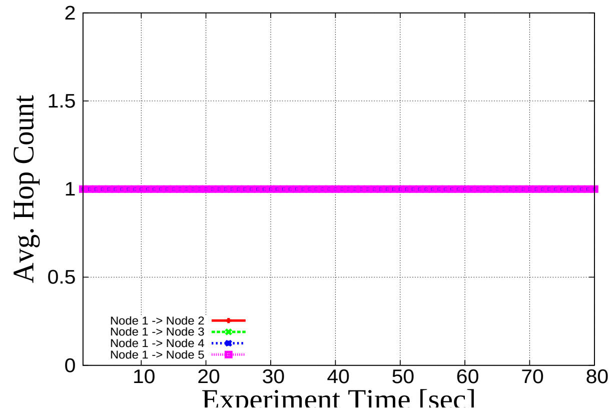
(e) Node 5

Fig. 9.7 Snapshot of nodes in the testbed for scenario 4.

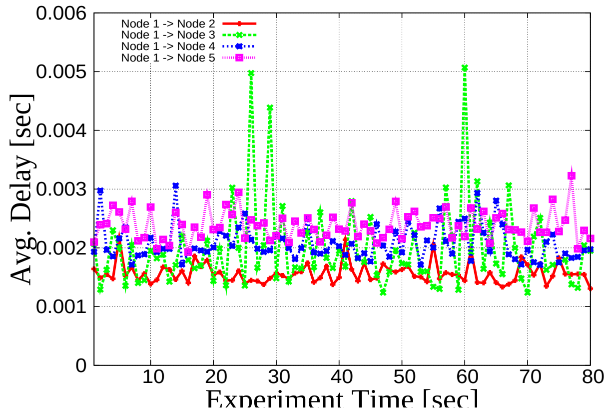
originally developed by NLANR/DAST as a modern alternative for measuring TCP and UDP bandwidth performance. The Iperf allows the tuning of various parameters and UDP characteristics.

Table 9.5 Experimental parameters for scenario 4.

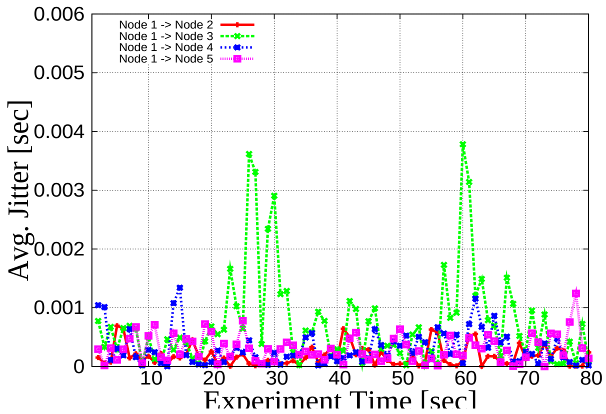
Functions	Values
Number of trials	100
Duration	80 [sec]
Number of mesh nodes	5
MAC	IEEE 802.11n
Routing protocol	OLSR
OLSRd	OLSRd 0.6.8
Transport protocol	UDP
Flow type	CBR
Bit rate	512 [Kbps]



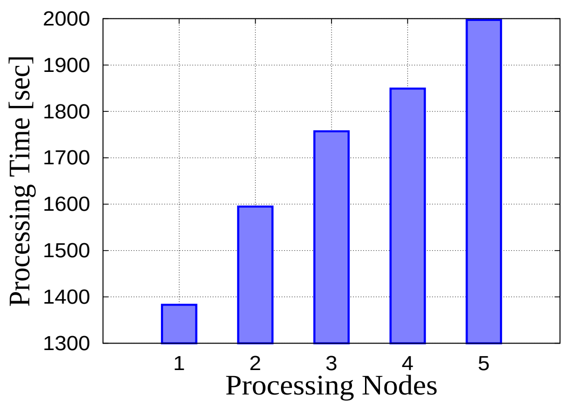
(a) Avg. Hop Count



(b) Avg. Delay



(c) Avg. Jitter



(d) Avg. Processing Time

Fig. 9.8 Experimental results for $1 \rightarrow n$.

9.4.3.2 Experimental Measurements

For evaluation, we used multiple flows from node 1 to node 2, 3, 4 and 5. In Fig. 9.8 and Table 9.6, we show the experimental results for LoS scenario. As shown in Fig. 9.8(a), the communication is done for 1 hop. In Fig. 9.8(b) and Fig. 9.8(c), we show the average delay and jitter. We can see that the average delay and jitter is lower than 0.00506 and 0.00378 [sec], respectively.

Table 9.6 Average for experimental results.

	1 → 2	1 → 3	1 → 4	1 → 5
Hop Count	1	1	1	1
Delay [sec]	0.00157	0.00197	0.002	0.0024
Jitter [sec]	0.00018	0.00075	0.0003	0.00031

In Fig. 9.8(d), we show the experimental results of NAS Parallel Benchmark for processing time. We can see that the processing time is lowest for node 1. This is because the delay time for node 1 is smaller than other nodes.

In this paper, we presented the implementation of testbed for WMNs. We analyzed by experiments the performance of WMNs considering hop count, delay, jitter and processing time metrics. We considered an indoor environment and all terminals are in LoS.

We carried out many experiments using the implemented WMN testbed. We transmitted single CBR flow over UDP. For experiments, we considered OLSRd, IEEE 802.11n and Open MPI. From experiments, we found the following results.

1. The communication is done for almost 1 hops.
2. The delay and jitter is lower than 0.00506 and 0.00378 [sec], respectively.
3. The processing time is lowest for node 1.

In the future, we would like to make extensive experiments for different experimental scenarios.

Chapter 10

Conclusions and Future Work

10.1 Conclusions

In this thesis, we designed and implemented some simulation systems and a testbed for WMNs in order to analyze the performance and compare the results of different routing protocols, mobility models and other environmental parameters. In both approaches we used different models, scenarios and traffic data models. We use our simulation results to improve the experimental environment. We experimented in indoor environment. We also considered fixed and mobile nodes.

Our thesis is constructed by 10 Chapters. In Chapter 1 was presented the background and the motivation of the thesis. In Chapter 2, we introduced general aspects of wireless networks. We discussed wireless architectures and wireless technologies giving advantages and disadvantages for each of them. We presented WMNs in Chapter 3. We discussed problems of WMNs and described the routing protocols and their features. In Chapter 4, we introduced intelligent algorithms and have shown different meta-heuristics methods such as: Genetic Algorithms (GAs), Tabu Search (TS), Hill Climbing (HC) and Simulated Annealing (SA). We presented in details the GA and its operators in Chapter 5. The simulation systems were presented in Chapter 6. We have shown in details the radio propagation models, mobility models and other parameters. Then, we presented different scenarios and the traffic data that we used for simulations. In Chapter 7, we introduced the implemented testbed. We have show, technical and environment settings, and the experimental scenarios. In Chapter 8 and Chapter 9, we discussed the results of the simulations and experiments, respectively. In Chapter 10, we presented some concluding remarks and future work.

10.1.1 Conclusions from Simulation Results

We summarize the conclusions for 5 simulation scenarios in following.

10.1.1.1 Simulation Scenario 1

- Using Friedman test we found out that GA, TS, HC and SA have difference in their performance.
- For Uniform distribution, the WMN-HC and WMN-SA perform better than WMN-GA and WMN-TS. However, for radius of communication distance 2×2:8×8, the SGC of WMN-TS is better than other systems.

- For Normal distribution, for radius of communication distance 2×2:8×8, the WMN-GA has the best performance.
- For Exponential distribution, the WMN-HC and WMN-SA perform better than WMN-GA for all radius of communication distances.
- For Weibull distribution, the WMN-TS has a good performance for radius of communication distance less than 2×2:6×6, but for 2×2:8×8, the WMN-GA, WMN-HC and WMN-SA perform better.

10.1.1.2 Simulation Scenario 2

- For Normal distribution, we found out that to cover all mesh clients 35 mesh routers are needed. For Uniform distribution, because the mesh clients are scattered in the grid area it was very difficult to cover all clients, so more mesh routers are needed.
- For Normal distribution, when the number of connections is 10, there is an improvement of throughput when the number of mesh routers increases. In the case of 20 connections, the network load is high and the throughput is almost the same for different number of mesh routers.
- In the case of Normal distribution, all mesh routers are concentrated in the center of grid and the communication becomes easy. On the other hand, for Uniform distribution the mesh routers are more scattered, the creating of links is more difficult and the communication can be done only with multiple hops.
- For Uniform distribution, for big number of mesh routers the total data rate for 20 connections is very high (24 [Mbps]), many packets are dropped because the congestions and the throughput is decreased.
- For Exponential distribution, when there are 20 mesh routers in the network, the performance of PDR is higher. When the number of mesh routers is increased the number of hops increases and PDR decreases.
- For Weibull distribution, when the number of mesh routers is increased the number of hops increases and PDR decreases. When there are 35 mesh routers in the network with 10 connections, the PDR is higher compared with other cases.
- For Exponential distribution, the throughput is higher for small number of mesh routers and decreases with the increase of number of mesh routers. This happens because the number of hops is increased.
- For Weibull distribution, when there are 20 connections, with the increase of the number of mesh routers the throughput is decreased much more than the case of Exponential distribution.
- For Normal and Uniform distributions, when the number of nodes in the network is increased, the delay is increased.
- For Exponential and Weibull distributions, the delay is almost the same for both distributions.

10.1.1.3 Simulation scenario 3

- For 10, 20 and 30 number of connections, the PDR is less than 60 [%] when the transmission rate is more than 1200 [kbps] for Exponential distribution. The PDR for Weibull distribution is higher than Exponential distribution.
- For different number of connections, the throughput is increased linearly with the increasing of the transmission rate. The throughput of Exponential distribution is higher than Weibull distribution.
- For 10 connections, the delay is very small until the transmission rate is 800 [kbps] for Exponential distribution. The delay of Exponential distribution is smaller than Weibull distribution.

10.1.1.4 Simulation scenario 4

- The throughput of I/B WMN is higher than Hybrid WMN architecture.
- The delay and jitter of Hybrid WMN is a lower compared with I/B WMN.
- The fairness index of 10 and 20 flows is higher than 30 flows for both architectures. In I/B architecture the fairness index of DCF is higher than EDCA. However, for Hybrid WMN, the fairness index of EDCA is higher than DCF.

10.1.1.5 Simulation scenario 5

- For Normal and Exponential distributions, the throughput of I/B WMN is a little bit higher than Hybrid architecture. For Weibull distribution the throughput is almost the same for both WMN architectures. However, for Uniform distribution the throughput of Hybrid WMN is higher than I/B WMN. This is because for Normal and Exponential distributions, the mesh routers are concentrated in the grid area, thus there are many collisions and the network becomes congested.
- Considering HWMP, for Normal and Weibull distributions, the delay is almost the same. However, for Uniform distribution the delay of Hybrid WMN is lower than I/B WMN. This is because in Hybrid WMN also the mesh client communicate between each other. But, for Exponential distribution the delay of I/B WMN is low compared with Hybrid WMN.
- For HWMP, in case of Normal and Exponential distributions, the energy decreases sharply, because the mesh routers are concentrated in the grid area and many packets collide with each other. For Weibull distribution, the energy decrease almost the same for both WMN architectures. For Uniform distribution, the remaining energy of I/B WMN is higher than Hybrid WMN. This is because in Hybrid WMN, there are three communications: mesh client to mesh client, mesh router to mesh router and mesh client to mesh router.
- For HWMP the throughput of Uniform distribution is higher than other distributions. But, the delay and remaining energy is better for Weibull distribution.
- For OLSR protocol and Normal and Uniform distributions, the throughput of Hybrid WMN is higher than I/B WMN architecture. But, for Exponential distribution, the throughput of

I/B WMN is higher than Hybrid WMN architecture. For Weibull distribution, the throughput is almost the same for both WMN architectures.

- Considering OLSR protocol, for Normal and Uniform distributions the delay of Hybrid WMN is a lower compared with I/B WMN. For Exponential and Weibull distributions, the delay is almost the same for both distributions. However, the delay of Weibull distribution is lower than other distributions.
- The remaining energy of OLSR protocol for Normal distribution, the remaining energy of Hybrid WMN is higher than I/B WMN. While for Exponential distribution, the remaining energy of I/B WMN is higher than Hybrid WMN. For Uniform and Weibull distributions, the energy decrease is almost the same for both WMN architectures. Also, for Uniform and Weibull distributions, the remaining energy is higher compared with Normal and Exponential distributions.
- For OLSR protocol, the throughput of Exponential distribution is better than other distributions. But, the delay and remaining energy is better for Weibull distribution.

10.1.2 Conclusions from Experimental Results

We summarize the conclusions from 4 experimental scenarios in following.

10.1.2.1 Experimental Scenario 1

In this scenario, we compared the performance for Raspbian and OpenWRT OSs.

- The average throughput of Raspbian and OpenWRT is 449 and 494 [Kbps].
- The average PDR of Raspbian and OpenWRT is 88 and 96 [%].
- The communication of Raspbian and OpenWRT is done for almost 1 hop.
- The average delay of Raspbian and OpenWRT is lower than 0.08 and 0.009 [sec], respectively.
- The average jitter of Raspbian and OpenWRT is lower than 0.08 and 0.011 [sec], respectively.
- The experimental results show that the testbed mounted in OpenWRT has better results than Raspbian.

10.1.2.2 Experimental Scenario 2

In this scenario, we compared the performance for LoS and NLoS environments.

- The average throughput of LoS and NLoS is 494 and 473 [Kbps].
- The PDR of LoS and NLoS is 96 and 92 [%].
- The communication of LoS and NLoS is done for 1 and 3 hops.
- The delay of LoS and NLoS is lower than 0.009 and 0.05 [sec], respectively.
- The jitter of LoS and NLoS is lower than 0.011 and 0.06 [sec], respectively.
- The experimental results show that the testbed mounted in LoS environment has better results than NLoS environment.

10.1.2.3 Experimental Scenario 3

For this scenario, we considered LoS environment and single flow for the case of a mobile node.

- The average throughput is 381 [Kbps].
- The average PDR is 77 [%].
- The communication is done for about 1 hop.
- The delay and jitter is lower than 0.14 [sec].

10.1.2.4 Experimental Scenario 4

For this scenarion, we evaluated the performance of the WMN testbed for distributed concurrent processing and LoS environment.

- The communication is done for almost 1 hop.
- The delay and jitter is lower than 0.00506 and 0.00378 [sec], respectively.
- The processing time is lowest for node 1.

10.2 Future Work

In the future work, we will deal with the following issues.

Simulated Mobility: During simulation, the mobility model used for node movement has a key influence on the performance of different protocols and algorithms. We would like to create more realistic mobility models, similar to real life cities, campuses or buildings.

Testbed Improvement: This work inspires us to improve our testbed with more nodes and spread our experiments in outdoor environment and all over the campus. We would also like to create a monitoring interface, in order to have access to real-time results and accurate node positions. Moreover, an accurate moving mechanism is important to create the same topology in different experimental settings.

Applications of WMN: WMN will be the next technology for building cheap and autonomous backbones of modern networks.

References

[1] D. A. Maltz, J. Broch, and D. B. Johnson, "Lessons from a Full-scale Multihop Wireless Ad Hoc Network Testbed", IEEE Personal Communications, Vol. 8, No. 1, pp. 8-15, February 2001.

[2] R. S. Gray, D. Kotz, C. Newport, N. Dubrovsky, A. Fiske, J. Liu, C. Masone, S. McGrath, and Y. Yuan, "Outdoor Experimental Comparison of Four Ad Hoc Routing Algorithms", in Proc. MSWiM-2004, pp. 220-229, 2004.

[3] G. Anastasi, E. Borgia, M. Conti, and E. Gregori, "IEEE 802.11b Ad Hoc Networks: Performance Measurements", Cluster Computing, Vol. 8, No. 2-3, pp. 135-145, 2005.

[4] D. Johnson and G. Hancke, "Comparison of Two Routing Metrics in OLSR on a Grid Based Mesh Network", Ad Hoc Networks, Vol. 7, pp. 374-387, March 2009.

[5] G. De Marco, M. Ikeda, T. Yang, and L. Barolli, "Experimental Performance Evaluation of a Pro-active Ad-hoc Routing Protocol in Out and Indoor Scenarios", in Proc. AINA-2007, pp. 7-14, Canada, May 2007.

[6] V. Kawadia and P. R. Kumar, "Experimental Investigations into TCP Performance over Wireless Multihop Networks", in Proc. of E-WIND-2005, pp. 29-34, 2005.

[7] T. Clausen, G. Hansen, L. Christensen, and G. Behrmann, "The Optimized Link State Routing Protocol, Evaluation Through Experiments and Simulation", in Proc. of IEEE Symposium on Wireless Personal Mobile Communications, September 2001, `http://hipercom.inria.fr/olsr/wpmc01.ps`.

[8] X. H. Wang and X. Zhou, "Design and Implementation of a Dual-Radio Wireless Mesh Network Testbed for Healthcare", International Conference on Information Technology and Applications in Biomedicine (ITAB-2008), pp. 300-304, May 2008.

[9] N. A. Ali, E. Ekram, A. Eljasmy and K. Shuaib, "Measured delay distribution in a Wireless Mesh Network Test-bed", IEEE/ACS International Conference on Computer Systems and Applications (AICCSA-2008), pp. 236-240, March 2008.

[10] D. Bansal, S. Sofat, "Deployment and Evaluation of IEEE 802.11 based Wireless Mesh Networks in Campus Environment", Proceedings of the 4th ACM Workshop on Networked Systems for Developing Regions (NSDR-2010), 2010.

[11] M. Kaur, Kunwarpal, "Optimize OLSR with Cognitive in Wireless Mesh Network", International Journal of Engineering and Advanced Technology (IJEAT), Vol. 2, No. 3, 2013.

[12] T. Oda, A. Barolli, F. Xhafa, L. Barolli, M. Ikeda, M. Takizawa, "WMN-GA: A Simulation System for WMNs and Its Evaluation Considering Selection Operators", In Press, Journal of Ambient Intelligence and Humanized Computing (JAIHC), Springer, Vol. 4, No. 3, pp. 323-330, June 2013.

[13] M. Ikeda, T. Oda, E. Kulla, M. Hiyama, L. Barolli and M. Younas, "Performance Evaluation of WMN Considering Number of Connections Using NS-3 Simulator", The Third International Workshop on Methods, Analysis and Protocols for Wireless Communication (MAPWC 2012), pp. 498-502, Victoria, Canada, November 12-14, 2012.

[14] T. Oda, S. Sakamoto, A. Barolli, M. Ikeda, L. Barolli, F. Xhafa, "A GA-Based Simulation System for WMNs: Performance Analysis for Different WMN Architectures Considering TCP", 2014 Eighth International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2014), Guangzhou, China, November 8-10, 2014.

[15] E. Nordstrom, "APE - a large scale ad hoc network testbed for reproducible performance tests", Master's thesis, Uppsala University, 2002.

[16] R. Draves, J. Padhye, and B. Zill, "Comparison of routing metrics for static multi-hop wireless networks," in SIGCOMM-04, pp. 133-144, 2004.

[17] S. N. Muthaiah and C. Rosenberg, "Single Gateway Placement in Wireless Mesh Networks", In Proc. of 8th

International IEEE Symposium on Computer Networks, Turkey, pp. 4754-4759, 2008.

[18] M. Tang, "Gateways placement in backbone wireless mesh networks", International Journal of Communications, Network and System Sciences, Vol. 2, No. 1, pp. 44-50, 2009.

[19] A. Franklin and C. Murthy, "Node placement algorithm for deployment of two-tier wireless mesh networks", IEEE GLOBECOM-2007, pp. 4823-4827, 2007.

[20] T. Vanhatupa, M. Hannikainen, and T. Hamalainen, "Genetic algorithm to optimize node placement and configuration for wlan planning", 4th International Symposium on Wireless Communication Systems, pp.612-616, 2007.

[21] A. Lim, B. Rodrigues, F. Wang, and Z. Xu, "k-Center problems with minimum coverage", Theoretical Computer Science, Vol. 332, pp. 1-17, 2005.

[22] J. Wang, B. Xie, K. Cai, and D. Agrawal, "Efficient mesh router placement in wireless mesh networks", MASS-2007, pp.9-11, 2007.

[23] X. Yao, "An empirical study of genetic operators in genetic algorithms", in EUROMICRO 93 Nineteenth EUROMICRO Symposium on Microprocessing and Microprogramming on Open System Design: Hardware, Software and Applications, pp. 707-714, 1993.

[24] J. Denzinger and J. Kidney, "Evaluating different genetic operators in the testing for unwanted emergent behavior using evolutionary learning of behavior", in IEEE/WIC/ACM International Conference on Intelligent Agent Technology, pp. 23-29, 2006.

[25] M. Odetayo, "Empirical study of the interdependencies of genetic algorithm parameters", in 23rd EUROMICRO Conference, New Frontiers of Information Technology, pp. 639-643, 1997.

[26] F. Xhafa, B. Duran, A. Abrahamy, and K. Daha, "Tuning struggle strategy in genetic algorithms for scheduling in computational grids", Neural Network World, Vol. 18, No. 3, pp. 209-225, 2008.

[27] F. Xhafa, L. Barolli, and A. Durresi, "An experimental study on genetic algorithms for resource allocation on grid systems", Journal of Interconnection Networks, Vol. 8, No. 4, pp. 427-443, 2007.

[28] F. Xhafa, C. Sanche, and L. Barolli, "Ad hoc and neighborhood search methods for placement of mesh routers in wireless mesh networks", in ICDCS Workshops of the IEEE 29th International Conference on Distributed Computing Systems (ICDCS-09), pp. 400-405, 2009.

[29] P. Santi, "Topology Control in Wireless Ad Hoc and Sensor Networks", Wiley, 2005.

[30] IEEE-SA, "IEEE 802.11 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", 1999.

[31] IEEE-SA, "IEEE 802.11e Amendment: Medium Access Control (MAC) Quality of Service (QoS) Enhancements", 2005.

[32] S. Mukherjee, P. Xiao-Hong, Q. Gao, "QoS Performances of IEEE 802.11 EDCA and DCF: A Testbed Approach", 5th International Conference Wireless Communications, Networking and Mobile Computing (WiCom-09), pp. 1-5, 2009.

[33] M. Abolhasan, T. Wysocki, E. Dutkiewicz, "A Review of Routing Protocols for Mobile Ad hoc Networks", Ad Hoc Networks, Vol. 2, No. 1, pp. 1-22, 2004.

[34] C. Perkins, E. Belding-Royer and S. Das, "Ad hoc On-demand Distance Vector (AODV) Routing", IETF RFC 3561, July 2003.

[35] L. Barolli, M. Ikeda, G.D. Marco, A. Durresi, F. Xhafa, "Performance Analysis of OLSR and BATMAN Protocols Considering Link Quality Parameter", Proc. of International Conference on Advanced Information Networking and Application, IEEE Computer Society, pp. 307-314, 2009.

[36] T. Clausen, P. Jacquet, "Optimized Link State Routing Protocol", Technical report, Project Hipercom, INRIA Rocquencourt, France, 2003.

[37] W. S. Conner, "IEEE 802.11 TGs Usage Models", IEEE802.11 document 04/662r16, Jan. 2005.

[38] W. S. Conner, "IEEE 802.11 TGs Functional Requirements and Scope", IEEE802.11 document 04/1175r10, Jan. 2005.

[39] One Laptop Per Child project. `http://www.laptop.org/en/`. Nokia Research Center, University of California, University of Cincinnati, 2003.

[40] S. K. Gaur, S. K. Tyagi, P. Singh, " "VANET" System for Vehicular Security Applications", International Journal of Soft Computing and Engineering (IJSCE), Vol. 2, No. 6, 2013.

[41] A. Tangavelu, "Location Identification and Vehicle Tracking using VANET (VETRAC)", IEEE ICSN-2007,

pp. 112-116, Feb. 22-24 2007.

[42] I. F. Akyildiz and I. H. Kasimoglu, “Wireless sensor and actor networks: Research challenges”, Ad Hoc Networks Journal, Elsevier, Vol. 2, No. 4, pp. 351-367, October 2004.

[43] M. Akbas and D. Turgut, “Apawsan: Actor positioning for aerial wireless sensor and actor networks”, IEEE 36th Conference on Local Computer Networks (LCN), pp. 563-570, Oct 2011.

[44] M. Akbas, M. Brust, and D. Turgut, “Local positioning for environmental monitoring in wireless sensor and actor networks”, IEEE 35th Conference on Local Computer Networks (LCN), pp. 806-813, Oct 2010.

[45] T. Melodia, D. Pompili, V. Gungor, and I. AkyildizZX, “Communication and coordination in wireless sensor and actor networks”, IEEE Transactions on Mobile Computing, Vol. 6, No. 10, pp. 1126-1129, October 2007.

[46] V. Gungor, O. Akan, and I. Akyildiz, “A real-time and reliable transport (rt2) protocol for wireless sensor and actor networks”, IEEE/ACM Transactions on Networking, Vol. 16, No. 2, pp. 359-370, April 2008.

[47] K. Selvaradjou, N. Handigol, A. Franklin, and C. Murthy, “Energy-efficient directional routing between partitioned actors in wireless sensor and actor networks”, Communications, IET, Vol. 4, No. 1, pp. 102-115, January 2010.

[48] A. Abbasi, M. Younis, and K. Akkaya, “Movement-assisted connectivity restoration in wireless sensor and actor networks”, IEEE Transactions on Parallel and Distributed Systems, Vol. 20, No. 9, pp. 1366-1379, Sept 2009.

[49] H. Nakayama, Z. Fadlullah, N. Ansari, and N. Kato, “A novel scheme for wsan sink mobility based on clustering and set packing techniques”, IEEE Transactions on Automatic Control, Vol. 56, No. 10, pp. 2381-2389, Oct 2011.

[50] A. Zanella, N. Bui, A. Castellani, L. Vangelista, “Internet of Things for Smart Cities”, IEEE Internet of Things Journal, Vol. 1, No. 1, pp. 22-32, 2014.

[51] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey”, Comput. Netw., Vol. 54, No. 15, pp. 2787-2805, 2010.

[52] P. Bellavista, G. Cardone, A. Corradi, and L. Foschini, “Convergence of MANET and WSN in IoT urban scenarios”, IEEE Sens. J., Vol. 13, No. 10, pp. 3558-3567, Oct. 2013.

[53] G. Acampora, D. Cook, P. Rashidi and A. V. Vasilakos, “A Survey on Ambient Intelligence in Health Care”, Proc. of the IEEE, Vol.101, No.12, pp. 2470-2494, 2013.

[54] E. Aarts and R. Wichert, “Ambient intelligence”, Technology Guide, pp. 244-249, 2009.

[55] E. Aarts and B. de Ruyter, “New research perspectives on ambient intelligence”, Journal of Ambient Intelligence and Smart Environments, Vol. 1, No. 1, pp. 5-14, 2009.

[56] A. Vasilakos and W. Pedrycz, “Ambient Intelligence, Wireless Networking, And Ubiquitous Computing. Norwood”, MA, USA: Artech House, Inc., 2006.

[57] F. Sadri, “Ambient intelligence: A survey”, ACM Comput. Surv., Vol. 43, No. 4, pp. 36:1-36:66, Oct. 2011.

[58] D. Cook, J. Augusto, and V. Jakkula, “Ambient intelligence: Technologies, applications, and opportunities”, Pervasive and Mobile Computing, Vol. 5, No. 4, pp. 277-298, 2009

[59] I. F. Akyldiz, X. Wang, “A Survey on Wireless Mesh Networks”, IEEE Communications Magazine, Vol. 43, No. 9, pp. S23-S30, 2005.

[60] F. Xhafa, C. Sanchez, and L. Barolli, “Locals search algorithms for efficient router nodes placement in wireless mesh networks”, in International Conference on Network-Based Information Systems (NBiS-2009), 2009, pp. 572-579.

[61] H. Lundgren, E. Nordstrom, C. Tschudin, “Coping with Communication Gray Zones in IEEE 802.11b based Ad hoc Networks”, Proc. of WoWMoM-02, pp. 49-55, 2002.

[62] S. Sanghani, T.X. Brown, S. Bhandare, S. Doshi, “EWANT: The Emulated Wireless Adhoc Testbed”, Proc. of IEEE WCNC-2003, pp. 1884-1849, March 2003.

[63] L. Rizzo, “Dummynet: A Simple Approach to the Evaluation of Network Protocols”, Journal of Computer Communication Review, Vol. 27, No. 1, pp. 31-41, 1997.

[64] S. Olafsson, “Metaheuristics”, in Nelson and Henderson (eds.). Handbook on Simulation, Handbooks in Operations Research and Management Science Elsevier, Vol. 7, pp. 633-654, 2006.

[65] J. April, F. Glover, J. P. Kelly, M. Laguna, “Practical introduction to simulation optimization”. Proceedings of the 2003 Winter Simulation Conference, pp. 71-78, 2003.

[66] M. C. Fu, “Optimization for simulation: Theory and practice” INFORMS J. Computing, Vol. 14, No. 3, pp. 192-215, 2002.

[67] S. Olafsson, J. Kim, “Simulation optimization”, Proc. of the 2002 Winter Simulation Conference. pp. 79-84,

2002.
[68] J.P. Kelly, 2002. "Simulation optimization is evolving", INFORMS Journal on Computing, Vol. 14, No. 3, pp. 223-225, 2002.
[69] F. Glover, M. Laguna and R. Marti. "Scatter search", To appear in Ghosh and Tsutsui (Eds.) Theory and Applications of Evolutionary Computation: Recent Trends, Springer, 2003.
[70] S. Kirkpatrick, C. D. Gelatt Jr., M. P. Vecchi, "Optimization by simulated annealing", Science, Vol. 220, pp. 671-680, 1983.
[71] R.W. Eglese, "Simulated annealing: a tool for operation research", European Journal of Operational Research, Vol. 46, pp. 271-281, 1990.
[72] M.A. Fleischer, "Simulated annealing: past present and future", Proc. of the 1995 Winter Simulation Conference, pp. 155-161, 1995.
[73] F. Glover, M. Laguna, "Tabu Search". Kluwer Academic, Boston, 1997.
[74] F. Glover, "Tabu Search -Part I". ORSA Journal on Computing, Vol. 1, pp. 190-206, 1989.
[75] F. Glover, "Tabu Search -Part II". ORSA Journal on Computing, Vol. 2, pp. 4-32, 1990.
[76] M.G.C. Resende, C.C. Ribeiro, "Greedy randomized adaptive search procedures", In Glover and Kochenberger (eds), Handbook of Metaheuristics. Kluwer, Boston, MA, 2003.
[77] P. Hansen, N. Mladenovic, "An introduction to variable neighborhood search", In S. Voss et al. (eds), Proc. of MIC-97 Conference, pp. 433-458, 1997.
[78] H. Muhlenbein, "Genetic algorithms. In Local Search in Combinatorial Optimization", Eds. E. Aarts and J. K. Lenstra, pp. 137-172, 1997.
[79] D.E. Goldberg, "Genetic Algorithm in Search, Optimization and Machine Learning", Addison-Wesley, Reading, MA, 1989.
[80] G. E. Liepins, M. R. Hilliard, "Genetic algorithm: foundations and applications", Annals of the Operations Research, Vol. 21, pp. 31-58, 1989.
[81] F. Glover, G.A. Kochenberger, "Handbook of Metaheuristics", Kluwer Academic Publishers, Boston, MA, 2003.
[82] L. Shi, S. Olafsson, "Nested partitions method for global optimization", Operations Research, Vol. 48, pp. 390-407, 2000.
[83] S. H. Jacobson, K. A. Sullivan, A. W. Johnson, "Discrete manufacturing process design optimization using computer simulation and generalized hill-climbing algorithm", Engineering Optimization, Vol. 31, pp. 247-260, 1998.
[84] A. W. Johnson, S. H. Jacobson, "On the convergence of generalized hill climbing algorithms", Discrete Applied Mathematics, Vol. 119, pp.37-57, 2002.
[85] K. A. Sullivan, S. H. Jacobson, "A convergence analysis of generalized hill climbing algorithms", IEEE Transaction on Automatic Control, Vol. 46, No. 8, pp. 1988-1293, 2001.
[86] J. H. Holland, "Adaptation in Natural and Artificial Systems", University of Michigan Press, 1975.
[87] D. Dasgupta, D. R. Mcgregor, "Nonstationary function optimization using the structured genetic algorithm", Parallel Problem Solving from Nature, Vol. 2, pp. 145-154, 1992.
[88] J. J. Grefenstette, "Genetic algorithm for changing environments", Parellel Problem Solving from Nature, Vol. 2, Elsevier Science Publishers B.V. Netherlands, 1992.
[89] H. Ishibuchi, T. Murata, 1996. "Multi-objective genetic local search algorithm", Proc. of IEEE International Conference on Evolutionary Computation, 119-124, 1996.
[90] F. Vavak, T. C. Fogarty, "Comparison of steady state and generational genetic algorithms for use in nonstationary environments", Proc. of IEEE International Conference on Evolutionary Computation, pp. 192-195, 1996.
[91] Y. Yoshitomi, R. Yamaguchi, "A genetic algorithm and the Monte Carlo method for stochastic job-shop scheduling", International Transaction of Operational Research, Vol. 10, pp. 577-596, 2003.
[92] J. Boesel, B. L. Nelson, N. Ishii, "A framework for simulation optimization software", IIE Transactions, Vol. 35, No. 3, pp. 221-229, 2003.
[93] T. Back, "Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms", Oxford University Press, New York, 1996.
[94] M. Mitchell, "Genetic Algorithms: An Overview", Complexity, Vol. 1, No. 1, pp. 31-39, 1995.

[95] J. H. Holland, "Adaptation In Natural and Artificial Systems", University of Michigan Press, 1975.
[96] P. Smolensky, "On the proper treatment of connectionism", Behavioral and Brain Sciences, Vol. 11, No. 1, pp. 1-14, 1988.
[97] R. L. Riolo, "Survival of the fittest bits", Scientific American, Vol. 267, No. 1, pp. 114-116, 1992.
[98] I. F. Akyildiz, Xudong Wang, Weilin Wang, "Wireless Mesh Networks: A Survey", Computer Networks, Vol.47, pp. 445-487, 2005.
[99] Y. Zhang, J. Luo, and H. Hu, "Wireless mesh networks: architecture, protocols and standards", Auerbach Publ., 2007.
[100] E. Amaldi, A. Capone, M. Cesana, I. Filippini, F. Malucelli, "Optimization models and methods for planning wireless mesh networks", Computer Networks, Vol. 52, pp. 2159-2171, 2008.
[101] Ch. Chen and Ch. Chekuri, "Urban Wireless Mesh Network Planning: The Case of Directional Antennas", Tech Report No. UIUCDCS-R-2007-2874, Department of Computer Science, University of Illinois at Urbana-Champaign, 2007.
[102] M. R. Garey and D. S. Johnson, "Computers and intractability: A guide to the theory of NP-completeness", Freeman, New York, 1979.
[103] J. Wong, R. Jafari, and M. Potkonjak, "Gateway placement for latency and energy efficient data aggregation", in Proceedings IEEE LCN, pp. 490-497, 2004.
[104] S.N. Muthaiah and C.P. Rosenberg, "Single gateway placement in wireless mesh networks", 8th International IEEE Symposium on Computer Networks, pp. 4754-4759, 2008.
[105] F. Li, Y. Wang, X.-Y. Li, A. Nusairat and Y. Wu. Gateway Placement for Throughput Optimization in Wireless Mesh Networks", Mobile Networks and Applications Vol. 13, No. 1-2, pp. 198-211, 2008.
[106] P. Zhou, B. S. Manoj, Ramesh Rao, "A gateway placement algorithm in wireless mesh networks", WICON-07 Proceedings of the 3rd international conference on Wireless internet Article, No. 1, 2007.
[107] K. Tahera, R. N. Ibrahim, P. B. Lochert, "Adopting dynamic operators in a genetic algorithm", Proceedings of the 9th annual conference on Genetic and evolutionary computation, July 07-11, 2007.
[108] F. Glover, "Future Paths for Integer Programming and Links to Artificial Intelligence", Computers and Op. Res., pp. 533-549, 1986.
[109] S. Kirkpatrik, C. D. Gelatt and M. P. Vecchi, "Optimization by Simulated Annealing", Journal of Science, Vol. 220, pp. 671-680, 1983.
[110] F. Xhafa, C. Sanchez, L. Barolli, R. Miho, "An Annealing Approach to Router Nodes Placement Problem in Wireless Mesh Networks", In Proc. of CISIS-2010, pp.245-252, 2010.
[111] F. Xhafa, A. Bravo, A. Barolli, M. Takizawa, "An Interface for Simulating Node Placement in Wireless Mesh Networks", In Proc. of 15th International Conference on Network-Based Information Systems (NBiS-2012), pp. 326-333, 2012.
[112] "ns-3 website" ,https://www.nsnam.org/.
[113] IEEE 802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", IEEE Computer Society Std., June 2007. [Online]. Available:
`http://standards.ieee.org/getieee802/download/802.11-2007.pdf`.
[114] M. Friedman, "The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance", Journal of the American Statistical Association, Vol. 32, No. 200, pp. 675-701, 1937.
[115] "Raspberry Pi Foundation.", `http://www.raspberrypi.org/`.
[116] "Raspbian: FrontPage.", `https://www.raspbian.org/`.
[117] "OpenWrt.", `https://openwrt.org/`.
[118] rt2800usb - Debian Wiki, Available Online at: `http://wiki.debian.org/rt2800usb`.
[119] Rt2x00 Project for Linux Wireless Cards Drivers, Available Online at: `http://rt2x00.serialmonkey.com/`.
[120] Ralink RT2570 USB Enhanced Driver, Available Online at:
`http://homepages.tu-darmstadt.de/~p_larbig/wlan/`.
[121] "Iperf - The TCP/UDP Bandwidth Measurement Tool.", `https://iperf.fr/`.
[122] "LuCI.", `http://luci.subsignal.org/`.
[123] J. Hursey, J. M. Squyres, T. I. Mattox, A. Lumsdaine, "The Design and Implementation of Checkpoint/Restart Process Fault Tolerance for Open MPI", IEEE International Parallel and Distributed Processing Symposium (IPDPS-2007), pp. 1-8, 2007.

[124] "NAS Parallel Benchmarks", `http://www.nas.nasa.gov/publications/npb.html`.

[125] F. C. Wong, R. P. Martin, R. H. Arpaci-Dusseau, and D. E. Culler, "Architectural Requirements and Scalability of the NAS Parallel Benchmarks", in Proceedings of the 1999 ACM/IEEE conference on Supercomputing (CDROM), ser. Supercomputing ' 99. New York, NY, USA: ACM, 1999.

[126] D. H. Bailey, E. Barszcz, J. T. Barton, D. S. Browning, R. L. Carter, D. Dagum, R. A. Fatoohi, P. O. Frederickson, T. A. Lasinski, R. S. Schreiber, H. D. Simon, V. Venkatakrishnan, and S. K. Weeratunga, "The NAS Parallel Benchmarks", The Intl. Journal of Supercomputer Applications, vol. 5, no. 3, pp. 63-73, Fall 1991.

[127] D. H. Bailey, E. Barszcz, J. T. Barton, D. S. Browning, R. L. Carter, L. Dagum, R. A. Fatoohi, P. O. Frederickson, T. A. Lasinski, R. S. Schreiber, H. D. Simon, V. Venkatakrishnan, and S. K. Weeratunga, "The NAS Parallel Benchmarks: Summary and Preliminary Results", in Proceedings of the 1991 ACM/IEEE conference on Supercomputing, ser. Supercomputing ' 91. New York, NY, USA: ACM, pp. 158-165, 1991.

[128] A. Ramachandran, J. Vienne, R. Van Der Wijngaart, L. Koesterke, I. Sharapov, "Performance Evaluation of NAS Parallel Benchmarks on Intel Xeon Phi", 42nd International Conference on Parallel Processing (ICPP), pp. 736-743, 2013.

List of Papers

Journal Papers (reviewed)

1. A. Barolli, V. Loia, T. Oda, L. Barolli, F. Xhafa, M. Takizawa, “Interface and Results Visualization of WMN-GA Simulation System: Evaluation for Exponential and Weibull Distributions Considering Different Transmission Rates”, Computer Standards & Interfaces (CSI), Elsevier, Vol. 44, pp. 150-158, 2016. (Published online: 8 May, 2015). (DOI:10.1016/j.csi.2015.04.003)
2. T. Inaba, D. Elmazi, S. Sakamoto, T. Oda, M. Ikeda, L. Barolli, “A Secure-Aware Call Admission Control Scheme for Wireless Cellular Networks Using Fuzzy Logic and Its Performance Evaluation”, Journal of Mobile Multimedia (JMM), Rinton Press, Vol. 11, No. 3&4, pp. 213-222, 2015.
3. T. Ishitaki, T. Oda, Y. Liu, D. Elmazi, K. Matsuo, L. Barolli, “A Neural Network Based Intrusion Detection and User Identification System for Tor Networks: Performance Evaluation for Different Number of Hidden Units Using Friedman Test”, Journal of Mobile Multimedia (JMM), Rinton Press, Vol. 11, No. 3&4, pp. 251-262, 2015.
4. D. Elmazi, E. Kulla, K. Matsuo, T. Oda, E. Spaho, L. Barolli, “A Mobility-Aware Fuzzy-based System for Actor Selection in Wireless Sensor-Actor Networks”, Journal of High Speed Networks, IOS Press, Vol. 21, No. 1, pp. 15-25, 2015. (DOI: 10.3233/JHS-150505)
5. D. Elmazi, E. Kulla, T. Oda, E. Spaho, S. Sakamoto, L. Barolli, “A Comparison Study of Two Fuzzy-based Systems for Selection of Actor Node in Wireless Sensor Actor Networks”, Journal of Ambient Intelligence and Humanized Computing (JAIHC), Springer, Vol. 6, No. 5, pp. 635-645, 2015.
6. A. Lala, V. Kolici, T. Oda, L. Barolli, A. Barolli, F. Xhafa, “A Web Interface for Wireless Mesh Networks Based on Heuristic Algorithms: Optimization and Analysis for Different Scenarios”, International Journal of Web and Grid Services (IJWGS), Inderscience, Vol. 11, No. 3, pp. 327-346, 2015. (DOI: 10.1504/IJWGS.2015.070986)
7. S. Sakamoto, A. Lala, T. Oda, V. Kolici, L. Barolli, F. Xhafa, “Application of WMN-SA Simulation System for Node Placement in Wireless Mesh Networks: A Case Study for a Realistic Scenario”, International Journal of Mobile Computing and Multimedia Communications (IJMCMC), Vol. 6, No. 2, pp. 13-21, April-June 2014. (DOI: 10.4018/ijmcmc.2014040102)
8. T. Oda, Y. Liu, S. Sakamoto, D. Elmazi, L. Barolli, F. Xhafa, “Analysis of Mesh Router Placement in Wireless Mesh Networks Using Friedman Test Considering Different Meta-heuristics”, International Journal of Communication Networks and Distributed Systems (IJCNDS), Inderscience, Vol. 15, No. 1, pp. 84-106, 2015. (DOI: 10.1504/IJCNDS.2015.0702895)
9. T. Oda, D. Elmazi, A. Barolli, S. Sakamoto, L. Barolli, F. Xhafa, “A Genetic Algorithm Based System for Wireless Mesh Networks: Analysis of System Data Considering Different Routing Protocols and Architectures”, Journal of Soft Computing (SOCO), Springer, pp. 1-14, 2015. (Published online: 31 March, 2015). (DOI: 10.1007/s00500-015-1663-z)
10. D. Elmazi, K. Matsuo, T. Oda, M. Ikeda and L. Barolli, “F3N: An Intelligent Fuzzy-based Cluster Head Selection System for WSNs and Its Performance Evaluation”, International Journal of Distributed Systems and Technologies (IJDST), IGI Global, Vol. 6, No. 2, 2015. (DOI: 10.4018/ijdst.2015040103)
11. D. Elmazi, E. Spaho, K. Matsuo, T. Oda, M. Ikeda and L. Barolli, “Implementation and Evaluation of a Fuzzy-Based Cluster-Head Selection System for Wireless Sensor Networks Considering Network Traffic”, Journal of Mobile Multimedia (JMM), Rinton Press, Vol. 11, No. 1&2, pp. 10-20, 2015.
12. A. Barolli, T. Oda, M. Ikeda, L. Barolli, F. Xhafa, V. Loia, “Node Placement for Wireless Mesh Networks: Analysis of WMN-GA System Simulation Results for Different Parameters and Distributions”, Journal of Computer and System Sciences (JCSS), Elsevier, Vol. 81, No. 8, pp. 1496-1507, December 2015. (DOI:10.1016/j.jcss.2014.12.024)

13. S. Sakamoto, E. Kulla, T. Oda, M. Ikeda, L. Barolli and F. Xhafa, "A Simulation System for WMN Based on SA: Performance Evaluation for Different Instances and Starting Temperature Values", International Journal of Space-Based and Situated Computing (IJSSC), Inderscience, Vol. 4, No. 3/4, pp. 209-216, 2014. (DOI: 10.1504/IJSSC.2014.065989)
14. T. Oda, S. Sakamoto, A. Barolli, E. Spaho, L. Barolli and F. Xhafa, "Effect of Different Grid Shapes in Wireless Mesh Network Genetic Algorithm System", International Journal of Web and Grid Services (IJWGS), Inderscience, Vol. 10, No. 4, pp. 371-395, 2014. (DOI: 10.1504/IJWGS.2014.064932)
15. A. Barolli, S. Sakamoto, T. Oda, E. Spaho, L. Barolli and F. Xhafa, "Performance Evaluation of WMN-GA System for Different Settings of Population Size and Number of Generations", Journal of Human-centric Computing and Information Sciences (HCIS), Vol. 4, No. 5, pp. 1-14, 2014. (DOI: 10.1186/s13673-014-0005-y)
16. S. Sakamoto, E. Kulla, T. Oda, M. Ikeda, L. Barolli and F. Xhafa, "A Comparison Study of Hill Climbing, Simulated Annealing and Genetic Algorithm for Node Placement Problem in WMNs", Journal of High Speed Networks, IOS Press, Vol. 20, No. 1, pp. 55-66, 2014. (DOI: 10.3233/JHS-140487)
17. S. Sakamoto, E. Kulla, T. Oda, M. Ikeda, L. Barolli and F. Xhafa, "Performance Evaluation Considering Iterations per Phase and SA Temperature in WMN-SA System", International Journal of Mobile Information Systems (MIS), IOS Press, Vol. 10, No. 3, pp. 321-330, 2014. (DOI 10.3233/MIS-130187)
18. S. Sakamoto, E. Kulla, T. Oda, M. Ikeda, L. Barolli and F. Xhafa, "A Comparison Study of Simulated Annealing and Genetic Algorithm for Node Placement Problem in Wireless Mesh Networks", Journal of Mobile Multimedia (JMM), Rinton Press, Vol. 9, No. 1& 2, pp. 101-110, November 2013.
19. E. Kulla, M. Ikeda, T. Oda, L. Barolli, F. Xhafa and A. Biberaj, "Experimental results from a MANET testbed in outdoor bridge environment considering BATMAN routing protocol", Journal of Computing, Springer, Vol. 95, No. 10-11, pp 1073-1086, October 2013. (DOI 10.1007/s00607-012-0225-9)
20. T. Oda, A. Barolli, E. Spaho, L. Barolli, F. Xhafa, M. Younas, "Effects of population size for location-aware node placement in WMNs: evaluation by a genetic algorithm based approach", International Journal of Personal and Ubiquitous Computing (PUC), Springer, Vol. 18, No. 2, pp. 261-269, February 2014. (DOI 10.1007/s00779-013-0643-5)
21. E. Kulla, T. Oda, M. Ikeda, L. Barolli, A. Biberaj, O. Shurdhi, "Source and Destination Mobility Effects on a Bridge Environment: Results from a MANET Testbed", International Journal of Wireless and Mobile Computing (IJWMC), Inderscience, Vol. 7, No. 4, pp. 318-326, 2014. (Published online: 2012). (DOI: 10.1504/IJWMC.2014.063058)
22. T. Oda, A. Barolli, F. Xhafa, L. Barolli, M. Ikeda, M. Takizawa, "WMN-GA: A Simulation System for WMNs and Its Evaluation Considering Selection Operators", Journal of Ambient Intelligence and Humanized Computing (JAIHC), Springer, Vol. 4, No. 3, pp. 323-330, June 2013. (Published online: January 2012).
23. T. Yang, T. Oda, G. Mino, L. Barolli, A. Durresi and F. Xhafa, "Energy-saving in Wireless Sensor Networks Considering Mobile Sensor Nodes, Single and Multi Events", International Journal of Computer Systems Science and Engineering (IJCSSE), CRL Publishing, Vol. 27, No. 5, pp. 317-326, 2012.
24. T. Yang, T. Oda, L. Barolli and F. Xhafa, "Impact of Mobile Sink Nodes on Performance of Wireless Sensor Networks", International Journal of Information Technology, Communication and Convergence (IJITCC), Inderscience, Vol. 2. No. 2, pp. 155-170, 2012. (DOI: 10.1504/IJITCC.2012.048483)
25. A. Barolli, T. Oda, F. Xhafa, L. Barolli, M. Takizawa, K. Uchida, "Evaluation of WMN-GA for Different Mutation and Crossover Rates Considering Giant Component Parameter", Journal of Interconnection Networks, World Scientific Publishing, Vol.12, No.3, pp.205-219, 2012. (DOI: 10.1142/S0219265911002952)
26. T. Oda, A. Barolli, E. Spaho, F. Xhafa, L. Barolli, M. Takizawa, "Evaluation of WMN-GA for Different Mutation Operators", International Journal of Space-Based and Situated Computing (IJSSC), Inderscience, Vol. 2. No. 3, pp. 149-157, 2012. (DOI: 10.1504/IJSSC.2012.048896)
27. A. Barolli, M. Takizawa, T. Oda, E. Spaho, L. Barolli, K. Uchida, F. Xhafa, "Performance Evaluation of WMN-GA Simulation System for Different Settings of Genetic Operators Considering Giant Component and Number of Covered Users", International Journal of Distributed Systems and Technologies (IJDST), Vol. 3, No. 3, pp. 1-14, 2012. (DOI: 10.4018/jdst.2012070101)

28. T. Oda, A. Barolli, F. Xhafa, L. Barolli, M. Ikeda, M. Takizawa, "Performance Evaluation of WMN-GA for Different Mutation and Crossover Rates Considering Number of Covered Users Parameter", Journal of Mobile Information Systems (MIS), IOS Press, Vol. 8. No. 1, pp. 1-16, 2012. (DOI: 10.3233/MIS-2011-0128)
29. M. Hiyama, E. Kulla, T. Oda, M. Ikeda and L. Barolli, "Application of a MANET Testbed for Horizontal and Vertical Scenarios: Performance Evaluation Using Delay and Jitter Metrics", Human-centric Computing and Information Sciences (HCIS) Journal, Springer, Vol. 1, No. 3, pp. 1-14, 2011. (DOI: 10.1186/2192-1962-1-3)
30. A. Barolli, T. Oda, F. Xhafa, L. Barolli, M. Takizawa, K. Uchida, "Evaluation of WMN-GA for Different Mutation and Crossover Rates Considering Giant Component Parameter", Journal of Interconnection Networks (JOIN), World Scientific Publishing, Vol. 12, No. 3, pp. 205-219, 2011. (DOI: 10.1142/S0219265911002952)

International Conferences Papers (reviewed)

1. T. Oda, L. Barolli, "Experimental Results of a Raspberry Pi Based WMN Testbed Considering CPU Frequency", Accepted, The 30-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2016), 2016.
2. A. Barolli, T. Oda, L. Barolli, M. Takizawa, "Experimental Results of a Raspberry Pi and OLSR Based Wireless Content Centric Network Testbed Considering OpenWRT OS", Accepted, The 30-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2016), 2016.
3. I. Shinko, T. Oda, E. Spaho, A. Barolli, V. Kolici, L. Barolli, "A GA-Based Simulation System for WMNs: Performance Analysis of WMN-GA System for Different WMN Architectures and Uniform Distribution Considering DCF and EDCA", Accepted, The 30-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2016), 2016.
4. S. Sakamoto, T. Oda, M. Ikeda, L. Barolli, F. Xhafa, "Implementation of a New Replacement Method in WMN-PSO Simulation System and Its Performance Evaluation", Accepted, The 30-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2016), 2016.
5. D. Elmazi, S. Sakamoto, T. Oda, E. Kulla, E. Spaho, L. Barolli, "Effect of Security Parameter for Selection of Actor Nodes in WSAN: A Comparison Study of Two Fuzzy-Based Systems", Accepted, The 30-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2016), 2016.
6. T. Ishitaki, T. Oda, L. Barolli, "A Neural Network Based User Identification for Tor Networks: Data Analysis Using Friedman Test", Accepted, The 9-th International Workshop on Bio and Intelligent Computing (BICom-2016), 2016.
7. R. Obukata, T. Oda, L. Barolli, "Design of an Ambient Intelligence Testbed for Improving Quality of Life", Accepted, The 9-th International Symposium on Mining and Web (MAW-2016), 2016.
8. M. Yamada, T. Oda, K. Matsuo, L. Barolli, "Design of an IoT-Based E-Learning Testbed", Accepted, The 9-th International Symposium on Mining and Web (MAW-2016), 2016.
9. T. Inaba, S. Sakamoto, T. Oda, M. Ikeda, L. Barolli, "A QoS-aware Admission Control System for WLAN Using Fuzzy Logic", Accepted, The 12-th International Workshop on Heterogeneous Wireless Networks (HWISE-2016), 2016.
10. T. Oda, D. Elmazi, T. Ishitaki, K. Matsuo, A. Barolli, L. Barolli, "Evaluation of a Raspberry Pi Based WMN Testbed for Distributed Concurrent Processing and Multiple Flows in Indoor Environment", The 10-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2015), Krakow, Poland, November 4-6, 2015
11. T. Oda, D. Elmazi, T. Ishitaki, K. Matsuo, M. Ikeda and L. Barolli, "Implementation and Experimental Results of a Raspberry Pi and OLSR Based Wireless Content-Centric Network Testbed", The 10-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2015), pp. 263-268, Krakow, Poland, November 4-6, 2015.
12. A. Barolli, T. Oda, K. Matsuo, M. Ikeda, L. Barolli and M. Takizawa, "Experimental Results of a Raspberry Pi Based WMN Testbed for Different OSs in Indoor Environment Considering LoS Scenario", The 10-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2015), pp. 207-212, Krakow, Poland, November 4-6, 2015.
13. S. Sakamoto, T. Oda, M. Ikeda, L. Barolli and F. Xhafa, "A PSO-based Simulation System for Node Placement in Wireless Mesh Networks: Evaluation Results for Different Replacement Methods", The 10-th Inter-

national Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2015), pp. 213-219, Krakow, Poland, November 4-6, 2015.

14. I. Shinko, T. Oda, E. Spaho, V. Kolici, M. Ikeda and L. Barolli, “A Simulation System Based on ONE and SUMO Simulators: Performance Evaluation of First Contact, Prophet and Spray-and-Wait DTN Protocols”, The 10-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2015), pp. 137-142, Krakow, Poland, November 4-6, 2015.
15. D. Elmazi, T. Inaba, S. Sakamoto, T. Oda, M. Ikeda and L. Barolli, “Selection of Secure Actors in Wireless Sensor and Actor Networks Using Fuzzy Logic”, The 10-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2015), pp. 125-131, Krakow, Poland, November 4-6, 2015.
16. T. Inaba, S. Sakamoto, T. Oda, L. Barolli, “Performance Evaluation of a Secure Call Connection Admission Control for Cellular Wireless Networks Using Fuzzy Logic”, The 10-th International Conference on Broadband and Wireless Computing, Communication and Applications, Krakow, Poland, November 4-6, 2015.
17. T. Ishitaki, T. Oda, L. Barolli, “Application of Neural Networks and Friedman Test for User Identification in Tor Networks”, The 6-th International Workshop on Methods, Analysis and Protocols for Wireless Communication (MAPWC-2015), Krakow, Poland, November 4-6, 2015.
18. T. Oda, M. Ikeda, L. Barolli, F. Xhafa and M. Takizawa, “A GA-Based Simulation System for WMNs: Performance Analysis of WMN-GA System for Different WMN Architectures Considering DCF and EDCA”, The 7-th International Conference on Intelligent Networking and Collaborative Systems (INCoS-2015), pp. 232-238, Taipei, Taiwan, September 2-4, 2015.
19. S. Sakamoto, T. Oda, M. Ikeda and L. Barolli, “Design and Implementation of a Simulation System Based on Particle Swarm Optimization for Node Placement Problem in Wireless Mesh Networks”, The 7-th International Conference on Intelligent Networking and Collaborative Systems (INCoS-2015), pp. 164-168, Taipei, Taiwan, September 2-4, 2015.
20. A. Barolli, T. Oda, K. Matsuo, L. Barolli, F. Xhafa, V. Loia, “Performance Analysis of WMN-GA Simulation System for Different WMN Architectures and Routing Protocols Considering Exponential Distribution”, The 7-th International Conference on Intelligent Networking and Collaborative Systems (INCoS-2015), pp.38-44, Taipei, Taiwan, September 2-4, 2015.
21. V. Kolici, T. Oda, A. Barolli, A. Lala, L. Barolli, F. Xhafa, “Application of WMN-SA Web Interface and NS-3 for Optimization and Analysis in WMNs Considering Different Number of Mesh Routers and Architectures”, The 7-th International Conference on Intelligent Networking and Collaborative Systems (INCoS-2015), pp. 74-81, Taipei, Taiwan, September 2-4 2015.
22. T. Oda, K. Matsuo, L. Barolli, M. Ikeda, M. Takizawa, “Performance Evaluation of a WMN Testbed in Indoor Environment Considering Mobile Mesh Node Scenario”, The 18-th International Conference on Network-Based Information Systems (NBiS-2015), pp. 93-98, Taipei, Taiwan, September 2-4, 2015.
23. A. Lala, T. Oda, A. Barolli, V. Kolici, L. Barolli, F. Xhafa, “Application of WMN-HC Web Interface and NS-3 for Optimization and Analysis in WMNs Considering Different Number of Mesh Routers and Architectures”, The 18-th International Conference on Network-Based Information Systems (NBiS-2015), pp.85-92, Taipei, Taiwan, September 2-4 2015.
24. D. Elmazi, T. Oda, E. Kulla, L. Barolli, E. Spaho, “Selection of Rendezvous Point in Content Centric Networks Using Fuzzy Logic”, The 18-th International Conference on Network-Based Information Systems (NBiS-2015), pp. 345-350, Taipei, Taiwan, September 2-4 2015.
25. K. Matsuo, T. Oda, A. Barolli, L. Barolli, F. Xhafa, “Performance Analysis of WMN-GA Simulation System for Different WMN Architectures and Routing Protocols Considering Weibull Distribution”, The 18-th International Conference on Network-Based Information Systems (NBiS-2015), pp.78-84, Taipei, Taiwan, September 2-4 2015.
26. T. Ishitaki, T. Oda, K. Matsuo, L. Barolli, M. Takizawa, “Performance Evaluation of A Neural Network Based Intrusion Detection System for Tor Networks Considering Different Hidden Units”, The 18-th International Conference on Network-Based Information Systems (NBiS-2015), pp. 620-627, Taipei, Taiwan, September 2-4 2015.

27. T. Inaba, S. Sakamoto, T. Oda, L. Barolli, M. Takizawa, “A New FACS for Cellular Wireless Networks Considering QoS: A Comparison Study of FuzzyC with MATLAB”, The 18-th International Conference on Network-Based Information Systems (NBiS-2015), pp. 338-344, Taipei, Taiwan, September 2-4 2015.
28. T. Oda, D. Elmazi, E. Spaho, V. Kolici, L. Barolli, “A Simulation System Based on ONE and SUMO Simulators: Performance Evaluation of Direct Delivery, Epidemic and Energy aware Epidemic DTN Protocols”, The 5-th International Workshop on Information Networking and Wireless Communications (INWC-2015), pp. 418-423, Taipei, Taiwan, September 2015.
29. Y. Sakai, T. Oda, M. Ikeda and L. Barolli, “An Object Tracking System Based on SIFT and SURF Feature Extraction Methods”, The 5-th International Workshop on Information Networking and Wireless Communications (INWC-2015), pp. 561-565, Taipei, Taiwan, September 2-4, 2015.
30. E. Kulla, T. Oda, M. Ikeda and L. Barolli, “SAMI: A Sensor Actor Network Matlab Implementation”, The 5-th International Workshop on Information Networking and Wireless Communications (INWC-2015), pp. 554-560, Taipei, Taiwan, September 2-4, 2015.
31. T. Oda, A. Barolli, K. Matsuo, S. Sakamoto, M. Ikeda and L. Barolli, “Implementation and Experimental Results of a WMN Testbed in Indoor Environment Considering NLoS Scenario”, The 9-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2015), pp. 283-288, Blumenau, Brazil, July 8-10, 2015.
32. A. Barolli, T. Oda, L. Barolli, F. Xhafa, M. Takizawa, V. Loia, “Performance Analysis of WMN-GA Simulation System for Different WMN Architectures Considering TCP and OLSR Protocols”, The 9-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2015), pp. 57-63, Blumenau, Brazil, July 2015.
33. S. Sakamoto, A. Lala, T. Oda, V. Kolici, L. Barolli, F. Xhafa, “Analysis of WMN-HC Simulation System Data Using Friedman Test”, The 9-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2015), pp. 254-259, Blumenau, Brazil, July 2015.
34. D. Elmazi, T. Oda, E. Spaho, E. Kulla, L. Barolli, “A Selection of Actor Node in Wireless Sensor Actor Networks: A Case Study for Static and Mobile Actor Nodes”, The 9-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2015), pp. 36-43, Blumenau, Brazil, July 2015.
35. D. Elmazi, T. Oda, A. Lala, V. Kolici, L. Barolli, F. Xhafa, “Analysis of Node Placement in Wireless Mesh Networks Using Friedman Test: A Comparison Study for Genetic Algorithms and Hill Climbing”, The 9th International Workshop on Engineering Complex Distributed Systems (ECDS-2015), pp. 513-520, Blumenau, Brazil, July 2015
36. T. Oda, D. Elmazi, A. Lala, V. Kolici, L. Barolli, F. Xhafa, “Analysis of Node Placement in Wireless Mesh Networks Using Friedman Test: A Comparison Study for Tabu Search and Hill Climbing”, The 8-th International Workshop on Intelligent Informatics and Natural Inspired Computing (IINIC-2015), pp. 133-140, Blumenau, Brazil, July 2015.
37. T. Oda, Y. Liu, K. Matsuo, S. Sakamoto, M. Ikeda and L. Barolli, “Experimental Results of a CentOS-based Ad-Hoc Network Testbed Considering LoS Scenario”, The 9-th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2015), pp. 9-14, Blumenau, Brazil, July 8-10, 2015.
38. D. Elmazi, T. Oda, S. Sakamoto, E. Spaho, E. Kulla, L. Barolli, “Friedman Test for Analysing WMNs: A Comparison Study for Genetic Algorithms and Simulated Annealing”, The 9-th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2015), pp. 171-178, Blumenau, Brazil, July 2015.
39. K. Matsuo, T. Oda, D. Elmazi, S. Sakamoto, L. Barolli, “Performance Evaluation of OLSR, AODV and HWMP Protocols in Ad-Hoc Networks and MANET Scenarios”, The 9-th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2015), pp. 15-21, Blumenau, Brazil, July 2015.
40. V. Kolici, T. Oda, S. Sakamoto, E. Spaho, M. Ikeda, L. Barolli, “Performance Evaluation of a VANET Simulation System Using NS-3 and SUMO Considering Number of Vehicles and Crossroad Scenario”, The 9-th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2015), pp. 22-27, Blumenau, Brazil, July 2015.

41. T. Oda, A. Barolli, S. Sakamoto, L. Barolli, M. Ikeda, K. Uchida, "Implementation and Experimental Results of a WMN Testbed in Indoor Environment Considering LoS Scenario", The 29-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2015), pp. 37-42, Gwangju, Korea, March 25-27, 2015.
42. S. Sakamoto, T. Oda, L. Barolli, M. Ikeda, F. Xhafa, K. Uchida, "Application of WMN-SA Simulation System forWMN Node Placement in a Realistic Scenario", The 29-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2015), pp. 43-47, Gwangju, Korea, March 25-27, 2015.
43. A. Barolli, T. Oda, S. Sakamoto, L. Barolli, F. Xhafa, M. Takizawa, V. Loia, "Performance Analysis of WMN-GA Simulation System for Different WMN Architectures Considering OLSR", The 29th IEEE International Conference on Advanced Information Networking and Applications (AINA-2015), pp. 207-214, Gwangju, Korea , March 25-27, 2015.
44. V. Kolici, L. Barolli, S. Sakamoto, T. Oda, F. Xhafa, K. Uchida, "Optimization of Giant Component and Number of Covered Users in Wireless Mesh Networks: A Comparison Study", The 29th IEEE International Conference on Advanced Information Networking and Applications (AINA-2015), pp. 215-222, Gwangju, Korea, March 25-27, 2015.
45. V. Kolici, L. Barolli, T. Oda, S. Sakamoto, E. Spaho, M. Ikeda, "Performance Evaluation of a VANET Simulation System Using NS-3 and SUMO", The 29-th IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA-2015), pp. 348-353, Gwangju, Korea, March 25-27, 2015.
46. D. Elmazi, T. Oda, E. Spaho, E. Kulla, L. Barolli, K. Uchida, "Selection of Actor Nodes in Wireless Sensor and Actor Networks: A Fuzzy Based Method Considering Actor Mobility", The 11-th International Workshop on Heterogeneous Wireless Networks (HWISE-2015), pp. 304-310, Gwangju, Korea, March 25-27, 2015. "**Best Paper Award**"
47. T. Ishitaki, T. Oda, L. Barolli, K. Uchida, "Application of Neural Networks for Intrusion Detection in Tor Networks", The 8-th International Workshop on Bio and Intelligent Computing (BICom-2015), pp. 67-72, Gwangju, Korea, March 25-27, 2015.
48. T. Oda, S. Sakamoto, A. Barolli, M. Ikeda, L. Barolli and F. Xhafa, "A GA-Based Simulation System for WMNs: Performance Analysis for Different WMN Architectures Considering TCP", The 9-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2014), pp. 120-126, Guangzhou, China, November 8-10, 2014. "**Best Paper Award**"
49. E. Kulla, T. Oda, L. Barolli, "A Fuzzy-Based Method for Selection of Actor Nodes in Wireless Sensor and Actor Networks", The 9-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2014), pp. 1-7, Guangzhou, China, November 8-10, 2014.
50. S. Sakamoto, T. Oda, M. Ikeda, L. Barolli, S. Caballe and F. Xhafa, "Optimization of Number of Mesh Routers in WMNs Using WMN-SA System Considering Uniform Distribution for Different Number of Mesh Clients", The 9-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2014), pp. 133-138, Guangzhou, China, November 8-10, 2014.
51. A. Barolli, T. Oda, S. Sakamoto, M. Ikeda, L. Barolli, F. Xhafa, "Analysis of WMN-GA Simulation System Results: A Comparison Study for Node Placement in WMNs Considering Exponential and Weibull Distributions and Different Transmission Rates", The 9-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2014), pp. 139-144, Guangzhou, China, November 8-10, 2014.
52. X. Chang, S. Sakamoto, T. Oda, M. Ikeda, L. Barolli and F. Xhafa, "Node Placement in WMNs for Different Movement Methods: A Hill Climbing System Considering Exponential and Weibull Distributions", The 9-th International Workshop on Next Generation of Wireless and Mobile Networks (NGWMN-2014), pp. 440-445, Guangzhou, China, November 8-10, 2014.
53. T. Oda, S. Sakamoto, A. Barolli, M. Ikeda, F. Xhafa and L. Barolli, "Analysis of Mesh Router Node Placement Using WMN-GA System Considering Different Architectures of WMNs", The 17-th International Conference on Network-Based Information Systems (NBiS-2014), pp. 39-44, Salerno, Italy, September 10-12, 2014.

54. M. Ikeda, T. Oda, S. Sakamoto, T. Honda and L. Barolli, "Analysis of WMN-SA and WMN-GA Simulation Results: A Comparison Performance for Wireless Mesh Networks", The 17-th International Conference on Network-Based Information Systems (NBiS-2014), pp. 45-52, Salerno, Italy, September 10-12, 2014.
55. S. Sakamoto, T. Oda, M. Ikeda, F. Xhafa, L. Barolli and K. Uchida, "Performance Analysis of WMN-SA: Node Placement Problem Using Simulated Annealing Algorithm for Different Replacement Methods", The 17-th International Conference on Network-Based Information Systems (NBiS-2014), pp. 69-74, Salerno, Italy, September 10-12, 2014.
56. A. Barolli, T. Oda, S. Caballe, F. Xhafa, L. Barolli, M. Takizawa, "Analysis of WMN-GA Simulation Results: Optimization of Number of Mesh Routers Considering Exponential and Weibull Distributions of Mesh Clients", The 6-th International Conference on Intelligent Networking and Collaborative Systems (INCoS-2014), pp. 138-144, Salerno, Italy, September 10-12, 2014.
57. T. Oda, S. Sakamoto, E. Spaho, A. Barolli, F. Xhafa, L. Barolli, "Node Placement in WMNs using WMN-GA System Considering Normal and Uniform Distribution of Mesh Clients", The 8-th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS-2014), pp. 120-127, Birmingham, UK, July 2-4, 2014
58. A. Barolli, T. Oda, E. Spaho, L. Barolli, F. Xhafa, M.Takizawa, "WMN-GA for Node Placement in WMN: Evaluation and Visualization Using HotSpot Ad-Hoc Method", The 8-th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS-2014), pp. 23-29, Birmingham, UK, July 2-4, 2014
59. S. Sakamoto, T. Oda, E. Kulla, F. Xhafa, M. Ikeda and L. Barolli, "Evaluation of Effects of Grid Shape in WMN-SA System for Solution of Node Placement Problem in WMNs", The 8-th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS-2014), pp. 113-119, Birmingham, UK, July 2-4, 2014.
60. E. Spaho, X. Chang, T. Oda, M. Ikeda, F. Xhafa, L. Barolli, "Performance Evaluation of WMN-HC System for Different Number of Mesh Clients and Mesh Routers", The 8-th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS-2014), pp. 9-15, Birmingham, UK, July 2-4, 2014.
61. S. Caballe, L. Barolli, M. Feidakis, K. Matsuo, F. Xhafa, T. Daradoumis, T. Oda, "A Study of Using SmartBox to Embed Emotion Awareness through Stimulation into e-Learning Environments", The 4-th International Workshop on Adaptive Learning via Interactive, Collaborative and Emotional Approaches (ALICE-2014), pp. 469-474, Birmingham, UK, July 2-4, 2014.
62. D. G. Jimenez, S. Caballe, J. Conesa, T. Oda, L. Barolli, "Evaluation of Using Software Infrastructure and Multimedia Technologies for the Creation of Complex Learning Resources", The 4-th International Workshop on Adaptive Learning via Interactive, Collaborative and Emotional Approaches (ALICE-2014), Birmingham, UK, pp. 457-462, July 2-4, 2014.
63. T. Oda, A. Barolli, E. Spaho, L. Barolli, F. Xhafa, "Analysis of Mesh Router Placement in Wireless Mesh Networks Using Friedman Test", Proc. of The 28th IEEE International Conference on Advanced Information Networking and Applications (IEEE AINA-2014), pp. 289-296, Victoria, Canada, May 13-16, 2014.
64. M. Ikeda, T. Honda, T. Oda, S. Sakamoto and L. Barolli, "Analysis of WMN-GA Simulation Results: WMN Performance Considering Stationary and Mobile Scenarios", The 28-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2014), pp. 337-342, Victoria, Canada, May 13-16, 2014.
65. X. Chang, E. Spaho, T. Oda, L. Barolli, F. Xhafa, "Node Placement in WMNs Using WMN-HC System and Different Movement Methods", The 28-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2014), pp. 1148-1153, Victoria, Canada, May 13-16, 2014.
66. S. Sakamoto, T. Oda, A. Bravo, L. Barolli, M. Ikeda and F. Xhafa, "WMN-SA System for Node Placement in WMNs: Evaluation for Different Realistic Distributions of Mesh Clients", The 28-th IEEE International Conference on Advanced Information Networking and Applications (IEEE AINA-2014), pp. 282-288, Victoria, Canada, May 13-16, 2014.
67. T. Oda, E. Spaho, A. Barolli, L. Barolli, F. Xhafa, M.Takizawa, "Mesh Router Node Placement in Wireless Mesh Networks Considering Different Initial Router Placement Methods", The 8-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2013), pp. 61-67, Compiegne, France, October 28-30,2013.

68. S. Sakamoto, E. Kulla, T. Oda, L. Barolli, M. Ikeda and F. Xhafa, “Performance Comparison for Different Communication Distances of Mesh Routers in WMN-SA Simulation System”, The 8-th International Conference on Broadband, Wireless Computing, Communication and Applications (BWCCA-2013), pp. 74-79, Compiegne, France, October 28-30, 2013.
69. A. Barolli, T. Oda, E. Spaho, L. Barolli, F. Xhafa, M. Takizawa, “Effects of Selection Operators for Mesh Router Placement in Wireless Mesh Networks Considering Weibull Distribution of Mesh Clients”, The 8-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2013), pp. 68-73, Compiegne, France, October 28-30, 2013.
70. T. Oda, E. Spaho, A. Barolli, L. Barolli, F. Xhafa, M.Takizawa, “Performance Evaluation of WMN-GA for Wireless Mesh Networks Considering Mobile Mesh Clients”, The 5-th International Conference on Intelligent Networking and Collaborative Systems (INCoS-2013), pp. 77-84, X’ian, China, September 9-11 2013.
71. X. Chang, T. Oda, E. Kulla, M. Ikeda, L. Barolli, “Node Placement in WMNs Using Hill Climbing Algorithm: Performance Comparision of Normal and Uniform Distributions of Mesh Clients”, The 4-th International Conference on Emerging Intelligent Data and Web Technologies (EIDWT-2013), pp. 61-67, X’ian, China, September 9-11 2013.
72. S. Sakamoto, T. Oda, E. Kulla, M. Ikeda, L. Barolli, “Performance Comparison of a WMN-SA System for Different Distributions of Mesh Clients”, The 4-th International Conference on Emerging Intelligent Data and Web Technologies (EIDWT-2013), pp. 607-611, X’ian, China, September 9-11 2013.
73. T. Oda, E. Spaho, A. Barolli, L. Barolli, F. Xhafa, M. Takizawa, “A Comparison Study of GA and HC for Mesh Router Node Placement in Wireless Mesh Networks”, The 16-th International Conference on Network-Based Information Systems (NBiS-2013), pp. 206-213, Gwangju, Korea, September 4-6 2013.
74. S. Sakamoto, T. Oda, E. Kulla, M.Ikeda, L. Barolli, “Performance Evaluation of Mesh Router Node Placement Using Simulated Annealing Considering Exponential and Weibull Distributions”, The 16-th International Conference on Network-Based Information Systems (NBiS-2013), pp. 201-205, Gwangju, Korea, September 4-6 2013.
75. T. Oda, E. Spaho, A. Barolli, L. Barolli, F. Xhafa, M. Takizawa, “WMN-GA System for Node Placement in WMNs: Effect of Grid Shape”, Lecture Notes in Electrical Engineering, Information Technology Convergence: Security, Robotics, Automations and Communication, Springer, Vol. 253, No. 1, pp. 47-56, Fukuoka, Japan, July 8-10 2013. (DOI 10.1007/978-94-007-6996-0)
76. A. Barolli, T. Oda, F. Xhafa, L. Barolli, P. Papajorgji and M. Takizawa, “WMN-GA System for Node Placement in WMNs: Performance Evaluation for Weibull Distribution of Mesh Clients”, Lecture Notes in Electrical Engineering, Information Technology Convergence: Security, Robotics, Automations and Communication, Springer, Vol. 253, No. 1, pp. 223-231, Fukuoka, Japan, July 8-10 2013. (DOI 10.1007/978-94-007-6996-0)
77. S. Sakamoto, T. Oda, E. Kulla, M. Ikeda, L. Barolli and F. Xhafa, “Performance Evaluation of WMNs Using Simulated Annealing Algorithm Considering Different Number Iterations per Phase and Normal Distribution”, Lecture Notes in Electrical Engineering, Information Technology Convergence: Security, Robotics, Automations and Communication, Springer, Vol. 253, No. 1, pp. 169-176, Fukuoka, Japan, July 8-10 2013. (DOI 10.1007/978-94-007-6996-0) “**Best Paper Award**”
78. X. Chang, T. Oda, E. Spaho, M. Ikeda, L. Barolli and F. Xhafa, “Performance Evaluation of WMNs Using Hill Climbing Algorithm Considering Giant Component and Different Distributions”, Lecture Notes in Electrical Engineering, Information Technology Convergence: Security, Robotics, Automations and Communication, Springer, Vol. 253, No. 1, pp. 161-167, Fukuoka, Japan, July 8-10 2013. (DOI 10.1007/978-94-007-6996-0)
79. T. Oda, A. Barolli, E. Spaho, L. Barolli, F. Xhafa, “Performance Evaluation of WMN-GA system for Node Placement in WMNs for Normal and Uniform Distributions of Mesh Clients Considering Different Grid Shapes”, The 7-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2013), pp. 150-156, Taichung, Taiwan, July 3-5 2013.
80. A. Barolli, T. Oda, E. Spaho, L. Barolli, F. Xhafa, M. Takizawa, “Performance Evaluation of WMN-GA System for Node Placement in WMNs Considering Exponential and Weibull Distribution of Mesh Clients and Different Selection and Mutation Operators”, The 7-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2013), pp. 169-175, Taichung, Taiwan, July 3-5 2013.

81. S. Sakamoto, T. Oda, E. Kulla, M. Ikeda, L. Barolli and F. Xhafa, “Performance Analysis of WMNs Using Simulated Annealing Algorithm for Different Temperature Values”, The 7-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS 2013), pp. 164-168, Taichung, Taiwan, July 3-5 2013.
82. M. Ikeda, T. Honda, T. Oda, S. Sakamoto, X. Chang and L. Barolli, “Analysis of WMN-GA Simulation Results: WMN Performance Optimizing the Number of Mesh Routers”, The 7-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS 2013), pp. 157-163, Taichung, Taiwan, July 3-5 2013.
83. X. Chang, T. Oda, E. Spaho, M. Ikeda, L. Barolli and F. Xhafa, “Performance Analysis of WMNs Using Hill Climbing Algorithm Considering Normal and Uniform Distribution of Mesh Clients”, The 7-th International Workshop on Engineering Complex Distributed Systems (ECDS-2013), pp. 424-427, Taichung, Taiwan, July 3-5 2013.
84. T. Oda, E. Spaho, A. Barolli, L. Barolli, F. Xhafa, M. Takizawa, “Node Placement in WMNs: Performance Evaluation of WMN-GA System for Weibull and Exponential Distribution of Mesh Clients”, The 27-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2013), pp. 304-310, Barcelona, Spain, March 2013.
85. A. Barolli, T. Oda, E. Spaho, L. Barolli, F. Xhafa, M. Takizawa, “Performance Evaluation of WMN-GA System for Node Placement in WMNs Considering Normal Distribution of Mesh Clients and Different Selection and Mutation Operators”, The 27-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2013), pp. 312-317, Barcelona, Spain, March 25-28 2013.
86. M. Ikeda, T. Oda, E. Kulla, T. Honda, L. Barolli and F. Xhafa, “Analysis of WMN-GA Simulation Results:WMN Performance Considering Hot-spot Scenario”, The IEEE 27-th International Conference on Advanced Information Networking and Applications (AINA-2013), pp. 318-324, Barcelona, Spain, March 25-28 2013
87. T. Oda, A. Barolli, E. Spaho, L. Barolli, F. Xhafa, J. Iwashige, “Genetic Algorithms for Node Placement in WMNs: Effect of Changes in Population Size and Number of Generations”, The 7-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2012), pp. 356-361, Victoria, Canada, November 12-14 2012.
88. A. Barolli, T. Oda, E. Spaho, F. Xhafa, M. Takizawa, “Impact of Population Size and Number of Generations on the Performance of Dense WMNs”, The 7-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2012), pp. 523-528, Victoria, Canada, November 2012.
89. T. Yang, E. Kulla, T. Oda, L. Barolli, M. Younas, M. Takizawa, “Performance Evaluation of WSNs for MAC and Routing Protocols Considering Depletion and Routing Efficiency Metrics”, The 7-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2012), pp. 503-510, Victoria, Canada, November 12-14 2012.
90. M. Ikeda, T. Oda, E. Kulla, M. Hiyama, L. Barolli and M. Younas, “Performance Evaluation of WMN Considering Number of Connections Using NS-3 Simulator”, The 3-rd International Workshop on Methods, Analysis and Protocols for Wireless Communication (MAPWC-2012), pp. 498-502, Victoria, Canada, November 12-14 2012.
91. T. Oda, A. Barolli, E. Spaho, L. Barolli, F. Xhafa, J. Iwashige, “Node Placement in WMNs and Visualization of Evolutionary Computation Process Using WMN-GA System”, The 15-th International Conference on Network-Based Information Systems (NBiS-2012), pp. 214-220, Melbourne, Australia, September 2012.
92. T. Yang, E. Kulla, T. Oda, L. Barolli, M. Takizawa, “Performance Evaluation of WSNs Considering MAC and Routing Protocols Using Goodput and Delay Metrics”, The 15-th International Conference on Network-Based Information Systems (NBiS-2012), pp. 341-348, Melbourne, Australia, September 2012.
93. A. Barolli, T. Oda, E. Spaho, L. Barolli, F. Xhafa, M. Takizawa, “Visualization of Evolutionary Computation Process for Node Placement in WMNs Considering Weibull and Exponential Distribution of Mesh Clients”, Proc. of The 3-rd International Conference on Emerging Intelligent Data and Web Technologies (EIDWT-2012), pp. 9-15, Bucharest, Romania, September 2012.
94. T. Oda, A. Barolli, E. Spaho, F. Xhafa, L. Barolli, M. Takizawa, “Performance Evaluation of WMN-GA System for Low Densities of Clients and Different Settings of Population Size”, The 6-th International

Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2012), pp. 314-319, Palermo, Italy, July 4-6 2012.

95. A. Barolli, T. Oda, E. Spaho, L. Barolli, F. Xhafa, M. Takizawa, “Performance Evaluation of WMN-GA System for Dense Networks Considering Different Distributions”, The 6-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2012), pp. 523-528, Palermo, Italy, July 4-6 2012.
96. M. Hiyama, E. Kulla, T. Oda, M. Ikeda, L. Barolli, M. Takizawa, “Performance Investigation of a MANET Testbed in Outdoor Stairs Environment for Different Scenarios”, The 6-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2012), pp. 284-289, Palermo, Italy, July 4-6 2012.
97. E. Kulla, T. Oda, M. Ikeda, L. Barolli, F. Xhafa, M.Takizawa, “Multimedia Transmissions over a MANET Testbed: Problems and Issues”, The 6-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2012), pp.141-147, Palermo, Italy, July 4-6 2012.
98. T. Yang, T. Oda, E. Kulla, L. Barolli, A. Durresi, M. Takizawa, “Performance Evaluation of WSNs for Different MAC Protocols Considering TwoRayGround Radio Model and AODV Routing Protocol”, The 6-th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS-2012), pp. 320-326, Palermo, Italy, July 4-6 2012.
99. T. Oda, A. Barolli, E. Spaho, F. Xhafa, L. Barolli, M. Takizawa, K. Uchida, “Effect of Population Size for Node Placement in WMNs Considering Giant Component and Number of Covered Users Parameters”, The 26-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2012), Fukuoka, Japan, pp. 454-459, March 26-29 2012. “**Highly Commended Paper Award**”
100. M. Hiyama, E. Kulla, T. Oda, M. Ikeda and L. Barolli, “Experimental Results of a MANET Testbed in a Mixed Environment Considering Horizontal and Vertical Topologies”, The 26-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2012), pp. 884-889, Fukuoka, Japan, March 26-29 2012.
101. E. Kulla, M. Ikeda, T. Oda, L. Barolli, F. Xhafa and A. Biberaj, “Evaluation of a MANET Testbed in Outdoor Bridge Environment Using BATMAN Routing Protocol”, The 26-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2012), pp. 384-390, Fukuoka, March 26-29 2012.
102. T. Yang, T. Oda, L. Barolli, F. Xhafa, A. Durresi and J. Iwashige, “Investigation of Packet Loss in Mobile WSNs for AODV Protocol and Different Radio Models”, The 26-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2012), pp. 709-715, Fukuoka, Japan, March 26-29 2012.
103. E. Spaho, T. Oda, A. Barolli, F. Xhafa, L. Barolli, M. Takizawa, “A Comparison Study for Different Settings of Crossover and Mutation Rates Using WMN-GA Simulation System”, Lecture Notes in Electrical Engineering, Computer Science and Convergence: CSA-2011 & WCC-2011 Proceedings, Vol. 114, pp. 643-650, Jeju, Korea, December 2011. (DOI 10.1007/978-94-007-2792-2_62)
104. L. Barolli, E. Spaho, T. Oda, A. Barolli, F. Xhafa, M. Takizawa, “Performance Evaluation for Different Settings of Crossover and Mutation Rates Considering Number of Covered Users: A Case Study”, The 9th International Conference on Advances in Mobile Computing & Multimedia (MoMM-2011), pp. 110-115, Ho Chi Minh City, Vietnam, December 2011.
105. T. Oda, A. Barolli, E. Spaho, F. Xhafa, L. Barolli, “Effects of Mutation and Crossover in Genetic Algorithms for Node Placement in WMNs Considering Number of Covered Users Parameter”, The 3-rd IEEE International Conference on Intelligent Networking and Collaborative Systems (INCoS-2011), pp. 188-195, Fukuoka, Japan, November-December 2011.
106. A. Barolli, T. Oda, E. Spaho, F. Xhafa, L. Barolli, M. Takizawa, “Effects of Mutation and Crossover in Genetic Algorithms for Node Placement in WMNs Considering Giant Component Parameter”, The 6-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2011), pp. 18-25, Barcelona, Spain, October 2011. “**Best Paper Award**”
107. T. Oda, A. Barolli, E. Spaho, F. Xhafa, L. Barolli, M. Takizawa, “Performance Evaluation of WMN Using WMN-GA System for Different Mutation Operators”, The 14-th International Conference on Network-Based Information Systems (NBiS-2011), pp. 400-406, Tirana, Albania, September 2011.

108. T. Oda, A. Barolli, E. Spaho, F. Xhafa, L. Barolli, M. Takizawa, "GA-based System for WMN and Its Performance Evaluation for Different Scenarios", The 5-th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS-2011), pp. 402-408, Seoul, Korea, June-July 2011.

Domestic Conferences Papers

1. 槙 俊孝, 大塚 信吾, 坂本 真仁, 本田 泰規, 小田 哲也, 高橋 和生, 稲葉 貴昭, "2014 年度 FIT Student Branch の活動報告", 第 22 回電子情報通信学会九州支部学生会講演会, Kagoshima, September 2014.
2. 槙 俊孝，本田 泰規，大塚 信吾，小田哲也，稲田 龍一，小峰 輝俊，高橋 和生，若原 俊彦, "Linked Data の時間管理を実現するデータベースの提案", 第 22 回電子情報通信学会九州支部学生会講演会, Kagoshima, September 2014.
3. 大塚 信吾，小田 哲也，槙 俊孝，本田 泰規，坂本 真仁，高橋 和生，若原 俊彦, "公衆無線 LAN を用いた JR 位置情報通知システムの提案", 第 22 回電子情報通信学会九州支部学生会講演会, Kagoshima, September 2014.
4. 坂本 真仁, 小田 哲也, クラ エリス, 池田 誠, バロリ レオナルド, "WMN-SA システムの評価：エリアサイズ規模と SA 温度パラメータの関係", 情報処理学会, 第 158 回マルチメディア通信と分散処理研究発表会 (DPS-158), Tokyo, March 2014.
5. 坂本 真仁, 小田 哲也, バロリ レオナルド, 池田 誠, "フェイズ毎の繰り返し回数と温度パラメータを考慮した WMN-SA の性能評価", 情報処理学会, 第 157 回マルチメディア通信と分散処理研究発表会 (DPS-157), Shizuoka, October 2013.
6. 坂本 真仁, 小田 哲也, バロリ レオナルド, 池田 誠, "WMN における Simulated Annealing を用いたメッシュルータ配置最適化手法の提案と評価", 第 21 回電子情報通信学会九州支部学生会講演会, Kumamoto, September 2013.
7. 小田 哲也, 坂本 真仁, 池田 誠, バロリ レオナルド, "多目的遺伝的アルゴリズムを用いたメッシュルータ配置における様々な遺伝的オペレータの検討", 信学技報, Vol. 112, No. 308, MoMuC2012-37, pp. 27-31, Fukuoka, November 2012.
8. 小田哲也, 坂本 真仁, 池田 誠, バロリ レオナルド, 岩重 二郎, "多目的遺伝的アルゴリズムを用いたメッシュルータ配置法に関する研究", 第 20 回電子情報通信学会九州支部学生会講演会, Nagasaki, September 2012.
9. 小田哲也, 池田 誠, バロリ レオナルド, "無線メッシュネットワークにおいて遺伝的アルゴリズムを用いたメッシュルータ配置法"， 第 19 回電子情報通信学会九州支部学生会講演会, Saga, September 2011.

Awards

1. "Best Paper Award" of The 29-th IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA-2015)
2. "Best Paper Award" of The 9-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2014)
3. "Best Paper Award" of The 5-th International Conference on Information Technology Convergence and Services (ITCS-2013)
4. "学術奨励賞" of IEICE Kyusyu Section (平成 24 年度 電子情報通信学会九州支部)
5. "Highly Commended Paper Award" of The 26-th IEEE International Conference on Advanced Information Networking and Applications (AINA-2012)
6. "Best Paper Award" of The 6-th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA-2011)