A Hybrid Nature-Inspired Optimizer for Wireless Mesh Networks Design

D. Benyamina\textsuperscript{1}, A. Hafid\textsuperscript{1}, N. Hallam\textsuperscript{1}, M. Gendreau\textsuperscript{\textsc{ii}}, J. C. Maureira\textsuperscript{\textsc{ii}}

\textsuperscript{1}NRL, University of Montreal, Canada
\textsuperscript{\textsc{ii}}CIRRELT, University of Montreal, Canada
\textsuperscript{1}CS&IT Dept., University of Bahrain, Bahrain
\textsuperscript{\textsc{ii}}INRIA, Sophia Antipolis, France

Abstract

Existing approaches for optimal planning of Wireless Mesh Networks (WMNs) deployment revolves around the deployment cost as the pivotal concept to optimize. In this paper, we adopt a new approach to optimize the planning of WMNs that guarantees an acceptable level of network performance prior to its deployment. It is a simultaneous optimization process of network deployment cost and network throughput objectives while taking into account all the parameters that have a significant impact on the network efficiency. We propose three multi-objective models for WMN planning problem, namely Load-Balanced model, Interference model, and Flow-Capacity model. We devise an evolutionary swarm-based algorithm that is a hybrid combination of Multi-Objective Particle Swarm Optimization (MOPSO) and Genetic Algorithms (GAs) to solve the three models. We use realistic network sizes (up to 100 mesh nodes) to perform a thorough comparative experimental study on these three instance models with different key-parameter settings. Finally, we use the network simulator OMNET++ to evaluate the three models in terms of the actual performance (network throughput). The results presented in this paper show that Load-Balanced Model totally supersedes the Flow-Capacity model and performs better than the Interference Model.

Keywords Wireless Mesh Network, Planning problem, Multi-objective optimization, Particle Swarm Optimization.

1 INTRODUCTION

Many real-world engineering optimization problems are characterized by multiple and often conflicting objectives to optimize and a huge search space to explore. The planning of a WMN is one of those complex optimization problems. A proper design of a WMN is a fundamental task and should be addressed carefully to determine the network efficiency in terms of coverage, throughput, and capacity.
WMNs [1] are multi-hop networks of wireless routers. Multi-hop infrastructure WMNs offer increased reliability, coverage and reduced equipment costs over their single-hop counterpart, Wireless Local Area Networks (WLANs). A WMN consists of a set of mesh nodes, offering connectivity to end user devices. The mesh nodes form a relatively-static infrastructure, composed of Access Points (APs), router/relays (MRs), and gateways (MGs) nodes for forwarding messages, and orthogonal channels using multi-radios interfaces for allowing simultaneous communications. AP nodes are the main servers to mesh clients. They also interconnect with each other through point-to-point wireless links using relay routers (MRs). Gateways are the main interface to the Internet backbone connection; they act as bridges between the wireless infrastructure and the Internet and do have extra functionalities which make them more expensive than routers. Fig. 1 illustrates a typical WMN infrastructure.

![Wireless mesh network infrastructure.](image)

Deploying such technologies requires considerable budgets even if they are relatively cheaper than other technologies such as (3G), and therefore any optimization strategy that minimizes the cost while providing a Quality of Service (QoS) is very much sought after. In fact, earlier WMNs deployments have been linked with a number of problems such as intermittent connectivity, poor performance and lack of coverage [2]. Moreover, the QoS is not necessarily better supported when the number of mesh nodes increases. Indeed, it is naively tempting to correlate the increase of the number of mesh nodes to, for instance, a better coverage or higher network throughput. On the contrary, it is quite usual that a mere increase in the number of mesh nodes usually increases the complexity of the channel assignment problem thus inducing high interference levels, which results in network performance degradation. Therefore, there is a need to develop sound solutions for optimizing the planning of WMNs.

Significant research has focused mainly on the problem of performance improvement of WMNs where the locations of routers and gateways are fixed a priori (i.e., fixed topology) [4, 5, 6, and 7]. Other studies (e.g., [8, 9]), consider topologies where the locations of gateways are fixed beforehand. Sen et al. [8] propose a planning solution for rural area networks to provide a set of villages with network connectivity from a given landline node (a positioned MG); they study the optimization problem as the minimization of the total cost of the multi-hop network topology and the antenna tower heights under the
constraints of throughput, power, and interference. The problem is broken down into four sub-problems: topology search, optimum height assignment, antenna assignment, and power assignment; for each sub-problem, they provide a formulation and apply a different solution technique. Chen et al. [9] consider the deployment plan for mesh routers that are equipped with directional antennas to form the mesh backbone (an urban WMN is considered). They make the assumption that that the locations of gateways are fixed a priori; the objective is to maximize the deployment profit and maintain the cost within the budget. However, they do not consider the power/channel assignment problem. The studies in [10, 11, 12, 13, 14, 15, 16, 17, 18] attempt to optimize the number of gateways given a fixed layout of mesh routers; the objective is to find the optimal locations of gateways that best fulfill a restricted set of requirements. Chandra et al. [10] address the problem of minimizing the number of gateways while satisfying the traffic demands by using a network flow model. Aoun et al. [11] propose a technique to place a minimum number of MGs, such that the three constraints of throughput, power, and interference are satisfied; the technique consists of using a recursive algorithm to divide the WMN into clusters of bounded radius under relay load and cluster size constraints. Robinson et al [14] study the MG placement problem as facility location and k-median problems; they propose two local search algorithms (namely minhopcount and mincontention) to estimate the unknown capacities of MGs. Li et al. [15] investigate how to place the MGs in the given mesh infrastructure in order to achieve optimal throughput. Hsu et al. [17] model the MG placement problem as a combinatorial optimization problem; they propose two algorithms namely, Self-Constituted Gateway Algorithm (SCGA) and Predefined Gateway Set Algorithm (PGSA). Both algorithms make use of a genetic search algorithm to search for feasible configurations coupled with a modified version of Dijkstra’s algorithm to look for paths with bounded delays.

Recent contributions in [3, 19, and 20] propose WMN planning schemes where the locations of routers and gateways are not fixed. Amaldi et al. [3, 20] construct and formulate the planning model of WMNs as an Integer Linear Problem (ILP) based on user-coverage satisfaction; however, they do not consider QoS requirements such as delay and throughput. Beljadjid et al. [19] propose a unified model for WMN design formulated as an ILP problem; the objective is to minimize the total installation cost by tuning all the network parameters where they consider the delay as a constraint. The contributions in [3, 19] opted for exact optimization techniques (most of them implemented using CPLEX) to find optimal planning solutions; this makes these techniques suitable only for small to medium-sized instance problems. However, Amaldi et al. [20] later developed heuristics (based on greedy selection) to solve the proposed ILP [19].

The common trait between all these related work regardless of the (fixed/non-fixed) topologies and the (exact/heuristic) optimization methods used is that they all consider a variant of single-objective optimization model. More precisely, the total deployment cost is the sole objective to optimize under other relevant QoS constraints. The only promising study [18] that deviates from this trend proposes a bi-objective optimization model for the gateway placement problem, where a weighted aggregate objective function is used. Aggregating many objectives into a single objective function has been (and is still being) successfully used in many optimization projects. However, the key shortcoming of the
aggregation approach, as is widely known in the multi-objective community, is its inability to find potential candidate solutions when the landscape of the objective functions is non-convex [21].

The Multi-Objective (MO) optimization approach produces several non-dominated solutions. Optimality in MO optimization problems is redefined by the non-dominance concept, better known as Pareto optimality, where none of the (non-dominated) solutions is better than the rest with respect to all objectives. Moreover, this set of “trade-off” solutions does naturally reflect the multi-criteria decision making used by engineers -who usually prefer multiple non-dominated solutions where each can be used in a different decision making scenario.

Up to date and to the best knowledge of the authors, there has been no attempt to model WMN planning problems using a pure MO optimization approach. Basically, we can argue that a WMN planning problem can be seen as an optimization problem where the two most important objectives are the deployment cost (to minimize) and the network performance (to maximize). Minimizing the deployment cost is mainly achieved by deploying less network devices (routers/gateways), but this will create longer delays in user traffics and induce bottlenecks, which undermine the network performance. Similarly, maximizing network performance can be achieved by strategically placing extra network devices, which fattens the deployment cost budget. Often, as in the previous example, the objectives in a multi-objective optimization problem do conflict with each other in the sense that an increase in one objective dimension undermines another objective. This clearly plays in favor of multi-objective optimizers as they are the best methods to return a spectrum of trade-off solutions.

The main contributions, in this paper, can be summarized as follows.

- We devise a generic MO optimization framework that captures and reflects the essence and the true nature of a WMNs planning problem. The goal is to minimize the cost and maximize the overall network performance. We propose a population-based MO optimizer in order to produce several non-dominated planning solutions, from which the network planner can choose those that better suit his/her budget and resources.

- As individual protocols are typically specified for a priori fixed topologies with different assumptions in mind, the end-to-end performance of these protocols in deployed wireless networks has not been always satisfactory. We do not assume any a priori fixed topologies and our planning WMNs solutions are constructed from scratch and in an incremental way to meet the QoS requirements, taking into consideration interference aware model while meeting the planner’s objectives and satisfying the relevant constraints.

- As network performance (throughput) can be optimized under different perspectives, we propose three different metrics. This led us to design three multi-objective WMN planning models.

- Genetic Algorithms (GAs) are very adequate in exploring huge and complex search spaces while Multi-Objective Particle Swarm Optimization (MOPSO) gets better results faster and cheaper; furthermore, MOPSO is easy to implement and requires few parameters to adjust; however, it does not perform well for discrete search spaces. We design a hybrid meta-heuristic evolutionary algorithm
based on GAs and MOPSO to solve the three WMNs planning models. A thorough comparative experimental study is then provided by tuning different WMN key-parameter on the three optimization models. Then, a network simulation using OMNET++ is also conducted to actually measure the performance of three specific (same-priced) topologies derived from the three models.

The rest of the paper is organized as follows. Section 2 describes the mathematical formulation of the three WMN planning optimization models. The evolutionary meta-heuristic algorithm to solve the proposed bi-objective models is detailed in Section 3. Experimental numerical results and a comparative analysis are presented in Section 4. Finally, we conclude the paper in Section 5.

2 MULTI-OBJECTIVE MODELING APPROACH AND FORMULATION

In this section, we present the terminology and notations used in describing our modeling approach. We then formulate three theoretical bi-objectives optimization models.

2.1 Wireless Mesh Network Planning Problem

Fig. 2 is a simplistic depiction of the terminology explained below.

![Fig. 2. WMN Planning Problem: Key parameters and variables.](image)

We define \( I = \{1, \ldots, n\} \) as the set of positions of \( n \) Traffic Spots (TSs) concentrations in the service area and \( L = \{1, \ldots, m\} \) as the set of positions of \( m \) Candidate Locations (CLs) where mesh nodes can be installed.

The planning problem aims at:

- Selecting a subset \( S \subseteq L \) of CLs where a mesh node should be installed so that the signal level is high enough to cover the considered TSs.
- Defining the gateway set by selecting a subset \( G \subseteq L \) of CLs where the wireless connectivity is assured.
• Maintaining the cardinalities of $G$ and $S$ as small as possible to meet the financial and performance requirements.

In the following, unless otherwise stated, $i$ and $j$ belong to $I$ and $L$ respectively. The traffic generated by TS$_i$ is denoted by $d_i$, while $u_{jl}$ is the traffic capacity of the wireless link between CL$_{j}$ and CL$_l$. The capacity of the radio access interface of an access point AP located at CL$_j$ is denoted by $v_j$. The WMN planning key parameters are $e_j$ the cost associated to installing a mesh node (AP, MR or MG) at location CL$_j$, and $p_j$ the additional cost required to install a gateway (MG) at that location.

2.1.1 Network coverage and connectivity setup.

The network coverage $a_{ij}$ and network connectivity $b_{jl}$ are the two main WMN planning parameters. The network coverage is a binary matrix that states whether a client at TS$_i$ can be covered by one or many locations CL$_j$.

$$a_{ij} = \begin{cases} 1 & \text{if TS}_i \text{ covered by } \text{CL}_j \\ 0 & \text{otherwise} \end{cases}$$

The network connectivity is a binary matrix and indicates whether two locations can be wirelessly connected.

$$b_{jl} = \begin{cases} 1 & \text{if CL}_j \text{ and CL}_l \text{ can be wirelessly connected} \\ 0 & \text{otherwise} \end{cases}$$

The main decision vector variables (see Fig.2) are the routers installation locations, the gateways installation locations and the assignment of the users TS$_i$ to CL$_j$.

$$t_{ij} = \begin{cases} 1 & \text{if a device installed at CL}_j \\ 0 & \text{otherwise} \end{cases}$$

$$g_{ij} = \begin{cases} 1 & \text{if a gateway installed at CL}_j \\ 0 & \text{otherwise} \end{cases}$$

$$x_{ij} = \begin{cases} 1 & \text{if TS}_i \text{ assigned to router at CL}_j \\ 0 & \text{otherwise} \end{cases}$$

2.1.2 Mesh Node Installation, Radio/Channel and Flow setup.

We suppose initially that mesh nodes operate using the same number of radios $R$, each with $k$ channels, $(k>R)$ and $k \in C$, where $C = \{1, \ldots, c\}$ and $c$ can be at most 12 orthogonal channels if IEEE802.11a is used.

Other extra installation variables are needed in a Multi-Radio Multi-Channel WMN:
• $z_{jq} = 1$ if a mesh node is installed at CL$_j$ and is assigned channel $q$, $q \leq k$, 
• \( y_{jl}^q = 1 \) if a there is a wireless link from a mesh node installed at \( CL_j \) to a mesh node installed at \( CL_l \) using channel \( q \), \( q \leq k \).

• \( N_{jl} \) is the set of links that cannot be simultaneously active with the link \( y_{jl}^q \).

Finally, we define the flow variables \( f_{jl}^q \) and \( F_j \). The variable \( f_{jl}^q \) denotes the traffic flow routed from a router in \( CL_j \) to a router in \( CL_l \) using channel \( q \). The variable \( F_j \) is the traffic flow on the wired link between a gateway at \( CL_j \) and the Internet.

For better readability, Table I summarizes the notation used in the problem formulation.

Table I: list of symbols used in the WMN design Models.

<table>
<thead>
<tr>
<th>Param./Var.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( AP )</td>
<td>Access Point</td>
</tr>
<tr>
<td>( MR )</td>
<td>Mesh Router</td>
</tr>
<tr>
<td>( MG )</td>
<td>Mesh Gateway</td>
</tr>
<tr>
<td>( n )</td>
<td>Number of Traffic Spots (TSs)</td>
</tr>
<tr>
<td>( m )</td>
<td>Number of Candidate Locations (CLs)</td>
</tr>
<tr>
<td>( d_i )</td>
<td>Traffic generated by TS ( i )</td>
</tr>
<tr>
<td>( u_{jl} )</td>
<td>Traffic capacity of wireless link (( CL_j,CL_l ))</td>
</tr>
<tr>
<td>( v_j )</td>
<td>Capacity limit for AP radio access interface</td>
</tr>
<tr>
<td>( e_j )</td>
<td>A device cost installation</td>
</tr>
<tr>
<td>( p_j )</td>
<td>A gateway additional cost installation</td>
</tr>
<tr>
<td>( R )</td>
<td>Number of radio interfaces</td>
</tr>
<tr>
<td>( k )</td>
<td>Number of channels</td>
</tr>
<tr>
<td>( a_{ij} )</td>
<td>Coverage of TS ( i ) by CL ( j )</td>
</tr>
<tr>
<td>( b_{jl} )</td>
<td>Wireless connectivity between CL ( j ) and CL ( l )</td>
</tr>
<tr>
<td>( t_j )</td>
<td>Installation of a device at CL ( j )</td>
</tr>
<tr>
<td>( g_j )</td>
<td>Installation of a gateway at CL ( j )</td>
</tr>
<tr>
<td>( x_{ij} )</td>
<td>Assignment of TS ( i ) to CL ( j )</td>
</tr>
<tr>
<td>( z^q_{jl} )</td>
<td>Installation of a device at CL ( j ), assignment of channel ( q ), ( q &lt; k )</td>
</tr>
<tr>
<td>( y_{jl}^q )</td>
<td>Establishing a wireless communication on channel ( q ) between (( CL_j,CL_l ))</td>
</tr>
<tr>
<td>( f_{jl}^q )</td>
<td>Flow on channel ( q ) between (( CL_j,CL_l ))</td>
</tr>
<tr>
<td>( F_j )</td>
<td>Flow on the wired link from CL ( j ) to Internet</td>
</tr>
<tr>
<td>( N_{jl} )</td>
<td>Set of links interfering with the link ( y_{jl}^q )</td>
</tr>
</tbody>
</table>

Next, we propose three instance models that attempt to simultaneously minimize the network deployment cost and maximize the network throughput. They differ, however, in modeling the network throughput objective.

### 2.2 WMN planning Optimization Models

Optimal WMN planning solutions under multi-objective approach are more realistic and much preferred by network planners in that they have to be cost-effective (the deployment cost is minimized while the throughput is maximized). While the deployment cost objective is straightforward, the throughput objective function can be viewed from different perspectives. The throughput objective function
could be maximized by balancing the loads over network channels, minimizing the aggregation of network interferences, or maximizing the culmination of the flows over the entire network.

### 2.2.1 Load-Balanced Model

It seems plausible to enhance the quality of service by minimizing contentions and traffic bottlenecks. One way to achieve this is to properly balance the loads on the links over the whole network. We formulate the Load-Balanced (optimization) model as follows:

\[
\begin{align*}
\text{min} & \quad \sum_{i \in I} \left[ c_i f_j + p_j g_j \right] \\
\text{subject to:} & \quad \sum_{i \in I} f_{ij} = 1 \quad \forall i \in I \tag{1} \\
& \quad x_{ij} \leq a_i f_j \quad \forall i \in I, \forall j \in L \tag{4} \\
& \quad \sum_{i \in L} d_i x_{ij} + \sum_{l \in L, q \in C} \left( f_{ij}^q - f_{ij}^p \right) - F_j = 0 \quad \forall j \in L \tag{5} \\
& \quad \sum_{k \in N_{ij}} y_{ij}^q \leq 1 \quad \forall q \in C, \forall j \in L \tag{6} \\
& \quad f_{ij}^q \leq y_{ij}^q \quad \forall q \in C, \forall j \in L \tag{7} \\
& \quad \sum_{i \in I} d_i x_{ij} \leq v_j \quad \forall j \in L \tag{8} \\
& \quad F_j \leq W g_j \quad \forall j \in L \tag{9} \\
& \quad 2y_{ij}^q \leq b_j \left( z_j^f + z_j^g \right) \quad \forall q \in C, \forall j \in L \tag{10} \\
& \quad g_j \leq t_j \quad \forall j \in L \tag{11} \\
& \quad \sum_{l \in L} y_{ij}^q \leq 1 \quad \forall q \in C, \forall j \in L \tag{12} \\
& \quad \sum_{q \in C} z_j^q \leq R t_j \quad \forall j \in L \tag{13}
\end{align*}
\]
\[ x_{ij}, x^q_{ij}, y^q_{ij}, t_j, s_j \in \{0,1\} \quad \forall i \in I, \forall j, l \in L, \forall q \in C \quad (14) \]

\[ f^q_{jl}, F_j \in R \quad \forall j, l \in L, \forall q \in C \quad (15) \]

In this model, the function objective (1) minimizes the total cost of the network including installation cost \( e_j \) and additional gateway installation cost \( p_j \). The load-balanced objective function (2.a) is the minimization of the standard deviation of the ratio of traffic flows over the network links.

Constraint (3) and Constraint (4) assign a TS \( s_i \) to an Access Point (AP) installed at location \( CL_j \). Constraint (3) makes sure that the TS \( s_i \) is assigned to exactly one and only one AP installed at \( CL_j \), while constraint (4) implies that the TS \( s_i \) and the assigned AP are within the coverage area.

Constraint (5) defines the flow balance for each mesh node at \( CL_j \). Constraint (6) limits link interferences, while inequalities (7) and (8) respectively defines the flow-link capacity and the demand-radio access capacity constraints. Constraint (9) stipulates that the flow routed to the wired backbone is different from zero only when the mesh node installed is a gateway. We set \( W \) with a very large number to limit the capacity of the installed gateway.

Constraint (10) forces a link between \( CL_j \) and \( CL_l \) using the same channel \( q \) to exist only when the two devices are installed, wirelessly connected and tuned to the same channel \( q \). Constraint (11) ensures that a device can be a gateway only if it is installed.

Constraint (12) prevents a mesh node from selecting the same channel more than once to assign it to its interfaces. Constraint (13) states that the number of links emanating from a node is limited by the number of its radio interfaces. It also states that if a channel is assigned only once to a mesh node, it is a sufficient condition for its existence.

All the above constraints are called hard constraints with the exception of constraint (5) (which is then called a soft constraint). The fact to have only constraint (5) violated while other constraints are satisfied can be explained by the inability to find routes to flow the traffic generated by mesh clients. This is mainly caused by the lack of node pairs that are tuned to the same channel to establish wireless communications. Therefore, a reassignment of channels for the same topology in later iterations (using mutation) could help in satisfying all traffic demand (constraint 5 will be then fulfilled).

### 2.2.2 Interference Model

Because of the limited number of orthogonal channels, the spatial reuse of channels results in high level of interferences. This naturally degrades the network performance which is reflected by an overall throughput decrease.

Therefore, we may argue that the overall network interference, modeled by constraint (6), is sufficiently important for the network performance to be elevated to the status of an objective function that is to be minimized. For this purpose, we propose a novel performance (interference level) metric that we call Balanced Channel Repartition (BCR). It is defined as follows:
\[ \varphi_q = \max \left| O_q - O_{q'} \right| \quad \forall q, q' \in C, \ q \neq q' \]

Where,
\[ O_q = \sum_{j \in L} y_{qj} \quad \forall q \in C \]

In other words, the number of occurrences of channel \( q \), denoted by \( O_q \), is used to compute the gap between the balanced allocation of channel \( q \) and the current allocation. Our aim is to minimize this gap.

The second objective function is then defined as
\[ \min \sum_{q \in C} \varphi_q \]

Fig. 3 shows that the spatial channel reuse in (b) is better than that in (a). The value of total \( \varphi_q \) in (a) is equal to 14 while total \( \varphi_q \) in (b) is equal to 5. This is caused by the unbalanced reuse of some channels in topology (a) -namely channel 2 and channel 3.

![Fig. 3](attachment:image.png)

The second model is therefore defined as minimizing both the following two objectives:

\[
\begin{align*}
\min & \sum (c_{ij}t_j + p_jg_j) \quad (1) \\
\min & \sum_{q \in C} \varphi_q \quad (2.b)
\end{align*}
\]

subject to the same set of constraints as the one defined in the first model but without constraint 6.

Using this new metric (BCR) together with constraint (12) prevents local imbalances in channel allocation. While BCR metric tries to balance the number of occurrences of channels over network links (global balance in channel allocation), constraint (12) eliminates the possibility to have the same channel allocated many times to the same mesh node through its radio interfaces. As a consequence, each node will have only one of its direct neighbors (one hop away) tuned to the same channel; thus BCR together with constraint (12) leads to local and global balance in allocating network channels.
2.2.3 Flow-Capacity Model

In this model, the throughput function objective is modeled as maximizing the total throughput by computing the overall flow-capacity ratio (also called link utilization) of the network.

The objectives of the flow-capacity model are given below. The model is subject to the same set of constraints as defined for the Load-Balanced Model.

\[
\min \sum_j (c_j f_j + p_j g_j) \tag{1}
\]

\[
\max \sum_{j \in L} \sum_{i \in L} \sum_{g \in C} \frac{f_{ij}}{u_{ij}} \tag{2.c}
\]

In Section 4, an introspective comparative experimental study is conducted on these three instance models.

3 The Solution Approach

Our WMN planning optimization is essentially the maximization of the network throughput (depending on which perspective is used) while at the same time ensuring the minimization of the total deployment cost. This is achieved by selecting a minimum number of routers/gateways and adequately choosing their positions so that the network connectivity is ensured while providing full coverage to all mesh clients. It is proven that a WMN planning optimization problem is NP hard [20]. Its difficulty lies on the fact that it tries to optimize the conflicting objectives (cost and throughput) simultaneously while addressing all the constraints.

As stated earlier, solving a Multi-Objective Problem (MOP) returns a set of Pareto-optimal solutions. Each Pareto solution represents a different trade-off between the objectives that is said to be “non-dominated”, since it is not possible to improve one criterion without worsening another.

3.1 Solving a Multi Objective Problem Using Evolutionary Algorithm

3.1.1 MOP Concepts and definitions.

In the last two decades, there have been growing interests in the field of multi-objective optimization to solve real-world problems. Good introduction to this field of research can be found in [22], [39].

Without loss of generality, we assume that the various objectives are to be minimized. Then, the optimization of a MOP can be formulated as:

\[
\text{minimize } y = f(x) = [f_1(x), f_2(x), \ldots, f_N(x)]
\]

where \( x = [x_1, x_2, \ldots, x_D] \in \text{decision space} \)

and \( y = [y_1, y_2, \ldots, y_N] \in \text{objective space} \).
For a constrained problem, the decision variables $x$ are subject to a set of constraints. Every decision variable vector $x$ in the decision space is evaluated through the objective functions. The objective values are then represented as points in the objective value space.

**Definition 1 (Pareto Dominance):** For two decision vectors $a$ and $b$, $a$ is said to dominate $b$ or $a \succ b$ if and only if: $\forall i \in \{1, \ldots, N\}, f_i(a) \leq f_i(b) \land \exists i \in \{1, \ldots, N\}, f_i(a) < f_i(b)$.

**Definition 2 (Pareto Optimality):** A decision vector $a$ is said to be Pareto Optimal if and only if $a$ is non-dominated. Formally, $\forall b, b \nprec a$.

**Definition 3 (Pareto Front):** The Pareto Front is a set of all Pareto Optimal solutions (non-dominated solutions) in the objective value space.

Illustration in Fig.4 shows that points $B$, $E$, $F$, and $H$ are non-dominated as they do not lie in any of the first quadrant of the other points. Point $D$ is dominated by points of $C$, $E$, $F$, and $G$.

![Fig.4. Pareto Dominance and Pareto Front for a 2-functions objective space.](image)

### 3.1.2 Evolutionary and Swarm Optimization Algorithms

There are many nature inspired optimization algorithms that have been very successful in solving complex optimization engineering problems. The noteworthy are Genetic Algorithms (GAs) [23], Particle Swarm Optimizers (PSO) [24], Ant Colony (ACO) [25], Simulated Annealing (SA) [26], and Tabu Search (TS) [27] to name a few.

On the one hand, GAs is a meta-heuristic search technique that is adequate in exploring huge and complex search spaces. It starts off with a set (first generation) of acceptable solutions and goes on mixing their structures (crossover) and/or mutating them through many cycles (generations) until suitable solutions are found. GAs is particularly strong in dealing with discrete search spaces in that...
they are better explorers. They are also found particularly suited for diverse network optimization problems [28,29,30,31,32].

On the other hand, Particle Swarm Optimization (PSO) is an optimization technique based on an evolutionary approach introduced by Kennedy and Eberhart [24]. It models the dynamic movement or behavior of the particles in a search space. By sharing information across the environment over generations, the search process is accelerated and is more likely to visit potential optimal or near-optimal solutions. PSO do not perform well for discrete search spaces, however it gets better results in a faster and cheaper way. Moreover, it is easy to implement with simple concepts and requires few parameters to adjust.

PSO has been extended to cope with an MOP which mainly consists of determining a local best and global best Potential Solutions (PSs) of a particle in order to obtain a front of optimal solutions. There are some efficient and well-known multi-objective techniques based on PSO algorithms, of which MOPSO [33] seems to be the most effective.

We devise a kind of a hybrid optimizer that borrows the mutation operator from GAs (to better explore the search space) and uses the velocity calculation, from PSO, to guide the search towards local and global (sub) optimums. More precisely, our WMN optimization algorithm (called VMOPSO) is a modified version of MOPSO equipped with the Crowding-Distance (CD) technique of NSGA-II [22] and uses a mutation procedure. The crowding distance value of a solution, as thoroughly studied in [22] and [34], is the average distance of its two neighboring solutions. The boundary solutions with the lowest or the highest objective function value are given an infinite crowding distance values so that they are always selected. This process is done for each objective. The final crowding distance value of a solution is computed by adding the entire individual crowding distance values in each objective value.

Both the mutation procedure and the CD technique strive to enhance the exploration process, though at different levels. The CD technique is applied on the archive, where the final set of solutions would be diverse. The mutation procedure, however, operates at generation level, where the algorithm will have (enough) frequent discrete jumps to allow for escaping the traps of the local-optima issue. We also add a constraint handling mechanism for solving constraints optimization problem, such as WMN design problem. In the following, we provide more details on how the multi-objective models are solved using VMOPSO.

3.2 Logical and Physical Modeling of a planning solution

This section describes how our WMN planning solutions are logically and physically modeled.
3.2.1 A Grid Topology for a Network Deployment Scheme

The first issue to address is what topology to adopt when constructing a network of mesh node to properly handle users TSs demands.

Robinson and Knightly [14] conducted a performance study of deployment factors and concluded the benefits of adopting grid topologies over other topologies. In the same context Li et al. [15] studied the gateway placement for throughput optimization in WMNs using a grid-based deployment scheme. Their method of placing exactly \( k \) gateways has achieved better throughput in the grid scheme than in random schemes.

Based on these findings, we adopt a square grid layout as the physical representation of our WMN planning. Each grid cell corner is a CL where a mesh node can be installed. If a mesh node is installed at a given CL, it may establish a wireless communication with its eight direct-neighbors. This assumption will increase the chances of selecting a candidate neighbor among the eight with which a wireless link will be set up in the channel assignment procedure.

3.2.2 A Particle in the Swarm: Modeling a WMN planning Solution.

In PSO, a particle (a position in the search space) represents a set of assignments that is a solution to the problem. In our case, a particle is a complex data structure that provides information about user connectivity \( (x_i) \), device installation \( (t_i) \) and \( (z'_j) \), devices connectivity \( (v''_{ij}) \), gateway existence \( (g_j) \), link flows \( (f''_{jl}) \), and gateway/backbone link flows \( (F_j) \). Fig 5.a depicts different components of a particle data structure. The building blocks of a particle structure are Positions, Links, Flows and Demands. The block Positions is the most important one, as it provides information about user connectivity and the type of devices, as well as their locations and installation. The mesh nodes component contains the locations of APs (represented by IZ vector), the locations of MGs (represented by GW vector) and the list of channels assigned to radio interfaces of every mesh node installed (MR included). Fig.5.b illustrates an example of the mesh nodes component of a particle.

3.3 The VMOPSO Algorithm

Given a set of TSs scattered in a geographical region, the idea is to construct a network of mesh nodes (APs, MRs, MGs) that will best service the users TSs with minimum cost and under the given constraints. The VMOPSO algorithm needs to breed a swarm (collection) of acceptable potential planning solutions, i.e. satisfying all the constraints defined in Section 2.2.1. In this Section, we show first how the initial swarm composed of feasible solutions it built, then we describe the VMOPSO algorithm to show how a new swarm of acceptable potential planning solutions is bred.
3.3.1 Building the initial set of feasible solutions

In continuous optimization problems, getting the initial position and velocity is more straightforward because a simple random initialization is used. However, since the problem of planning a WMN is a constrained optimization problem, the initial positions must represent feasible solutions, and thus, need to be designed carefully.

Constructing an initial set of feasible solutions that satisfy the constraints (3) to (15) represents the most challenging part in our optimization process. Building such an initial set requires three main design stages, namely coverage insurance, connectivity augmentation and gateway assignment.

Coverage insurance: Recall that a TS_i can be covered by one or many CLs. This stage handles the assignment of each TS_i to one and only one CL_i. We start by selecting randomly a CL_i from the set of CLs that cover that TS_i (Fig 6.a). An AP (Access Point) is then installed at this location CL_i only if it has not yet been selected (see Fig 6.b). By applying the same procedure to all TSs, we obtain a set S_1 of APs location that provides full coverage to all TSs. More formally, S_1= { j ∈ L, CL_j covers TS_i, i ∈ I}.

At this stage, constraints (3) and (4) are satisfied and the initial set contains vertices of a disconnected graph as shown in Fig.6.b.

Connectivity augmentation: Once the coverage is done, there is no guarantee that the graph is connected. Therefore there is a need to augment the set S_1 by adding new MRs (Mesh Routers) to
connect the APs together. We apply a neighborhood based selection algorithm to find the next node to be inserted. The augmentation algorithm consists mainly, on choosing the closest neighbor in one component graph to any node of a different component. Then, the path between the two nodes is augmented. The algorithm stops when all nodes belong to the same graph component (see Fig. 6.c).

**Gateway assignment**: is based on a random selection from the set of nodes that are eligible to be gateways. However, this last design stage (gateway assignment) could be a subject of further investigation to improve network performance without changing the generic model.

For computational purposes, we use a symmetric adjacency matrix to represent the connectivity graph. We apply the fixed channel assignment algorithm described by Das et al. [35] and we implement Edmonds-Karp’s max flow algorithm [36] to assign a value on each link \( y_{ij} \) using channel \( q \) to route a flow. All remaining constraints (i.e., 5-14) are then satisfied.

![Fig. 6: A Particle position example: (a) TSs locations, (b) TSs assigned to CLs (c) S1 augmented, MGs selected.](image)

A feasible solution must satisfy all hard and soft constraints. However, those solutions that violate only the soft constraint (5) can be included in the population if space allows. This increases the likelihood of a non-feasible solution to mutate and provide a feasible one in later generations.

### 3.3.2 Breeding Potential Planning Solutions

The very first step in VMOPSO Algorithm (Algorithm 1) is to initialize the positions, as described above, initialize the boundary limits and the velocities of each solution \( i \) (particle) in \( S_w \). At this step, only feasible solutions are considered.

Each of these particles would then go through an evaluation process, i.e., an assessment of the quality of the solution, which is nothing but the evaluation of the two objective functions.

During the exploration of the search space, each particle has access to two pieces of information: the best Potential Solution (PS) that it had encountered \( (pBest) \) and the best PS encountered by its neighbors \( (gBest) \). This information is used to direct the search by computing velocities (see Algorithm 2):
velocity[i] = iw * velocity[i] + r1 * (pBest[i] – position[i]) + r2 * (Archive[gBest] – position[i])

Where \( r_1 \) and \( r_2 \) are random numbers in the range of [0,1] and \( iw \) is the inertia weight. A large inertia value will cause the particles to explore more of the search space, while a small one directs the particles to a more refined region.

The Archive is then updated by inserting into it all the currently non-dominated (fittest) solutions. This insertion process ends up in removing dominated solutions. In the case where the archive is full and there are still non-dominated solutions to be inserted, priority is then given to those particles that would ultimately enhance the diversity of the archive set, which is achieved by using the crowding distance technique (see section 3.1.2). When the decision variable exceeds its boundaries, it takes the value of its corresponding boundary and the velocity is changed to the opposite direction.

---

Algorithm 1: VMOPSO Main Algorithm

Input  
Sw: swarm, gMut: Generational Mutation factor, MaxGeneration

Output  
 Archive: External repository

Step 1:
1. Initialize the swarm Sw
   For each particle \( i \) in Sw  
   //Build feasible solutions that satisfy all constraints,
   a. Initialize feasible position,  
      // the three main steps shown in Section 3.3.1
   b. Specify lowerBound, and upperBound,  
      //boundary limits
   c. Initialize velocity  
      // initially set to Zero
   d. Set the global best guide gBest to pBest
   e. Set the personal best guide pBest to that position
End For
2. Initialize the iteration counter \( t=0 \)
3. Evaluate all particles in Sw  
   //compute objective functions \( f1 \) and \( f2 \)
4. Filter non-dominated solutions from Sw and Store them into Archive.

Step 2: Repeat
1. Process the Archive.
   a. Sort the Archive in a descending order of one of the objective functions \( f1 \) or \( f2 \).
   b. Compute the crowding distance (CD) values for each \( j \in \text{Archive} \).
   c. Sort the Archive in a descending order of CD values.
2. Set \( gBest[i] \) to the randomly selected particle from the top 10% of the sorted Archive.
3. ConstructWMNPlanningSolution  
   // invoke Algorithm 2.
4. Check for constraints satisfaction
5. Update the Archive:  
   // insert non-dominated and feasible particles in the Archive.
   a. If any particle \( k \) in Sw dominates any particle \( l \) in Archive then:
      Delete \( l \) from Archive and insert \( k \) in Archive.
   b. If Archive is full and there is non-dominated particle (candidate) in Sw then
      • Compute the crowding distance values for each \( j \in \text{Archive} \)
      • Select the victim: a Random particle in the bottom 10% of the CD-sorted Archive (most
crowded portion).

- Replace it with the new candidate.

End If

6. Update pBest
7. Increment t
Until (t>= MaxGeneration)

For each particle in the swarm, the iterative algorithm (Algorithm 2) consists of constructing a subset $S_1$ of APs locations to cover all TSs, mutating it, placing gateways and then assigning flows and channels. The most important phase is the repetitive task of constructing the set $S_1$ and then mutating it over and over until it satisfies at least all hard constraints. Then $S_1$ is augmented to ensure the connectivity constraints.

After this solution-construction process, the velocities, the positions and the fitnesses (values of the two objective functions) of the particles are computed. Then some of these particles are inserted into the archive provided that they dominate or at least are non-dominated by the previously “archived” non-dominated solutions.

Algorithm 2: ConstructWMNPlanningSolution

Input $Sw$: Swarm, t: generation counter, MaxGeneration,
gMut: Generational mutation factor,
sMut: Swarm mutation factor //adopted from MOPSO [33],
// $sMut = (1-t/MaxGeneration*gMut)^{3/2}$

Output $Sw$: Swarm

mutateEnabled:= true;
If (t>= MaxGeneration*gMut) then mutateEnabled :=false;
for each particle $i$ in Sw.

if (mutateEnabled) and (sMut=1) //Mutation at early generations, i.e. MaxGeneration*gMut
    $S_1$: Mutate($S_1$) // $S_1$ is the subset of APs locations
endif

$S := Augment(S_1)$; // Connectivity augmentation

$Y_1:=Construct_connectivity_matrix(S)$ ;
$Y_2:= Assign_channels(Y_1)$ ;
$G := PlaceGateways(Y_2)$; //Gateway assignment

Compute_flows(G) ;

Construct_New_Particle(); // with the newly generated $S,Y_2,G$ and flows
Compute_Velocities(); // As described in the beginning of this subsection
Update_Positions(); // New position= current position + computed velocity
Check particle boundaries, if violated change particle search direction (i.e., velocity(i)* -1)

Evaluate_Particles(); // Compute objective functions f1 and f2

endfor
A position in the search space is a solution to our planning problem; however, the values, returned by Update_Positions() procedure in Algorithm 2 are not guaranteed to be integers (0 or 1). For this purpose, we add a final process that we call particle filtering to allow only particles with a considerable progress to change to 0 (respectively 1). If the difference between the two positions (initial and the updated one) of a particle goes beyond a given threshold $\alpha$ (based on experiments, $\alpha$ is set to 0.3), then the final position will be the reverse of the initial one (i.e., 0 if it was 1 and vice-versa); otherwise, the new position is discarded. i.e., the particle remains in its original position for further improvement.

4 **NUMERICAL EXPERIMENTS, SIMULATION AND ANALYSIS**

In this section, we use the algorithms we devised and described in Section 3 to solve the three model problems proposed in Section 2.2 - (a) Load-Balanced Model, (b) Interference Model, (c) Flow-Capacity Model. The obtained results are presented and discussed.

4.1 **Experiments Setup**

Our numerical analysis setup is based on WMN key parameters which are as follows: the number of clients (TSs) $n$, the number of CL $m$, the client demands $d_i$, the gateway factor cost $p_j$, and the number of radio interfaces $R$. In this regard, we define the Standard Setting (SS) of the WMN key parameters as the following:

$$SS=\{(n:150), (m:49), (d_i:2\text{Mb/s}), (v_j:54\text{Mb/s}), (u_j:54\text{Mb/s}), (M:128\text{Mb/s}), (e_j:200), (p_j:8*e_j), (R:3), (k:11)\}.$$  

The positions of the $n$ TSs are randomly generated for the first run and kept fixed for the remaining runs. A run of our algorithm involves 100 generations each with a population size and an archive size of 50 and 20 particles respectively. Finally, we set the mutation rate to 50% ($gMut=0.5$), which has proven to lead to the best Pareto front [37]. The algorithm is coded in the Java programming language and all the experiments were carried out on a Pentium M 1.5GHz machine.

We study the performance of our algorithm over grid graphs and under many deployment scenarios. For practical reasons, the throughput objective of the Flow-Capacity Model is rewritten as a minimization of the inverse of the overall network flow-capacity aggregate. Obviously, for the sake of a consistent results’ interpretation, the same initial configurations (clients’ distribution) are saved and used for the three models. Lastly, for each execution scenario (key parameter variation study), results are reported on 10 runs thus requiring additional filtering process to maintain the non-dominance aspect amongst the resulted Pareto fronts.

4.2 **Measuring the Performance**

Diversity and convergence of the returned set of solutions are the two main characteristics of any multi-objective optimization problem solver. In this subsection we introduce the most widely used metrics of diversity and convergence to compare the contending fronts.
4.2.1 Diversity Measure

First, we use the Schott Spacing Metric [38] to measure the range variance of neighboring vectors in the Archive (the Pareto Front, PF). It is defined as:

\[ S = \sqrt{\frac{1}{n(n-1)} \sum_{i=1}^{n} (d - d_i)^2} \]

Where \( n \) is the number of solutions in the PF;

\[ d_i = \min_j \left( |f_i^j - f_{i+1,2}^j| + 1 |f_i^j - f_{2,2}^j| \right) \]

\( d \) : The mean of all \( d_i \).

A zero value for this metric means that all solutions of the PF are equidistantly spaced, however, in this study we are not interested in how close the metric value is to zero, but on the values of this metric returned by each PF. The lower the value returned by a PF, the better that PF is.

4.2.2 Convergence Measure

The coverage and the hyper-area metrics proposed in by Zitzler [39] are powerful metrics that assess the convergence of a given front. The quasi-totality of works in the domain of performance assessment in the realm of MOEA uses at least one of these two metrics.

The convergence metric, \( \gamma \), proposed by Deb in [22] assumes the presence of the true Pareto-optimal front. A sample set called \( H \) of \( N \) equidistantly-spaced solutions from the true Pareto-optimal front is extracted. Then each point obtained with an algorithm, a minimum Euclidean distance from it to the set \( H \) is computed. The averages of these values constitute \( \gamma \).

This metric is misleading and therefore is not used in the experiments. In Fig.7, the front \( F_2 \) (a singleton for the sake of simplicity) is deemed to be better than the front \( F_1 \) –as it is closer to the true Pareto front than \( F_1 \). However, \( F_2 \) is strictly dominated by \( F_1 \).

![Fig.7. Anomaly of the Convergence metric \( \gamma \).](image)

Authors in [40] proposed a new quality indicator that measures the minimum improvement a vector solution (a point) in the objective search space has to undertake in order to reach the non-dominance status. For this purpose, a set of Potential Pareto Regions (SPPRs) is constructed. Each PPR in the SPPRs is a region within which non-dominance status is verified. For each point outside the SPPRs, we can measure its Expected Improvement (EI) as the length of the segment of the line.
originating from the point and intersecting the closest PPR (see Algorithm 3 below). In other words, EI is a scalar value that a point in the objective space has to gain in order to reach the status of non-dominance. An illustrative example is depicted in Fig. 8. For more details on this technique and the algorithms used to compute the EIs, we refer the reader to the lecture notes in [40].

![Fig. 8. Potential Pareto Region (PPR) and the Expected Improvement (EI).](image)

In our experimentations, only inertia ($iw$) value setup and radio ($R$) variation experiments involve contending comparable fronts. To decide on which value of $iw$ to choose and which radio variation is optimal, we use the Spacing Metric, the *Expected Improvement* EI metric and some coverage related metrics, such as:

\[
PF_i^{\text{Pareto}} = PF_i \cap CPF.
\]

- $|PF_i|$: The size of the set of non-dominated vector solutions returned,
- $|CPF|$: The size of the CPF filtered, CPF-size
- $|PF_i|^{\text{Pareto}}$: The number of vector solutions inside the CPF, $(PF_i \cap CPF)$
- $|PF_i|$-$CPF$: The number of vector solutions outside the CPF,
- $\%$fromCPF : The percentage of each $PF_i$ covering the CPF: $\frac{|PF_i^{\text{Pareto}}|}{|CPF|}$,
- $\%$fromPF: The percentage of each $PF_i$ covering the front returned by $PF_i$ in question: $\frac{|PF_i^{\text{Pareto}}|}{|PF_i|}$.

It must be noted that some of the above metrics are redundant. However, they are reported for a better contrast.
4.3 Plotting and Graphs Interpretation

For each model, the planning solutions (deployment cost against performance) for a given value of each key parameter variation constitute a (Pareto) front of non-dominated solutions that is plotted in a (objective space) graph. On the other hand, only the cheapest solution is considered for plotting the resource utilization graph. For that, we plot the number of MRs, the number of APs, the number of MGs, and finally the number of links to show the network connectivity level.

In this subsection we compare the characteristics of the solutions, which can prove very important in decision making. These characteristics are the number of the solutions, the width of the spectrum of the solutions, and the uniform-distribution of the solutions. In addition, for each scenario we further plot the device utilization graphs in terms of the number of gateways, routers, and links. This makes the comparisons between the three models possible.

4.4 Results and Analysis

4.4.1 Setting the Inertia value

A large inertia value will cause the particles to explore more of the search space, while a small one directs the particles to a more refined region. The importance of inertia weight was pointed out by Shi and Eberhart [41] who reported that 0.4 is the best value. However, for a different type of problem, such as WMN design problem, a different value of $i_w$ may lead to better exploration and exploitation of a search space. To set an appropriate value of $i_w$ for our numerical experiments, different runs are carried out for the same model (Load-balanced Model for instance), by changing only the inertia weight $i_w$, while maintaining the same SS as defined in Section 4.1. In Fig.9, the Pareto fronts of ten runs, for different value of $i_w$ (0.2, 0.4, 0.6, 0.8) are plotted.

It is clear from Fig.9, that the solutions in Pareto front corresponding to $i_w=0.8$ are almost all dominated by the solutions of the other fronts (i.e., $i_w=0.2, 0.4, 0.6$).

Referring to TABLE II, The size of the non-dominated set (IPF) for $i_w=0.2$ is not as bigger as the size of the non-dominated set when $i_w=0.6$, but it is taking around of 55% of the CPF (a merger of the 3 contending fronts) and 66% of the front are non-dominated. The diversity of the solutions is also much better when $i_w=0.2$. Based on these results, we set $i_w=0.2$ for all remaining numerical experiments.
Fig.9. Pareto fronts of planning solutions for different value of inertia weight, \( iw \).

TABLE II: Test results of Load-balanced Model for different value of inertia weight.

<table>
<thead>
<tr>
<th>Inertia weight</th>
<th>( iw=0.2 )</th>
<th>( iw=0.4 )</th>
<th>( iw=0.6 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spacing S</td>
<td>0.0526</td>
<td>0.1014</td>
<td>0.153</td>
</tr>
<tr>
<td>(</td>
<td>PF_i</td>
<td>)</td>
<td>15</td>
</tr>
<tr>
<td>(</td>
<td>CPF</td>
<td>)</td>
<td>18</td>
</tr>
<tr>
<td>(</td>
<td>PF_i</td>
<td>_{Pareto})</td>
<td>10</td>
</tr>
<tr>
<td>(</td>
<td>PF_i-CPF</td>
<td>)</td>
<td>5</td>
</tr>
<tr>
<td>% fromCPF</td>
<td>0.5556</td>
<td>0.1667</td>
<td>0.2778</td>
</tr>
<tr>
<td>% fromPF</td>
<td>0.6667</td>
<td>0.2</td>
<td>0.2942</td>
</tr>
</tbody>
</table>

4.4.2 Radio Interfaces (\( R \)) Variation

The number of radio interfaces \( R \) is varied from 2 to 5, and each of these radio interfaces is equipped with 11 channels. In this experiment, to better test the channels we do not use the standard setting SS as defined in section 4.1, but choose a heavily loaded and condensed network \( (m=100, n=150, d=5\text{Mb/s}) \) instead. The Pareto fronts illustrated in Fig.10 show that Load-Balanced model provides better, diverse and greater number of solutions than the other two models. In Table III, we choose to perform head-to-head comparisons for a better analysis. For each model, we compare the results of instance models of 2-radios with 3-radios, 4-radios with 3-radios, 5-radios with 3-radios, and finally 5-radios with 4-radios. For all the three models, the 3-radios decision choice always outperforms the 2-radios one, while the 4-radios and 5-radios always outperform the 3-radios instances.

The Load-balanced model gives the larger sets of solutions \( (|PF_i|) \) while Link utilization shows a better diversity. In Load-balanced and Interference models, choosing 4 radios seems the best decision choice since the fronts related to 4-radios, when compared to the 5-radios design choice, make up 62% to 71% of the CPF.

In Link Utilization model, the solutions of the 5-radios decision choice are all in the CPF, though not well-spaced \( (S=0.6959) \) and as can also be seen in Fig.10.c. The 4-radios decision choice, however, end up with a better diverse set of solutions. It must also be noted that all of its solutions that are outside the CPF (see line \( |PF_i-CPF| \)) actually resides on the limit of the non-dominance status \( (E_l=0) \).
The more radio interfaces are deployed the more links are established, and the less gateways are needed. This remains true when $R$ shifts from 2 to 3 and from 3 to 4 (Fig. 10). The Load-Balanced Model profits from a radio gain by decreasing the number of APs, relays, and gateways and increasing the number of links. The Interference Model does follow the same pattern as the Load Balanced Model in reducing the number of resources.

On the other hand, all models show a disruption when the number of radios goes from 4 to 5 (see Fig.11.a and b). This may be caused by the high level of interferences related to the increase in network links.

We can then stipulate that no gain can be obtained if we deploy more than four radio interfaces, which stands true for all three models, under the same SS.

### 4.4.3 Grid Size (m) Variation

The number of candidate locations $m$ is gradually increased while all other parameters are maintained fixed. Results (Fig.12 and Fig.13) show that there is consent in all models that a 7x7-grid is the best in satisfying the Standard Setting SS. A very important remark is that the cardinality and the width of the spectrum of the planning solutions are greater in the Load-Balanced Model than they are in the other two models. This fact makes the first model the best to find cheaper planning solutions. On the other hand, Interference-based model seems to generate more and well spaced solutions than the third model does. These observations, drawn from Fig.12, suggest that a network planner with ‘flexible’ requirements would possibly opt for Load-Balanced Model as it offers more and better diverse planning solutions.

When we turn our attention to resource utilization, it is clear from Fig.13 that the Load-Balanced Model dominates the flow-capacity model since it uses less network mesh nodes in all types of grids. On the other hand, the interference based model requires fewer routers but more gateways for grids larger than 8x8. One can observe that the Load-Balanced Model is more careful in using the gateways (MG) (Fig.13.b) as it rather adds more routers (MR) (Fig.13.a) and precisely more access points (AP) as shown in Fig.13.b. Notice that in all models, a higher number of candidate locations leads to an increase in the number of routers and gateways even for the same number of users. The first reason is the fact that increasing the number of CLs increases the probability of a TS (Traffic Spot for a client) not being connected to an AP through a multi hop wireless path, which leads to installing more nodes. A larger size of grid can improve the network performance as more flexibility in choosing routing paths is possible, and consequently, the probability to have traffic contention and bottlenecks is very low, but also increases the total deployment cost, which is highly affected by the number of gateways deployed. Therefore, in practice, the network planner has to decide on the appropriate grid size that satisfy both cost and performance requirements.
TABLE III: Radio Setup results for a heavy condensed network. 

\((m=100, \ n=150, \ d=5\text{Mb/s})\)

<table>
<thead>
<tr>
<th>interference mode</th>
<th>#Radios</th>
<th>3</th>
<th>2</th>
<th>4</th>
<th>3</th>
<th>5</th>
<th>3</th>
<th>5</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PF</td>
<td>7</td>
<td>7</td>
<td>6</td>
<td>7</td>
<td>5</td>
<td>7</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>CPF</td>
<td>7</td>
<td>7</td>
<td>6</td>
<td>7</td>
<td>5</td>
<td>7</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>PFi</td>
<td>7</td>
<td>0</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>CPF</td>
<td>0</td>
<td>7</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>%fromCPF</td>
<td>100.0</td>
<td>0.0</td>
<td>57.14</td>
<td>42.86</td>
<td>66.67</td>
<td>33.33</td>
<td>28.57</td>
<td>71.43</td>
</tr>
<tr>
<td></td>
<td>%fromPF</td>
<td>100.0</td>
<td>0.0</td>
<td>66.7</td>
<td>42.9</td>
<td>80.0</td>
<td>28.6</td>
<td>40.0</td>
<td>71.4</td>
</tr>
<tr>
<td></td>
<td>Avg El</td>
<td>0</td>
<td>860.72</td>
<td>0</td>
<td>0.75</td>
<td>0</td>
<td>40.4</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td></td>
<td>stdv El</td>
<td>0</td>
<td>1252.11</td>
<td>0</td>
<td>0.83</td>
<td>0</td>
<td>79.81</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td></td>
<td>Spacing</td>
<td>0.4842</td>
<td>0.7970</td>
<td>0.5145</td>
<td>0.4842</td>
<td>1.7979</td>
<td>0.4842</td>
<td>1.7979</td>
<td>0.5145</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>load balanced Model</th>
<th>#Radios</th>
<th>3</th>
<th>2</th>
<th>4</th>
<th>3</th>
<th>5</th>
<th>3</th>
<th>5</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PF</td>
<td>18</td>
<td>13</td>
<td>21</td>
<td>19</td>
<td>20</td>
<td>19</td>
<td>20</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>CPF</td>
<td>20</td>
<td>21</td>
<td>20</td>
<td>20</td>
<td>21</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PFi</td>
<td>18</td>
<td>2</td>
<td>13</td>
<td>8</td>
<td>16</td>
<td>4</td>
<td>8</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>CPF</td>
<td>0</td>
<td>11</td>
<td>8</td>
<td>11</td>
<td>4</td>
<td>15</td>
<td>12</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>%fromCPF</td>
<td>90.0</td>
<td>10.0</td>
<td>61.9</td>
<td>38.1</td>
<td>80.0</td>
<td>20.0</td>
<td>38.1</td>
<td>61.9</td>
</tr>
<tr>
<td></td>
<td>%fromPF</td>
<td>100.0</td>
<td>15.4</td>
<td>61.9</td>
<td>42.1</td>
<td>80.0</td>
<td>21.1</td>
<td>40.0</td>
<td>59.1</td>
</tr>
<tr>
<td></td>
<td>Avg El</td>
<td>0</td>
<td>1727.28</td>
<td>0</td>
<td>127.28</td>
<td>0</td>
<td>880.0</td>
<td>0</td>
<td>2822.2</td>
</tr>
<tr>
<td></td>
<td>stdv El</td>
<td>0</td>
<td>2162.94</td>
<td>0</td>
<td>213.58</td>
<td>0</td>
<td>1726.34</td>
<td>0</td>
<td>3900.4</td>
</tr>
<tr>
<td></td>
<td>Spacing</td>
<td>0.8143</td>
<td>0.5811</td>
<td>0.5835</td>
<td>0.8143</td>
<td>0.4035</td>
<td>0.8143</td>
<td>0.4035</td>
<td>0.5835</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>link utilization Model</th>
<th>#Radios</th>
<th>3</th>
<th>2</th>
<th>4</th>
<th>3</th>
<th>5</th>
<th>3</th>
<th>5</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PF</td>
<td>7</td>
<td>7</td>
<td>8</td>
<td>7</td>
<td>10</td>
<td>7</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>CPF</td>
<td>8</td>
<td>9</td>
<td>11</td>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PFi</td>
<td>7</td>
<td>1</td>
<td>7</td>
<td>2</td>
<td>10</td>
<td>1</td>
<td>10</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>CPF</td>
<td>0</td>
<td>6</td>
<td>1</td>
<td>5</td>
<td>0</td>
<td>6</td>
<td>0</td>
<td>3</td>
</tr>
</tbody>
</table>

Fig.10. Pareto Fronts of Planning Solutions For different radio interfaces: 
(a) Load-Balanced Model, (b) Interference Model, (c) Flow-Capacity Model.
<table>
<thead>
<tr>
<th></th>
<th>87.5</th>
<th>12.5</th>
<th>77.78</th>
<th>22.22</th>
<th>90.91</th>
<th>9.09</th>
<th>57.14</th>
<th>42.86</th>
</tr>
</thead>
<tbody>
<tr>
<td>%fromCPF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>%fromPF</td>
<td>100.0</td>
<td>14.3</td>
<td>87.5</td>
<td>28.6</td>
<td>100.0</td>
<td>14.3</td>
<td>100.0</td>
<td>66.7</td>
</tr>
<tr>
<td>Avg EI</td>
<td>0</td>
<td>6600</td>
<td>0</td>
<td>1520</td>
<td>0</td>
<td>1866.67</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>stdv EI</td>
<td>0</td>
<td>5938.57</td>
<td>0</td>
<td>1330.26</td>
<td>0</td>
<td>1431.39</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Spacing S</td>
<td>0.3311</td>
<td>0.5066</td>
<td>0.1431</td>
<td>0.3311</td>
<td>0.6959</td>
<td>0.3311</td>
<td>0.6959</td>
<td>0.1431</td>
</tr>
</tbody>
</table>

Fig. 11. Network Devices Utilization For Three Models with Different Radio Interfaces. (a) Number of MRs-Links, (b) Number of APs-MGs

Fig. 12. Pareto Fronts of Planning Solutions For different Grids. (a) Load-Balanced Model, (b) Interference Model, (c) Flow-Capacity Model.

Fig. 13. Network Devices Utilization For Three Models with Different Grids. (a) number of MRs, (b) number of APs and MGs

4.4.4 Demand (d) Variation

Regarding Pareto fronts for $d_i=1,2,3,4$ and 5Mb/s, the Load-Balanced Model (Fig. 14.a) returns larger set of planning solutions while the non-dominated planning solutions provided by the Interference
Model (Fig.14.b) are better stretched and evenly spaced than those of Flow-Capacity Model (Fig.14.c).

As can be seen from Fig. 15.a, when demands increase the number of gateways increases accordingly to satisfy connectivity constraints by creating new routing paths. More relays MRs than APs are added in order to connect these APs to newly added gateways. All models deploy almost the same number of gateways when demand varies, however, Interference Model seems to better handle the increase of demands, as shown in Fig. 15.b, by using less APs.

4.4.5 Traffic Spots (n) Variation

As with previous scenarios, Fig.16 shows that more and diverse planning solutions are produced by Load-Balanced Model. Load-balanced and Flow-Capacity models require few gateways, relays, and links to be added when more users are added in, compared to Interference Model which adds more of these devices (see Fig.17). On the other side, the Load-Balanced Model deploys less APs than the other models; this suggests that the Load-Balanced Model may be better in handling the scalability issue.
4.5 A Comparison with Related Work

We introduced three bi-objective models with two conflicting objectives (deployment cost and network performance) that need to be optimized concurrently while satisfying all the QoS constraints. Validating our results against other known models for WMN planning problems turns out to be impossible as it is unpractical to compare a set of Pareto (two-dimension) optimal solutions with a one-dimension optimal solution. Nevertheless, we can at least check the one common objective function (deployment cost) to see whether the results fall in the same range.

We compare our results to the closest related work results obtained by Amaldi et al [3]. We refer to their model as AML and to ours as MOCB (using Load balanced Model). They used the following parameters setup ($d_i=3$Mb/s, $n=100$, $m=50$, $R=3$ and $k=11$) and obtained a “single” planning solution per run. They reported the mean value calculated over the ten runs ($#MR=23.65$, $#MG=3.3$, $#Links=21.35$). Using the same parameters setup, we obtained 15 non-dominated planning solutions (see TABLE IV). We report the two extreme planning solutions (cheapest, most expensive) together with Amaldi et al.’s mean value solution in TABLE V.
TABLE IV: solution of MOCB (Load-Balanced Model),
\[ \text{d} = 3\text{Mb/s}, n = 100, m = 50, \text{R} = 3 \text{ and } \text{ch} = 11. \]

<table>
<thead>
<tr>
<th>MR</th>
<th>AP</th>
<th>MG</th>
<th>Links</th>
<th>Cost</th>
<th>Load balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>13</td>
<td>3</td>
<td>31</td>
<td>9400</td>
<td>0.898454</td>
</tr>
<tr>
<td>23</td>
<td>12</td>
<td>3</td>
<td>35</td>
<td>10000</td>
<td>0.811035</td>
</tr>
<tr>
<td>18</td>
<td>13</td>
<td>4</td>
<td>30</td>
<td>10800</td>
<td>0.800554</td>
</tr>
<tr>
<td>22</td>
<td>13</td>
<td>4</td>
<td>43</td>
<td>11600</td>
<td>0.796620</td>
</tr>
<tr>
<td>23</td>
<td>13</td>
<td>4</td>
<td>37</td>
<td>11800</td>
<td>0.790363</td>
</tr>
<tr>
<td>25</td>
<td>12</td>
<td>4</td>
<td>40</td>
<td>12200</td>
<td>0.770813</td>
</tr>
<tr>
<td>29</td>
<td>13</td>
<td>4</td>
<td>46</td>
<td>13000</td>
<td>0.760068</td>
</tr>
<tr>
<td>25</td>
<td>14</td>
<td>5</td>
<td>43</td>
<td>14000</td>
<td>0.728297</td>
</tr>
<tr>
<td>21</td>
<td>12</td>
<td>6</td>
<td>35</td>
<td>15000</td>
<td>0.726868</td>
</tr>
<tr>
<td>24</td>
<td>13</td>
<td>6</td>
<td>43</td>
<td>15600</td>
<td>0.713449</td>
</tr>
<tr>
<td>21</td>
<td>12</td>
<td>7</td>
<td>38</td>
<td>16800</td>
<td>0.683738</td>
</tr>
<tr>
<td>29</td>
<td>15</td>
<td>7</td>
<td>50</td>
<td>18400</td>
<td>0.675634</td>
</tr>
<tr>
<td>23</td>
<td>14</td>
<td>8</td>
<td>42</td>
<td>19000</td>
<td>0.661046</td>
</tr>
<tr>
<td>25</td>
<td>14</td>
<td>11</td>
<td>46</td>
<td>24800</td>
<td>0.648230</td>
</tr>
<tr>
<td>26</td>
<td>15</td>
<td>15</td>
<td>57</td>
<td>32200</td>
<td>0.640597</td>
</tr>
</tbody>
</table>

TABLE V: Two extreme planning solutions of MOCB versus the solution of AML

<table>
<thead>
<tr>
<th>AML</th>
<th>MR</th>
<th>MG</th>
<th>Links</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>23.65</td>
<td>3.3</td>
<td>21.35</td>
<td>10660.05</td>
</tr>
<tr>
<td>MOCB_{cheapest}</td>
<td>20</td>
<td>3</td>
<td>31</td>
<td>9400.05</td>
</tr>
<tr>
<td>MOCB_{expensive}</td>
<td>26</td>
<td>15</td>
<td>57</td>
<td>32200.05</td>
</tr>
</tbody>
</table>

Our planning solutions of MOCB are numerous and diverse, ranging from the very cheap planning solution (MOCB_{cheapest} in TABLE V) with less balanced channels’ load to the very expensive planning solution (MOCB_{expensive} in TABLE V) with well-balanced distribution of load over network channels. Results in TABLE IV show that our approach tends to provide some planning solutions which may be more expensive than that of AML, but guarantee load balanced topologies while others are cheaper. Balancing load over network channels is a desirable QoS metric that increases network performance. Indeed, it minimizes traffic contentions and bottlenecks, which increases global network throughput. The AML model is a single-objective model, which does not consider any QoS metric in the formulation. This fact led us to compare only the common objective (cost objective function) on a single objective basis. TABLE V shows that MOCB generates from 12% cheaper planning solutions to 30.2% more expensive planning solutions than the average-value solution generated by the AML model for the same parameters setting.

4.6 Performance Evaluation via simulation

The main goal of this section is to evaluate the performance of the three models using a common metric, namely the overall network throughput. The analysis on the results obtained in Section 4.4, by varying different key parameters, has shown that Load-Balanced Model always generates broader, diverse and well-dispersed set of non-dominated planning solutions. However a clear conclusion
about the performance of the planning solutions could not be drawn without comparing the throughput each model generates. To this end, we have run simulations with the discrete event network simulator OMNET++ and INETMANET framework [42] to support the MR-MC topologies.

In order to define the simulation scenario and get more insights on the true performance of the planned WMNs, we consider one solution provided by each model under a specific standard setting \( SS=\left( n; 150 \right), \left( m; 49 \right), \left( d; 1\text{Mb/s} \right), \left( u; 54\text{Mb/s} \right), \left( v; 54\text{Mb/s} \right), \left( M; 128\text{Mb/s} \right), \left( e; 200 \right), \left( p; 8*e \right), \left( R; 3\right), \left( k; 11 \right) \). The three solutions selected have the same deployment cost (same value of \( f_1 \)) and are built under the same distribution of clients TSs. Fig.18 depicts the three simulated topologies.

We feed the simulator with the positions and the types (AP, MR, MG) of mesh nodes, as well as the channel matrix and the routing matrix derived from Edmond’s approach [36]. We run the simulations with \textit{ftp} traffic on top of TCP application. Each AP is transferring a file (\textit{ftp}) and each gateway is wired (connected by cable) to a switch. Links capacity on these cables is set to 100Mbps and the wireless link capacity is set to 11Mbps. All the simulations are implementing fully the IP stack and all the routing between stations is IP routing (layer 3). Each station must negotiate each
ARP with its next hop. This introduces an additional delay at the beginning of each traffic transaction, as normally happen in the real devices. The radio transmission power is set to 100mW and all the radios in the same channel are under mutual interference. Finally we use the wireless propagation model PathLoss with alpha set to 2.

To compare the throughputs, we use the received bytes (or bits) (throughout the simulation time) by each server associated to an AP. The graph presented in Fig.19 illustrates the bits/s TCP throughput for all servers (APs) per topology. The higher slopes in the graph indicate bigger throughputs. We can see that in the Interference Model, 64% of the servers have their bits/s values oscillate above 5000bits/s, while 52% of the servers in the Load-Balanced Model and 43% of the servers in the Flow-Capacity Model have their bits/s values above 5000bits/s. This suggests that the Interference Model handles better the initial network loads (users’ demand).

We further calculate the overall TCP throughput and standard deviation (Std) of the bytes received by all the gateways of each topology. The results reported in TABLE VI show that the topology of the Load-Balanced Model performs better as it has the highest overall throughput. Moreover, it is easy to see that the traffic loads are well balanced by this approach, leading also to a fair use of gateways capacity since the gateways are equitably used (better Std value) compared to the other two models.

![Fig.19. Servers (bits/s) TCP throughput. (a): Load-Balanced Model, (b): Interference Model, (c): Flow-Capacity Model.](image)

<table>
<thead>
<tr>
<th></th>
<th>Interference Model</th>
<th>Load-Balanced Model</th>
<th>Flow-Capacity Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>MG index</td>
<td>Bytes Received/s</td>
<td>MG index</td>
<td>Bytes Received/s</td>
</tr>
<tr>
<td>MG#2</td>
<td>41373.01</td>
<td>MG#9</td>
<td>30946.56</td>
</tr>
<tr>
<td>MG#13</td>
<td>44270.08</td>
<td>MG#13</td>
<td>30292.91</td>
</tr>
<tr>
<td>MG#20</td>
<td>13009.49</td>
<td>MG#14</td>
<td>43800.69</td>
</tr>
<tr>
<td>MG#25</td>
<td>22225.49</td>
<td>MG#30</td>
<td>31813.97</td>
</tr>
<tr>
<td>MG#35</td>
<td>45850.88</td>
<td>MG#39</td>
<td>26081.71</td>
</tr>
<tr>
<td>MG#43</td>
<td>26627.84</td>
<td>MG#42</td>
<td>31375.36</td>
</tr>
<tr>
<td>Tot. TCP Throughput</td>
<td>32226.13</td>
<td>32301.87</td>
<td>30302.58</td>
</tr>
<tr>
<td>Std</td>
<td>13527.67</td>
<td>5772.93</td>
<td>11089.94</td>
</tr>
</tbody>
</table>
5 CONCLUDING REMARKS

The bulk of the contributions in solving the WMNs planning problems assume a fixed topology, are all bounded to medium size instance problems, and optimize a single objective, namely the network deployment cost.

In this paper, we have shown that the optimization of WMN planning problem is naturally multi-objective. We proposed a generic WMN planning model where the two objectives of deployment cost and network throughput are optimized simultaneously. While the deployment cost is trivial, maximizing the throughput can be achieved in three ways: either maximizing the overall flow-capacity ratio, minimizing the network interference or balancing the network links’ load. We instantiated three specific WMN planning models, namely Load-Balanced Model, Interference Model, and Flow-Capacity Model.

We conducted some numerical experiments on the three models to study the impacts of some key parameter variations on the network performance. In the light of the results shown in Section 4, and from a decision making perspective, Load-Balanced Model always generate a broader set of non-dominated solutions, favors cost-effective solutions, and guarantees a diverse and well dispersed set of solutions than the other two models. This makes this model the better of the three to generate cheaper planning solutions. There is also a tendency in using the costly gateways with care in a sense that it usually adds more links and routers than deploying expensive gateways. On the other side, Interference Model handles better the increase of demands and no more gain can be obtained with more than four radio interfaces.

Finally, the actual performance of our three planning bi-objective models is assessed by a component-based network simulator to derive the actual network throughput. The results clearly show that the Load-balanced Model provides better overall throughput. Equally important, gateways receive better balanced shares of the network traffic due to the inherent design approach which strives to balance the loads all over the network links.

REFERENCES


