**Les Cahiers du GERAD** ISSN: 0711–2440

# Tactical wireless network design with multi-beam antennas

V. Perreault, A. Hertz, A. Lodi

G–2023–34

August 2023



**Citation suggérée :** V. Perreault, A. Hertz, A. Lodi (Août 2023). Tactical wireless network design with multi-beam antennas, Rapport technique, Les Cahiers du GERAD G– 2023–34, GERAD, HEC Montréal, Canada.

**Avant de citer ce rapport technique,** veuillez visiter notre site Web (https://www.gerad.ca/fr/papers/G-2023-34) afin de mettre à jour vos données de référence, s'il a été publié dans une revue scientifique.

**Suggested citation:** V. Perreault, A. Hertz, A. Lodi (August 2023). Tactical wireless network design with multi-beam antennas, Technical report, Les Cahiers du GERAD G–2023–34, GERAD, HEC Montréal, Canada.

**Before citing this technical report,** please visit our website (https://www.gerad.ca/en/papers/G-2023-34) to update your reference data, if it has been published in a scientific journal.

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2023
– Bibliothèque et Archives Canada, 2023

Legal deposit – Bibliothèque et Archives nationales du Québec, 2023
– Library and Archives Canada, 2023

**GERAD** HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

**Tél. : 514 340-6053**
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

# Tactical wireless network design with multi-beam antennas

**Vincent Perreault** [a, b]

**Alain Hertz** [a, b]

**Andrea Lodi** [a, b, c]

[a] *MAGI, Polytechnique Montréal, Montréal (Qc), Canada, H3T 1J4*

[b] *GERAD, Montréal (Qc), Canada, H3T 1J4*

[c] *Jacobs Technion-Cornell Institute, Cornell Tech and Technion - IIT, New York, NY 10044, USA*

alain.hertz@gerad.ca

**August 2023**
**Les Cahiers du GERAD**
**G–2023–34**

**Abstract :** Tactical wireless networks are used in cases where standard telecommunication networks are unavailable or unusable, e.g. disaster relief operations. We fully model the design of these tactical networks as a nested optimization problem with a physically-modeled signal and three elementary data traffic scenarios. Specifically, we consider the problem for tree networks with two channels, four possible frequencies and multi-beam antennas of 24 beams. We propose a multi-level algorithm that uses 1) a Tabu Beam Search for the topology design, 2) a simple geometrical heuristic for the antenna configuration and 3) exact methods, heuristics, meta-heuristics and bounding procedures for the network configuration. Synthetic experiments suggest that our method finds very good networks and that it significantly outperforms a previous algorithm.

**Keywords :** telecommunication, network design, optimization techniques

**Acknowledgements:** We gratefully acknowledge financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) through the grants DCPJ–538896–19 and RGPIN–05688–2017, as well as the Canada Graduate Scholarship for master's program. Special thanks go to Ghassan Dahman for guiding us through the intricacies of the telecommunication aspects of the problem.

# 1 Introduction

Wireless communication is foundational to all essential technology today. In cases where standard telecommunication networks are unavailable or unusable, temporary tactical networks must be set up to support the necessary technological needs. For instance, such tactical networks are crucial for disaster relief operations and military operations in foreign territory.

More specifically, we are interested in designing a wireless network that connects multiple key coordinates of a certain region, typically between 10 and 50. These nodes are connected such that data traffic can be streamed between all pairs of nodes. To this end, each node is equipped with a radio and two antennas, which can be omni-directional (i.e. with the same signal strength in all directions), single-beam (i.e. with a beam of signal centered on a specified direction along which the signal is strongest) or multi-beam (i.e. have a circular grid of individual beams, in our case 24) [1]. The scope of this work is restricted to multi-beam antennas which provide the most difficult version of this problem.

The radios have two possible channels (one for each antenna) on frequency bands 3+ and 4 respectively. We consider two possible signal frequencies per channel (3+: 2000 MHz and 2400 MHz; 4: 4500 MHz and 5000 MHz). A connection between two or more antennas is characterized by a channel, a signal frequency as well as a waveform, which is a type of communication protocol. We consider two waveforms: Point-To-Point (PTP) between two antennas and Point-to-MultiPoint (PMP) between an antenna and multiple other antennas. These waveforms constrain the network to have a tree topology. The root of this tree topology is called the *master hub* and it can have a special role in the network depending on its use case. The leaves of the tree are called *users* and the rest of the nodes are *middle hubs*.

The quality of a tactical network is evaluated according to which traffic scenario(s) it is expected to support. In all cases, the network's bottleneck is its weakest connection in terms of effective throughput (the direct data transmission speed between two nodes divided by how many data streams use it). We maximize this bottleneck to have the best possible worst-case data transmission speed in the network, so that vital information can be relayed among all nodes as quickly as possible.

In practice, usual instances of the problem have around 50 nodes and a solution is typically designed by hand. With up to around 20 nodes, Mixed Integer Programming formulations of approximated versions of the problem have had success finding good suboptimal solutions by using Iterated Local Search [2] and Column Generation [3]. With 10 nodes, exhaustive enumeration is possible.

We consider the full problem with a physically-modeled signal, i.e. this is the version that our industrial partner *Ultra Intelligence & Communications* is engaged to give practical solutions. Our approach separates it into three nested sub-problems. At the highest level, our algorithm uses a Tabu Beam Search for the topology design. At the mid level, a mix of exhaustive enumeration, meta-heuristics, heuristics and bounding procedures is used for the network configuration. At the lowest level, we use an intuitive geometrically-based heuristic to align and configure the antennas.

The paper is organized as follows. In Section 2, we describe how we modeled the tactical wireless network design problem with multi-beam antennas. In Section 3, we present our multi-level algorithm for this problem. In Section 4, we give experimental results on the performance of this algorithm on synthetic data. Finally, concluding remarks are reported in Section 5.

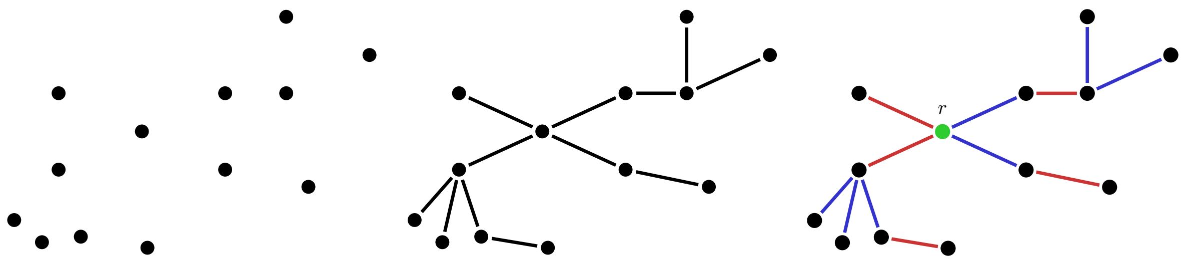
# 2 Problem description

Given an instance of the problem, we wish to design a tactical network that connects the nodes in a way that maximizes the bottleneck of the network.

## 2.1 Instance

An instance of the problem is defined by a set of nodes $V$, geographically distributed according to known $(x_v, y_v)$ coordinates $\forall\, v \in V$, Figure 1(a). Moreover, for every signal frequency $f \in F = \{2000, 2400, 4500, 5000\}$ and for every pair of nodes $u, v \in V$ (with $u \neq v$), the two physical properties that describe the amount of signal that is lost between them are given. They are the path loss $p^f_{uv}$ and fade margin $m^f_{uv}$, both given in decibels. Since these quantities are the same in both directions, both $p^f$ and $m^f$ can be represented as symmetric matrices with zeros on their diagonals.

**Figure 1: An Example of the problem**




**(a) Instance** **(b) Topology** **(c) Configured Network**

Finally, an instance of the problem also defines a vector of weights $(\omega_A, \omega_B, \omega_C)$ that represent the relative weights of the three possible network traffic scenarios, used in the computation of the objective function. They represent how much of each traffic scenario the network is expected to support. These weights will be explained in more depth with the objective function.

## 2.2 Network

A network on this instance is made up of

- a tree topology $(V, E)$, Figure 1(b);
- a network configuration, which is comprised of a master hub $r \in V$ (the green node in Figure 1(c)), a waveform assignment that partitions the topology's edges into PTP and PMP connections and a channel and frequency assignment which gives each connection a signal frequency;
- the antenna configurations, which include the alignments of every used antenna as well as their sets of activated beams.

A valid master hub is any node $r \in V$.

A valid topology is an undirected tree $(V, E)$ such that it respects the maximum degree constraint of at most 11 (for all nodes except the master hub which can have degree 20). This is because, in practice, we consider that a PMP connection between a node $v \in V$ and a set of other nodes $W \subset V$ is limited to $|W| = 10$ connected nodes. Furthermore, only the master hub can have two different outgoing PMP connections, as pictured in the example in Figure 1. All other nodes have one ingoing arc and at most one outgoing PMP connection.

Given a valid topology $(V, E)$, a master hub $r \in V$ defines an arborescence $(V, \vec{E}^*)$ on the topology, i.e. a directed tree rooted in $r$. This arborescence defines a single predecessor for every node $v \in V$ except for $r$. It also defines successors for $r$ and every $v$ that is not a leaf of the topology. For every node $v \in V$, we define a number $d_v$ of descendants (including itself) as

$$d_v = 1 + \sum_{w \text{ direct successor of } v} d_w. \tag{1}$$

Hence, the number $d_v$ of descendants of a leaf is equal to 1.

A valid waveform assignment partitions all the topology's edges $E$ into individual connections. For every regular node $v \neq r$, the edges between it and all of its successors form a single connection. For the master hub, they can be partitioned in up to two connections. The connections are either PTP if they contain a single edge or PMP otherwise. For a given topology and master hub, the number of valid waveform assignments is equal to the number of partitions of the master hub's successors in 2 sets.

A valid channel assignment is such that for every node $v \in V$ that is in two connections, these connections must be on a different channel. Because of the tree structure of the network and because we consider only two channels, there are only two possible channel assignments for a given topology with a master hub and a waveform assignment. An example of valid channel assignment is given in Figure 1(c)) where the colors red and blue are used to differentiate the two channels. Given a valid channel assignment, a valid frequency assignment simply assigns to every connection one of two possible signal frequencies belonging to its channel (i.e., 2000 MHz or 2400 MHz for the channel on frequency band 3+ and 4500 MHz or 5000 MHz for the channel on frequency band 4).

For every used antenna $a$, a valid alignment is an angle $\phi_a \in [0, 2\pi[$ and a valid beam configuration is a set $B_a \subset [\![0, 23]\!]$ of beams with $0 \leq |B_a| \leq 7$. If $B_a = \emptyset$, then the antenna is set to its omni-mode, which sends and receives the same amount of signal in all directions.

## 2.3 Objective

Given every parameter of the network, we use a physically-based signal modeling to compute the direct throughput $TP_{uv}$ (transmission speed) for every edge $[u, v] \in E$. For each of the three traffic modeling scenarios $A, B$ and $C$, we compute a number of data streams $n_{uv}^X$ (with $X \in \{A, B, C\}$) for every edge $[u, v]$. The objective function that we aim to maximize takes into account both the bottleneck and the mean throughput of each scenario, namely

$$\min_{[u,v]\in E} \frac{TP_{uv}}{n_{uv}^X} \qquad \text{and} \qquad \operatorname*{mean}_{[u,v]\in E} \frac{TP_{uv}}{n_{uv}^X} \qquad \text{for } X \in \{A, B, C\},$$

where mean is the usual arithmetic mean. More detail is given below.

In order to compute the direct throughput of an edge $[u, v] \in E$ in a connection with signal frequency $f$, we begin by computing the antenna gain $g_{uv}^f$ of the antenna $a$ at node $u$ in the direction of node $v$ and $g_{vu}^f$ in the opposite direction. The antenna gain at node $u$ in the direction of node $v$ with angle $\phi_a$ and beam set $B_a$ with $|B_a| \geq 1$ is given (in decibels) by

$$g_{uv}^f = 10 \log_{10} \sum_{b \in B_a} \exp_{10} \left( g_{uv,b}^f \middle/ 10 \right), \tag{2}$$

with

$$g_{uv,b}^f = g_{\max}(|B_a|, f) \quad -3 \, \frac{\log_{10} \left( \cos \left( \Delta_\phi(\phi_a + b\frac{2\pi}{24}, x_u, y_u, x_v, y_v) \right) \right)_+}{\log_{10} \left( \cos \left( \Delta_{3dB}(f)/2 \right) \right)_+}, \tag{3}$$

where $g_{\max}(|B_a|, f)$ is the gain in decibels in the maximal direction of each beam, $\Delta_\phi(\phi_a + b\frac{2\pi}{24}, x_u, y_u, x_v, y_v)$ is the angle deviation between this maximal direction and the node $v$ relative to node $u$, $\Delta_{3dB}(f)$ is the 3 dB beam width (the width of the beam at which there is a 3 dB loss in signal) and $(\cdot)_+ = \max\{\cdot, 0\}$. We use

$$g_{\max}(|B_a|, f) = \begin{cases} 13 - 10 \log_{10}(|B_a|), & \text{if channel}(f) = 3+ \\ 15 - 10 \log_{10}(|B_a|), & \text{if channel}(f) = 4, \end{cases}$$

and

$$\Delta_{3dB}(f) = \begin{cases} 60^\circ, & \text{if channel}(f) = 3+ \\ 50^\circ, & \text{if channel}(f) = 4. \end{cases}$$

In the case of $|B_a| = 0$ (the antenna being in omni-mode), the antenna gain is simplified to

$$g_{uv}^f = g_{\max}(0, f) = \begin{cases} 4, & \text{if channel}(f) = 3+ \\ 6, & \text{if channel}(f) = 4. \end{cases}$$

Even if all connected nodes are perfectly aligned to activated beams, we can see that the omni-mode is more advantageous if more than $|B_a| = 7$ beams are necessary.

Given the antenna gains in both directions $g_{uv}^f$ and $g_{vu}^f$ and the path loss $p_{uv}^f$ between the two nodes, we can then compute the signal strength (in decibel-milliwatts)

$$s_{uv}^f = 30 + g_{uv}^f + g_{vu}^f - p_{uv}^f. \tag{4}$$

These signal strengths are computed between every pair of antennas that use the same frequency and not only the pairs that are actually connected in the network. The extra signal from the unconnected pairs that use the same frequency creates interference on the actual connections. The resulting interference between connected nodes $u$ and $v$ on frequency $f$ is given (in milliwatts) by

$$i_{uv}^f = \sum_{\substack{w \neq u,v \text{ with an antenna} \\ \text{that uses frequency } f}} \left( \exp_{10}\left(\frac{s_{uw}^f}{10}\right) + \exp_{10}\left(\frac{s_{wv}^f}{10}\right) \right). \tag{5}$$

With the interference, we can compute the Signal-to-Noise Ratio (SINR) of the connection between nodes $u$ and $v$ (in decibel-milliwatts) with

$$S_{uv}^f = s_{uv}^f - m_{uv}^f - 10 \log_{10}\left( i_{uv}^f + \exp_{10}\left(\frac{NP}{10}\right)\right), \tag{6}$$

where $m_{uv}^f$ is the fade margin and $NP$ is the receiver antenna's noise power. Assuming a 20 MHz bandwidth and a 10 dB noise figure, we use

$$NP = -174 + 10 \log_{10}\left(20 \cdot 10^6\right) + 10.$$

The direct throughput $TP_{uv}$ is then finally taken from Table 1.

**Table 1: Direct throughput lookup table**

| SINR $S$ | Throughput |
|---|---|
| $S < 2$ | 0 |
| $2 \leq S < 5$ | 6.5 |
| $5 \leq S < 9$ | 13 |
| $9 \leq S < 11$ | 19.5 |
| $11 \leq S < 15$ | 26 |
| $15 \leq S < 18$ | 39 |
| $18 \leq S < 20$ | 52 |
| $20 \leq S < 25$ | 58.5 |
| $25 \leq S < 29$ | 65 |
| $29 \leq S$ | 78 |

Since a PMP connection uses the same channel for multiple edges, this channel is shared in time among the edges because only one edge can communicate at once. A PTP connection, being comprised of a single edge, does not have to account for channel sharing in its protocol and thus can transmit more throughput. In order to take this into account, the direct throughput for PTP connections is multiplied by a factor $\lambda_{\text{PTP}} = 2$.

The three traffic modeling scenarios we use are the following:

$A$: a single data stream from any node in the network to any other node (1 single data stream at once);

$B$: a data stream from every node to the master hub or, equivalently, from the master hub to every node ($|V|-1$ data streams at once);

$C$: a data stream from every node to every other node ($|V|(|V|-1)$ data streams at once).

Since there is one single data stream in traffic scenario $A$, we set $n_{uv}^A = 1$ for all edges $[u,v] \in E$.

For traffic scenario $B$, all nodes are sending traffic to the master hub $r$ except for $r$ itself. Each edge that is connected to a leaf in the topology is carrying a single data stream. The closer an edge is to the root $r$, the more data streams it will carry. We therefore consider the number $d_v$ of descendants of every node $v$, as defined above, and set $n_{uv}^B = d_v$ for all edges $[u,v] \in E$.

For traffic scenario $C$, all nodes are sending traffic to each other. Each edge is carrying all the signals of its descendants to the rest of the nodes as well as the signals coming from the other direction. In this case, we set $n_{uv}^C = 2\,d_v(|V|-d_v)$ for all edges $[u,v] \in E$.

We can now give the explicit form of the objective function. It depends on the topology $T$, the choice of the master hub $r$, the partition $\pi$ of the successors of $r$ into two sets, the antenna configurations $\alpha$, the channel assignment $\sigma$ and the frequency assignment $\varphi$. As explained above, it is a weighted sum of bottleneck and mean values, namely

$$\begin{aligned} O(T,r,\pi,\sigma,\varphi,\alpha) =& \omega_A \left( \min_{[u,v]\in E} TP_{uv} + \frac{1}{Z} \operatorname*{mean}_{[u,v]\in E} TP_{uv} \right) \\ &+ \omega_B \left( \min_{[u,v]\in E} \frac{TP_{uv}}{d_v} + \frac{1}{Z} \operatorname*{mean}_{[u,v]\in E} \frac{TP_{uv}}{d_v} \right) \\ &+ \omega_C \left( \min_{[u,v]\in E} \frac{TP_{uv}}{2\,d_v(|V|-d_v)} + \frac{1}{Z} \operatorname*{mean}_{[u,v]\in E} \frac{TP_{uv}}{2\,d_v(|V|-d_v)} \right), \end{aligned} \tag{7}$$

where $Z$ is a normalization constant that we fix equal to half of the maximum direct throughput, that is $Z = \frac{78}{2} = 39$. Note that the smallest possible number $n_{uv}^C = 2d_v(|V|-d_v)$ of data streams associated with an edge $[u,v]$ is $|V|-1$ for scenario C (where $v$ is then a leaf) while $n_{uv}^A$ and $n_{uv}^B$ can be equal to 1. In order to ensure that the magnitude of the values coming from scenario C are comparable to those of scenarios $A$ and $B$, we choose the weights $\omega_A, \omega_B, \omega_C$ such that

$$\omega_A + \omega_B + \frac{\omega_C}{|V|-1} = 1.$$

The full optimization problem we are trying to solve can be written as

$$\max_{\substack{\text{valid}\\ \text{topologies } T}} \ \max_{r\in V} \ \max_{\substack{\text{partitions } \pi \text{ of}\\ r\text{'s successors}\\ \text{in 2 sets}}} \ \max_{\substack{\text{antenna}\\ \text{configurations } \alpha}} \ \max_{\substack{\text{channel}\\ \text{assignments } \sigma}} \ \max_{\substack{\text{frequency}\\ \text{assignments } \varphi}} O(T,r,\pi,\alpha,\sigma,\varphi). \tag{8}$$

Given a valid topology $T$, $O^{max}(T)$ will denote the largest value $O(T,r,\pi,\alpha,\sigma,\varphi)$ over all valid choices for $r,\pi,\alpha,\sigma$ and $\varphi$. Hence, Problem (8) can be rewritten as

$$\max_{\substack{\text{valid}\\ \text{topologies } T}} \quad O^{max}(T). \tag{9}$$

# 3 Algorithm

In order to solve Problem (9) we use a scheme that can be roughly described as follows:

- A Tabu Beam Search (TBS) explores the space of valid topologies;
- For a topology $T$ generated by the TBS, we consider valid choices for a master hub $r$ and a partition $\pi$ of its successors in two sets. For each choice of $r$ and $\pi$, we consider a heuristic procedure that determines a valid antenna configuration $\alpha$;
- Given a topology $T$, a muster hub $r$, a partition $\pi$ of the successors of $r$ and an antenna configuration $\alpha$, we consider the two possible valid channel assignments $\sigma$ and give procedures to either generate a valid frequency assignment $\varphi$ or to bound the value $O(T, r, \pi, \alpha, \sigma, \varphi)$ over all possible valid frequency assignments.

In the following, we describe in detail the algorithmic choices of these three phases.

## 3.1 Master hub selection

Let $T$ be a valid topology with maximum degree at most 20 and with at most one vertex of degree larger than 11. Let $R(T)$ be the set of nodes in $T$ that are not leaves. If $T$ contains a vertex $v$ of degree larger than 11, then we set $R'(T) = \{v\}$. Otherwise, for each node $v$ in $R(T)$, we determine a number $n(v) = p_d(v) + p_e(v)$, where $p_d(v)$ is the position of $v$ in $R(T)$ if the set is ordered by increasing degree, while $p_e(v)$ is its position when $R(T)$ is ordered by decreasing eccentricity. Hence, $n(v)$ has a high value if $v$ has a large degree and is close to the center of $T$. We then define $R'(T)$ as the set of vertices $v$ in $R(T)$ with $n(v) \geq \text{median}_{u \in R(T)}\, n(u)$. The nodes in $R'(T)$ are the only candidates to be the master hub in $T$.

## 3.2 Partition of the successors

As will be explained in Section 3.6, we consider three ways to compute the quality of a topology $T$. We compute either an estimation $O^{Est}(T)$, an upper bound $O^{UB}(T)$ or a lower bound $O^{LB}(T)$ on $O^{Max}(T)$. When computing $O^{Est}(T)$ and $O^{UB}(T)$, we consider all possible partitions of the successors of the master hub. The way to compute $O^{LB}(T)$ depends on the degree of the master hub $r$. If it is at most 7, then we consider all possible partitions of the successors of $r$. Otherwise, we use a Greedy Randomized Adaptive Search Procedure (GRASP). Its initial solution is the partition that leads to $O^{UB}(T)$ and the local search is a steepest descent with a neighborhood that considers every way of switching a successor from one set to the other. The implementation details are described in [1].

## 3.3 Antenna configuration

Let $T$ be a valid topology, let $r$ be a valid master hub for $T$ and let $\pi$ be a valid partition of the successors of $r$ into two sets. As explained in Section 2, for every node $u \neq r$, the edges linking $u$ to its successors form a single connection. For the master hub $r$, both sets of the partition $\pi$ form a connection.

To configure an antenna $a$ at node $u \in V$ for its connection to a set $W \subset V$ of successors, we use a simple geometrically-based heuristic that minimizes the number of activated beams $B_a \subseteq [\![0, 23]\!]$, since additional beams decrease the signal strength for all edges in the connection.

If $W$ contains only one successor $v$ of $u$, a single activated beam $B_a = \{0\}$ is optimal and the optimal alignment is the angle between nodes $v$ and $u$ with respect to the azimuth. For the case with multiple connected nodes (i.e., $|W| \geq 2$), we define, for a beam $b \in [\![0, 23]\!]$ with beam 0 starting at an angle of $\theta$, the angular range $\Phi_b(\theta) \subseteq [0, 2\pi[$ covered by beam $b$ as

$$\Phi_b(\theta) = \begin{cases} [l, u[\,, & \text{if } u > l \\ [l, 2\pi[ \cup [0, u[\,, & \text{otherwise,} \end{cases}$$

with

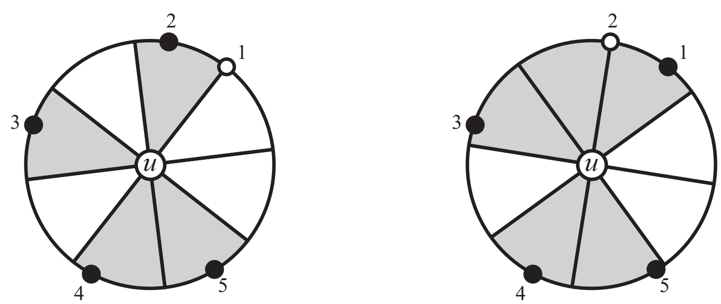
$$l = \left(\theta + \frac{2\pi}{24} b\right) \bmod 2\pi \quad \text{and} \quad u = \left(\theta + \frac{2\pi}{24}(b+1)\right) \bmod 2\pi,$$

where we assume an effective beam width of $\frac{2\pi}{24}$. For instance, the range of beam 0 is given by

$$\Phi_0(\theta) = \begin{cases} \left[\theta, \theta + \frac{2\pi}{24}\right[, & \text{if } \theta + \frac{2\pi}{24} < 2\pi \\ \left[\theta, 2\pi\right[ \cup \left[0, \theta + \frac{2\pi}{24} - 2\pi\right[, & \text{otherwise.} \end{cases}$$

Let $\theta_v$ be the angle between $u$ and a node $v \in W$. In order to partition the nodes of $W$ into activated beams, we try, for each $v \in W$, a configuration of the beams in which $v$ is at the minimum angle of the range covered by beam 0 (i.e. the range of beam 0 is $\Phi_0(\theta_v)$). This is depicted in Figure 2 for an antenna with 8 beams (for clarity), with $W = \{1, 2, 3, 4, 5\}$ and $v = 1$ or 2.

**Figure 2: Two beam partitionings for an example with $W = \{1, 2, 3, 4, 5\}$ and 8 beams**




**(a)** $v = 1$ **requires 4 activated beams** **(b)** $v = 2$ **requires 5 activated beams**

Let $v^*$ be equal to the vertex $v \in W$ that minimizes the number of required beams (ties are broken at random). If more than 7 beams must be activated, it is preferable to switch the antenna to its omni-mode. Otherwise, we set $B_a$ equal to the set of beams that contain at least one vertex of $W$ when beam 0 starts at angle $\theta_{v^*}$. For instance, in Figure 2(a), the set of activated beams would be $B_a = \{0, 2, 4, 5\}$.

If we set $\phi_a$ equal to $\theta_{v^*} + \frac{\pi}{24}$, then $v^*$ will be exactly on the border of beam 0 and will therefore not be served well. To avoid this, we consider the extra angular space we have at the end of the activated beams. More precisely, for each $v \in W$, we compute the angular space $s_v$ between $v$ and the end of the beam $b_v \in B_a$ to which it belongs

$$s_v = \left( \left( \theta_{v^*} + \frac{2\pi}{24}(b_v + 1) \right) \bmod 2\pi - \theta_v \right) \bmod 2\pi.$$

We then set

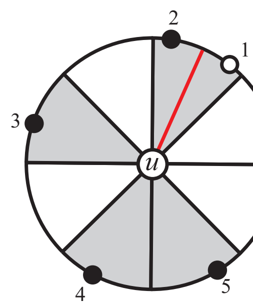
$$\phi_a = \left( \theta_{v^*} + \frac{\pi}{24} - \frac{1}{2} \min_{v \in W} s_v \right) \bmod 2\pi. \tag{10}$$

For illustration, the smallest extra angular space in Figure 2(a) comes from node 2 and the corresponding antenna alignement is shown with a red line in Figure 3.

## 3.4 Channel assignment

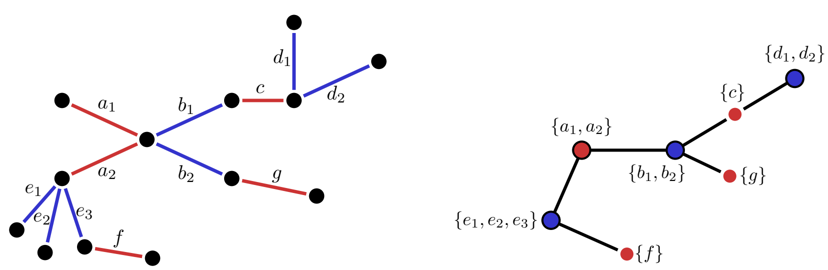
Given a valid topology $T$ with a master hub $r$ and a bipartition of its successors in two sets, a valid channel assignment is such that the connections incident to a node must have different channels. This subproblem can be modeled as a coloring problem on a graph $G$, called *channel assignment tree*, where each connection is a node, and the connections of each node $v$ in $T$ form a clique in $G$. Since we consider only two possible channels, the cliques contain at most two connections and are simply edges in the channel assignment graph. Moreover, because $T$ is a tree, the graph $G$ is also a tree. The channel assignment subproblem is thus reduced to a 2-coloring of the nodes of $G$, which has only two possible solutions. In our algorithms, we try both valid channel assignments. For illustration, the master hub

**Figure 3: Antenna alignment for the example of Figure 2**




in Figure 4(a) is the node incident to $a_1, a_2, b_1, b_2$. It has two connections, one that contains the edges $a_1$ and $a_2$, and another one that contains the edges $b_1$ and $b_2$. This corresponds to the edge that links $\{a_1, a_2\}$ to $\{b_1, b_2\}$ in Figure 4(b). The node at the other extremity of $a_2$ has a connection made of $e_1, e_2, e_3$ and another one made of $a_1, a_2$, and this induces the edge that links $\{e_1, e_2, e_3\}$ to $\{a_1, a_2\}$ in Figure 4(b).

**Figure 4: Channel assignment is a 2-coloring problem**




**(a) Edges grouped by connections** **(b) Channel assignment tree $G$**

## 3.5 Frequency assignment

Let $T$ be a valid topology, let $r$ be a valid master hub for $T$, let $\pi$ be a valid partition of the successors of $r$ into two sets, let $\alpha$ be an antenna configuration, and let $\sigma$ be a channel assignment. We aim to determine a valid frequency assignment $\varphi$ that minimizes $O(T, r, \pi, \alpha, \sigma, \phi)$. This is a difficult problem since each connection in $T$ has two possible frequency assignments. Exhaustive enumeration is not applicable for instances of typical practical size. To overcome this issue, we show how to determine upper and lower bounds on the optimal value of a frequency assignment, and we also describe a simple way to produce a rough estimation of this optimal value.

Let us start with the upper bounding procedure. Let $c(u,v)$ be the set of valid frequencies for the edge $[u,v]$. As already mentioned, $c(u,v) = \{2000, 2400\}$ if the connection that includes $[u,v]$ uses band 3+ and $c(u,v) = \{4500, 5000\}$ if it uses band 4. We modify equations (2), (4) and (6) by ignoring interference and by replacing $p^f_{uv}$ and $m^f_{uv}$ by lower bounds $p^{LB}_{uv}, m^{LB}_{uv}$ and $g^f_{uv,b}, s^f_{uv}, S^f_{uv}$ by upper bounds $g^{UB}_{uv,b}, s^{UB}_{uv}, S^{UB}_{uv}$ where

$$p^{LB}_{uv} = \min_{f \in c(u,v)} p^f_{uv},$$

$$m^{LB}_{uv} = \min_{f \in c(u,v)} m^f_{uv},$$

$$g_{uv}^{UB} = 10\,\log_{10} \sum_{b \in B_a} \exp_{10}\left( \frac{\max_{f\in c(u,v)} g_{\max}(|B_a|, f) - 3\,\frac{\log_{10}\left(\cos\left(\Delta_\phi(\phi_a + b\frac{2\pi}{24}, x_u, y_u, x_v, y_v)\right)\right)_+}{\max_{f\in c(u,v)} \log_{10}(\cos(\Delta_{3dB}(f)/2))_+}}{10} \right),$$

$$s_{uv}^{UB} = 30 + g_{uv}^{UB} + g_{vu}^{UB} - p_{uv}^{LB},$$

$$S_{uv}^{UB} = s_{uv}^{UB} - m_{uv}^{LB} - NP.$$

An upper bound on the maximal value of $O(T, r, \pi, \alpha, \sigma, \varphi)$ over all valid frequency assignments $\varphi$ can be obtained by considering $S_{uv}^{UB}$ instead of $S_{uv}^{f}$ to derive the throughputs.

An estimation on the maximal value of $O(T, r, \pi, \alpha, \sigma, \varphi)$ over all valid frequency assignments $\varphi$ can be obtained in a similar way by considering $p_{uv}^{Est}, m_{uv}^{Est}, g_{uv,b}^{Est}, s_{uv}^{Est}$ and $S_{uv}^{Est}$, where $\min\limits_{f\in c(u,v)}\{\cdot\}$ and $\max\limits_{f\in c(u,v)}\{\cdot\}$ are replaced in the above equations by $\underset{f\in c(u,v)}{\text{mean}}\{\cdot\}$.

For getting a lower bound on the maximal value of $O(T, r, \pi, \alpha, \sigma, \varphi)$ over all valid frequency assignments $\varphi$, we build a solution with a greedy algorithm that considers the connections in a breadth-first manner, starting from the master hub all the way to the leaves. At each step, we choose the valid frequency for the considered connection that maximizes the objective.

## 3.6 Measuring the quality of a topology

We consider three ways to measure the quality of a topology $T$. The first one builds a feasible solution and its value is therefore a lower bound on $O^{Max}(T)$. To do this, we have to determine a master hub $r$, a partition $\pi$ of its successors, an antenna configuration $\alpha$, a channel assignment $\sigma$ and a frequency assignment $\varphi$. We compute $O(T, r, \pi, \alpha, \sigma, \varphi)$ for the following combinations of $r, \pi, \alpha, \sigma$ and $\varphi$:

- $r$ is any node in $R'(T)$ (see Section 3.1);
- if $r$ has degree at most 7, we try all partitions $\pi$ of its successors in two parts, otherwise we consider only the partitions visited by the GRASP algorithm (see Section 3.2);
- we use the antenna configuration $\alpha$ obtained from $T, r, \pi$, as explained in Section 3.3;
- $\sigma$ is one of the two possible channel assignments (see Section 3.4);
- we use the greedy algorithm to determine a frequency assignment $\varphi$ (see Section 3.5).

The combination that leads to the largest value $O(T, r, \pi, \alpha, \sigma, \varphi)$ is a feasible solution and $O^{LB}(T)$ denotes its value, which is a lower bound on $O^{Max}(T)$.

Determining $O^{LB}(T)$ is computationally expensive. We therefore consider another function that estimates the value of $O^{Max}(T)$. We try the same combinations of $r, \pi, \alpha, \sigma$ as above, except that all partitions $\pi$ of the successors of the master hub $r$ are tested, regardless of the degree of $r$. Instead of using the greedy algorithm for determining a frequency assignment $\varphi$, we use $S_{uv}^{Est}$ to determine the throughput $TP_{uv}$ (see Section 3.5). In what follows, $O^{Est}(T)$ will denote the largest estimation of $O^{Max}(T)$ for the tested combinations of $r, \pi, \alpha, \sigma$. Intensive tests on instances having up to 30 nodes have shown that a large proportion of pairs $(T, T')$ of topologies verify $O^{Max}(T) > O^{Max}(T')$ if and only if $O^{Est}(T) > O^{Est}(T')$. Hence, in the algorithm that we have designed to solve Problem (8), we will use $O^{Est}$ to select topologies in large sets of candidates, and we will then compute lower bounds on the values of the selected topologies using $O^{LB}$. This will be explained in more details in the next section.

Let $\mathcal{T}^*$ be a set of topologies $T$ with large values $O^{LB}(T)$. In order to determine whether a topology $T'$ with a large value $O^{Est}(T')$ is potentially better than a topology in $\mathcal{T}^*$, we compute an upper bound $O^{UB}(T')$ on $O^{Max}(T')$. If this upper bound is smaller that the minimum value $O^{LB}(T)$ for $T$ in $\mathcal{T}^*$, we know that $T'$ is worse than all topologies in $\mathcal{T}^*$. This upper bound is obtained in a similar way to what we described to compute $O^{Est}$. The only difference is that we use $S_{uv}^{UB}$ to determine the throughput

$TP_{uv}$ (see Section 3.5). Here also, $O^{UB}(T)$ will denote the largest upper bound on $O^{Max}(T)$ for the tested combinations of $r, \pi, \alpha, \sigma$.

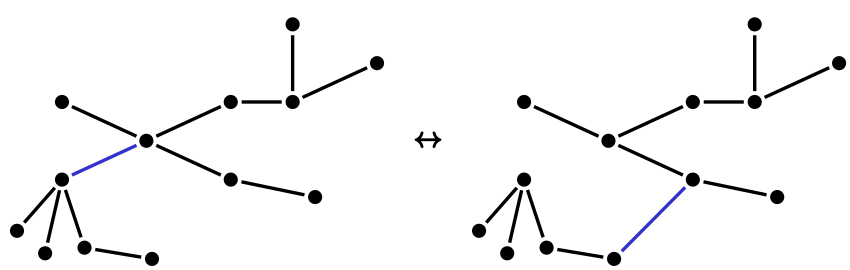
## 3.7 A Tabu Beam Search

The strategy we have chosen to determine an optimal solution to Problem (8) is a Tabu Beam Search [4] also called Sequential Fan Candidate List Search. At every iteration, we have a set $\mathcal{T}$ of $\kappa \geq 1$ topologies and we choose $\kappa$ promising neighbors for each $T \in \mathcal{T}$. Among these $\kappa^2$ generated neighbors, we keep the $\kappa$ best ones that constitute the new set $\mathcal{T}$ for the next iteration. Each topology in $\mathcal{T}$ has an associated tabu list. When a neighbor $T'$ of a topology $T \in \mathcal{T}$ is kept for the next iteration, we obtain the tabu list associated with $T'$ from that of $T$ by forbidding the move that would bring the search back to $T$. With $\kappa = 1$, this corresponds to a standard Tabu Search.

For two nodes $u$ and $v$, we can compute the ideal direct throughput between them, assuming a perfect alignment of the antenna with only one activated beam, ignoring interference and using a PMP waveform and using the largest frequency 4500 MHz. If this ideal throughput is null, we do not consider $[u, v]$ as a potential edge in a topology. We use $E_{not}$ to denote the set of such pairs $(u, v)$ with $u < v$.

The neighborhood $N(T)$ of a topology $T$ consists of all the topologies that can be obtained by dropping an edge $e$ and reconnecting the topology by adding another edge $e'$. The candidate list $L_e$ of edges that can replace $e$ contains all pairs $(u, v)$ with $u < v$ such that $u$ and $v$ are in different connected components after dropping $e$, $(u, v) \notin E_{not}$, and the resulting topology has at most one node of degree larger than 11 and no vertex of degree larger than 20. An example is given in Figure 5.

**Figure 5: An Example of a neighbor topology**



Every considered topology has two associated Tabu lists that contain respectively the edges that cannot be dropped (Tabu-drop list) and the edges that cannot be added (Tabu-add list). The Tabu-drop list forbids dropping an edge for $\frac{1}{2}\sqrt{|V| - 1}$ iterations, which is half of the square root of the number of edges in the topology. The Tabu-add list forbids adding an edge for $\sqrt{|V|(|V| - 1)/2 - |E_{not}|}$ iterations, which is the square root of the number of edges that can potentially belong to an optimal topology. To avoid cycling, we add equiprobable values in $\{-1, 0, 1\}$ to the above Tabu list sizes.

The central procedure of our TBS builds $\kappa$ promising neighbors of a topology $T$. This is done as follows. Let $L$ be the list of pairs $(u, v)$ of nodes with $u < v$, ordered by increasing value $d_{uv}$, where

$$d_{uv} = \underset{f \in \{2000,2400,4500,5000\}}{\text{mean}} (p_{uv}^f + m_{uv}^f). \tag{11}$$

The candidate list $L_e$ of edges that can replace an edge $e$ in $T$ is a sublist of $L$. Let $\rho \in (0, 1]$ be a parameter that controls the proportion of neighbors explored at each iteration. We generate a subset $L'_e$ that contains $\lfloor \rho \, |L_e| \rfloor$ pairs of $L_e$. These pairs are selected at random with a probability that a pair is chosen inversely proportional to its position in $L_e$.

For every edge $e$ in $T$ and every pair $(u, v)$ in $L'_e$, we can produce a neighbor $T_{e,(u,v)}$ of $T$ by replacing $e$ with $[u, v]$. Let $\mathcal{L}(T) = \{(e, (u, v)) \,|\, e \in T \text{ and } (u, v) \in L'_e\}$. The elements of $\mathcal{L}(T)$

are randomly ordered with the only constraint that $(e,(u,v))$ precedes $(e',(u',v'))$ if $e = e'$ and $(u,v)$ precedes $(u',v')$ in $L_e$. We then consider the elements of $\mathcal{L}(T)$ sequentially to construct a set $N'(T) \subseteq N(T)$ of $\kappa$ promising neighbors of $T$. Let $O^{Est}_{best}$ be the largest value of function $O^{Est}$ encountered so far. In what follows, we say that $(e,(u,v))$ is *tabu* if $e$ belongs to the Tabu-drop list of $T$ or $[u,v]$ belongs to its Tabu-add list, and $O^{Est}(T_{e,(u,v)}) \leq O^{Est}_{best}$. A topology $T_{e,(u,v)}$ can belong to $N'(T)$ only if $(e,(u,v))$ is not tabu.

The procedure that generates $N'(T)$ from $T$ is described in Algorithm 1 that we now explain. The set $N'(T)$ is initialized with the topologies $T_{e,(u,v)}$ of the first $\kappa$ non-tabu elements $(e,(u,v))$ of $\mathcal{L}(T)$. For the next non-tabu elements $(e,(u,v))$ of $\mathcal{L}(T)$, we check if $O^{Est}(T_{e,(u,v)}) > O^{Est}(T')$ where $T' = \text{argmin}_{T'' \in N'(T)} O^{Est}(T'')$, in which case we replace $T'$ with $T_{e,(u,v)}$. We also update $O^{Est}_{best}$ if $O^{Est}(T_{e,(u,v)}) > O^{Est}_{best}$. This process is stopped when a topology is encountered with $O^{Est}(T_{e,(u,v)}) > O^{Est}(T)$ or when the end of list $\mathcal{L}(T)$ is reached.

**Algorithm 1:** Generation of $\kappa$ promising neighbors of a topology $T$

Let $O^{Est}_{best}$ be the largest value of function $O^{Est}$ encountered so far
Initialize $N'(T)$ with the topologies $T_{e,(u,v)}$ associated with the first $\kappa$ non-tabu elements $(e,(u,v))$ of $\mathcal{L}(T)$.
**If** $N'(T)$ does not contain any topology $T'$ with $O^{Est}(T') > O^{Est}(T)$
  **Repeat**
    Select the next non-tabu element $(e,(u,v))$ in $\mathcal{L}(T)$
    **If** $O^{Est}(T_{e,(u,v)}) > O^{Est}(T')$, where $T' = \arg\min_{T'' \in N'(T)} O^{Est}(T'')$
      replace $T'$ with $T_{e,(u,v)}$ in $N'(T)$.
    **If** $O^{Est}(T_{e,(u,v)}) > O^{Est}_{best}$
      update $O^{Est}_{best} \leftarrow O^{Est}(T_{e,(u,v)})$.
  **Until** $O^{Est}(T_{e,(u,v)}) > O^{Est}(T)$ **or** the end of $\mathcal{L}(T)$ is reached.

The TBS algorithm is described in Algorithm 2. To initialize it, we determine a minimum cost spanning tree $T$ where the cost of an edge $[u,v]$ is the value $d_{uv}$ of Equation (11) that we have used to sort the elements of $L$. We then use Algorithm 1 to generate $\kappa$ promising neighbors of $T$ that are put in a set $\mathcal{T}$. We set $O^{LB}_{best} = \max_{T' \in \mathcal{T} \cup \{T\}} O^{LB}(T')$ and $O^{Est}_{best} = \max_{T' \in \mathcal{T} \cup \{T\}} O^{Est}(T')$. Also, we set $T^*$ equal to a topology $T'$ in $\mathcal{T} \cup \{T\}$ with $O^{LB}(T') = O^{LB}_{best}$. The set $\mathcal{T}^*$ that contains the $\kappa$ best encountered topologies is initialized with the $\kappa$ topologies $T'$ in $\mathcal{T} \cup \{T\}$ with largest value $O^{LB}(T')$. To every topology $T'$ in $\mathcal{T}$, we associate a Tabu-add list that contains the edge that was removed from $T$ to obtain $T'$, and a Tabu-drop list that contains the edge that was added to obtain $T'$.

At every iteration of the TBS algorithm, we generate $\kappa$ promising neighbors for each $T \in \mathcal{T}$. Among these $\kappa^2$ generated neighbors, we keep the $\kappa$ best ones according to function $O^{Est}$, avoiding duplicate topologies. They constitute the new set $\mathcal{T}$ for the next iteration. When a neighbor $T'$ of a topology $T \in \mathcal{T}$ is kept for the next iteration, we obtain the tabu lists associated with $T'$ from that of $T$ by forbidding the move that would bring the search back to $T$. Hence, if $T'$ if obtained from $T$ by dropping edge $e$ and adding edge $e'$, we add $e'$ in the Tabu-drop list of $T'$ and $e$ in its Tabu-add list. When the best encountered topology $T^*$ is updated with a topology $T'$ of value $O^{LB}(T') >^{LB}_{best}$, we remove all elements of the tabu lists of $T'$ except the edges involved in the move that led from a topology $T$ to $T'$. Hence, if an edge $e$ was replaced by an edge $e'$, the new Tabu-add list of $T'$ is $\{e\}$ while its new Tabu-drop list is $\{e'\}$.

Once the set $\mathcal{T}$ of $\kappa$ most promising neighbors has been generated, we test if some of these topologies should become part of the set $\mathcal{T}^*$ of $\kappa$ best encountered topologies. To do this, we compute the upper bound $O^{UB}(T')$ for every $T'$ in $\mathcal{T}$. We then compute $O^{LB}(T')$ only if $O^{UB}(T') \geq \min_{T'' \in \mathcal{T}^*} O^{LB}(T'')$.

Every $i_{Rmax}$ iterations, the search restarts with the set $\mathcal{T}^*$ that contains the $\kappa$ best topologies encountered since the beginning of the search. The search also restarts if $i^*_{Rmax}$ iterations have been performed without improving $O^{LB}_{best}$. The algorithm is stopped when a time limit $t_{max}$ is reached.

**Algorithm 2:** Tabu Beam Search

```
Find a minimum cost spanning tree T using d_uv as cost function, assign two empty tabu lists to T and set the
  best estimate O^Est_best of the optimal value to O^Est(T).
Initialize 𝒯 with the κ topologies produced by Algorithm 1 when applied to T.
For every topology T' ∈ 𝒯, set its Tabu-add list as {e} and its Tabu-drop list as {e'}, where T' is obtained
  from T by replacing e with e'.
Update O^Est_best and set O^LB_best ← max_{T'∈𝒯∪{T}} O^LB(T').
Set T* ← T', where T' is any topology in 𝒯 ∪ {T} such that O^LB_best = O^LB(T').
Put in 𝒯* the κ topologies T' in 𝒯 ∪ {T} with largest value O^LB(T').
Set i_R ← 1 and i*_R ← 1.
While time t ≤ t_max
    If i_R > i_Rmax or i*_R > i*_Rmax
        Empty all tabu lists and set 𝒯 ← 𝒯*, i_R ← 1 and i*_R ← 1.
    Generate κ promising neighbors for every topology in 𝒯 using Algorithm 1.
    Among these κ^2 topologies, determine the κ best ones according to function O^Est, avoiding duplicate
      topologies, and put them in 𝒯, removing the old ones.
    For each T' ∈ 𝒯
        Let T be the topology such that T' was obtained from T by dropping an edge e and adding an edge e'.
          Associate the tabu lists of T to T' and add e in its Tabu-add list and e' in its Tabu-drop list
        If O^UB(T') ≥ min_{T''∈𝒯*} O^LB(T'')
            If O^LB(T') ≥ min_{T''∈𝒯*} O^LB(T'')
                Add T' to 𝒯* and remove the topology T'' with lowest value O^LB(T'').
                Restart the counter i*_R ← 0.
                If O^LB(T') > O^LB_best
                    Set T* ← T' and O^LB_best ← O^LB(T').
                    Set the Tabu-add list of T' as {e} and its Tabu-drop list as {e'}.
    Set i_R ← i_R + 1 and i*_R ← i*_R + 1.
Return T*.
```

# 4 Computational experiments

We evaluate the performance of our algorithm on synthetic instances of 10, 30 and 50 nodes. For each number of nodes $|V|$, we generate 3 instances and run the algorithm 5 times on each instance. We have used relative traffic scenario weights of

$$\omega_A = \frac{1}{13}, \quad \omega_B = \frac{4}{13}, \quad \omega_C = \frac{8}{13}(|V|-1),$$

thus putting more weight on the more challenging objectives.

## 4.1 Instance generation

Every instance is randomly generated with the same procedure. First, independent coordinates for the nodes $v \in V$ are generated using

$$(x_v, y_v) = \left(\sqrt{U_0}\cos(2\pi U_1),\ \sqrt{U_2}\sin(2\pi U_3)\right), \tag{12}$$

where $U_0, U_1, U_2, U_3 \sim \mathcal{U}(0,1)$ and are independent. These coordinates are then scaled to match the average distance ratio of 10 km. This coordinate generation is repeated until the minimum distance is above 2 km and the maximum distance is below 150 km.

Finally, for each possible edge $[u,v]$ with $u, v \in V$, a single random uniform variable is sampled for the path loss and another for the fade margin. These random variables are then used to compute the path losses and fade margins for $[u,v]$ for the 4 possible frequencies according to empirical cumulative distribution functions derived from North American datasets.

## 4.2 Parameter grid search

We study the influence of two parameters on the performance of our algorithm: the number of parallel searches $\kappa$ and the explored neighborhood ratio $\rho$. We test values of $\kappa \in \{1, 2, 4, 8\}$ and of $\rho \in \{1, 1/2, 1/4\}$. We only consider instances of 10 nodes and 30 nodes. We let our algorithm run for 10 minutes for 10 nodes and for 2 hours for 30 nodes.

For every tested instance, let $O^*$ be the largest lower bound ever reached in all runs and parameter configurations and let $O_{init} = O^{LB}(T_0)$, where $T_0$ is the minimum cost spanning tree. Every $t = t_{max}/1000$ seconds, let $O_t$ be the average value of $O^{LB}(T^*)$ on the five runs. To evaluate the performance of our algorithm, we compute the scaled primal gap between the initial objective value and the maximum ever found per instance with

$$\frac{O^* - O_t}{O^* - O_{init}},$$

as well as the scaled primal integral [5]

$$\frac{1}{t_{\max}} \int_0^{t_{\max}} \frac{O^* - O_t}{O^* - O_{init}} \, dt.$$

The results are given in Figure 6.

As expected, when the number of parallel searches $\kappa$ increases, each iteration takes proportionally longer and the algorithm can do fewer iterations for the same amount of time. When the explored neighborhood ratio $\rho$ decreases away from 1, each iteration takes proportionally less time and the algorithm can do more iterations for the same amount of time. Both of these parameters control the trade-off between exploration and exploitation of the neighborhood structure. By increasing $\kappa$, we allow more local exploration around the current topologies, but we do so at the cost of doing fewer iterations. By decreasing $\rho$, we allow more exploration by doing more iterations but each iteration does not exploit the full local neighborhood as much, which can miss the most promising topologies.

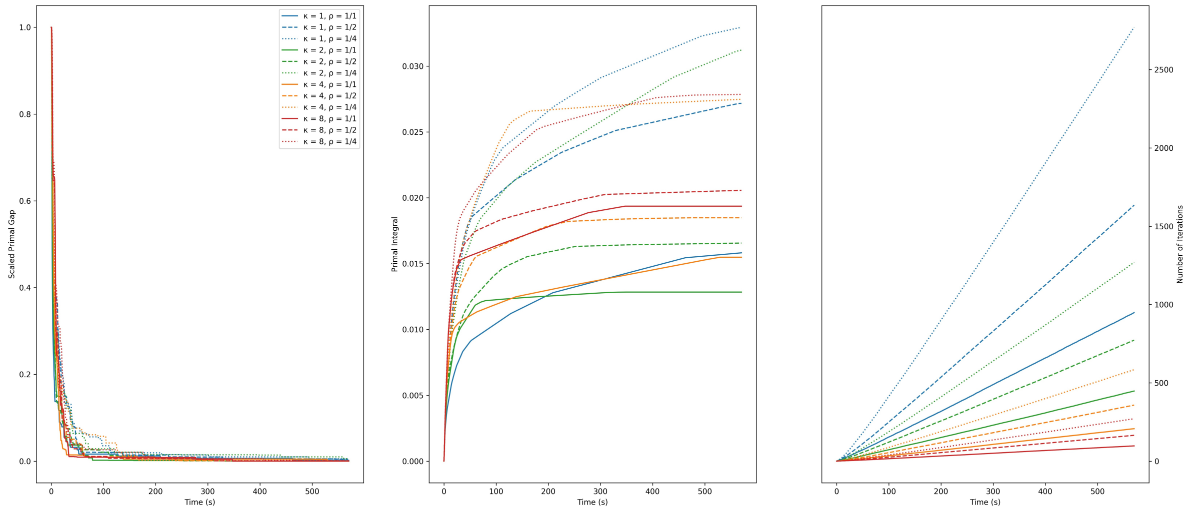
In the small-scale instances with 10 nodes, there are not many possible trees and the algorithm can quickly find the optimal topology in tens to thousands of iterations depending on the parameters, as verified by brute force exhaustive search. In the mid-scale instances with 30 nodes, the algorithm can do less than 100 full iterations (with $\kappa = 1$) in 2 hours and it is still only beginning to explore the intractable size of the space of possible trees.

Looking at the primal integral, we can see that, for both 10 and 30 nodes, the best combinations $(\kappa, \rho)$ of parameters are $(2, 1), (2, \frac{1}{4})$ and $(1, \frac{1}{2})$. The combination $(2, 1)$ seems faster in producing good topologies and we therefore fix the values of the parameters to this combination for the sequel.

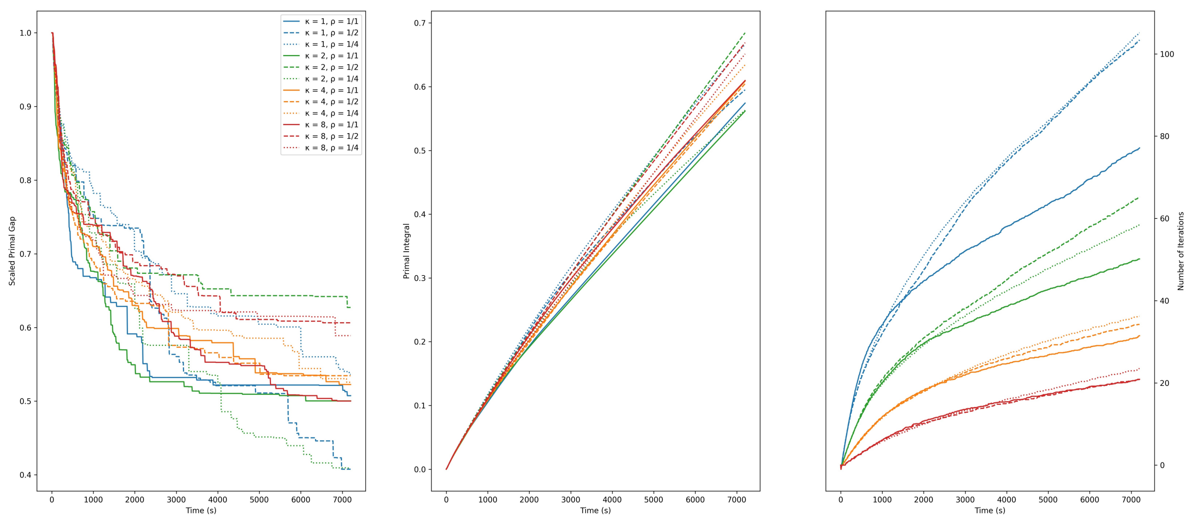
## 4.3 Additional tests and comparisons

For the next expriment, we run our algorithm on instances of 10, 30 and 50 nodes, 3 instances of each. For 10 nodes, we let our algorithm run for 10 minutes, for 30 nodes, 10 hours and, for 50 nodes, 2 days. For each instance, we compare the results of our algorithm against the best known value $O_{best}$. We indicate in Table 2 how close to $O_{best}$ our algorithm reaches in percentage. For 10 nodes, $O_{best}$ is the value of an optimal topology obtained by brute force exhaustive search. For 30 and 50 nodes, we let our algorithm run once for respectively 2 days and 1 week to find suitable approximations. We evaluate the deviation from $O_{best}$ both for our minimum cost spanning tree initial topology and for the final value reached by our TBS algorithm. For perspective, we also evaluate random trees from the space of possible topologies. For 10 nodes, we can also compare our algorithm with a version of the algorithm in [2] adapted to our problem [6]. The means and standard deviations of the results are given in Table 2.

**Figure 6: Parameter grid search**




**(a) Primal gap, primal integral and number of iterations for 10 nodes**




**(b) Primal gap, primal integral and number of iterations for 30 nodes**

By initializing with a minimum spanning tree instead of a random tree, we start the local search from an already promising region of the solution space with 40 to 70 % of the best objective value, when compared to around 2 % for a random tree. This is important since metaheuristic searches like Tabu search are very dependent on the initial solution. When starting from a random tree, it can take many iterations for a local search to reach a value comparable to our minimum spanning tree initialization. Since in our case each iteration is computationally very expensive, this saves a lot of computing time. For instance, we ran our algorithm from a random seed 5 times for each 30 node instance and it achieved a final relative performance of $75.6 \pm 16.0\%$ compared to $83.4 \pm 14.1\%$ when starting from the minimum spanning tree.

Our Tabu Beam Search succeeds in finding much better solutions than the initializations. For 10 nodes, it found the optimal for all instances and in all runs. For 30 and 50 nodes however, we cannot hope for the algorithm to stumble on the optimal topology in an exponentially large solution space, in less than 200 iterations. The search still manages to find very good solutions with more than 80 % of the best known objective value.

**Table 2: Deviation from the best known values**

| | | | Our TBS algorithm | | | |
|---|---|---|---|---|---|---|
| $\|V\|$ | $t_{\max}$, | iterations | Initialization | Final | Algorithm in [6] | Random Tree |
| 10 | 10 m | 473±10 | 71.8 ± 7.7 % | 100.0 ±0.0 % | 65.9 ±9.6 % | 1.9 ±0.8 % |
| 30 | 10 h | 140±63 | 47.2 ± 4.0 % | 83.4 ±14.1 % | – | 1.7 ± 0.4 % |
| 50 | 48 h | 161±22 | 51.6 ±4.5 % | 82.9 ±11.1 % | – | 2.9 ±0.8 % |

The baseline adapted from [2] cannot deal with more than 10 nodes. Moreover, the values it finds for 10 nodes are worse than our very simple minimum spanning tree initialization. Our full algorithm performs even better as it successfully finds the optimal solutions in that case.

# 5 Conclusions

We fully modeled the problem of designing tactical wireless networks with multi-beam antennas. We proposed a multi-level algorithm that tackles this complex problem in three nested procedures:

- a Tabu Beam Search for the exploration of the space of valid topologies;
- exhaustive enumerations, meta-heuristics and heuristics for the choice of the master hub, the partition of its successors, and the channel and frequency assignments;
- a simple heuristic for the antenna configurations.

We performed synthetic experiments to evaluate the performance of our method and found that it is able to produce very good networks and that it significantly outperforms a previous baseline algorithm.

Our algorithm can be refined in many different ways. Currently the computation of the lower bound $O^{LB}(T)$ on $O^{Max}(T)$ is the part that has the most room to improve. Indeed, the exhaustive enumeration that takes place for the choice of the master hub, the partition of its successors and the channel assignment is very time consuming. Perhaps should we make these choices by solving a series of nested metaheuristic local searches with different neighborhoods defined for master hubs, waveform assignments, channel assignments and frequency assignments. The master hub neighborhood could be the adjacent nodes of the current master hub and the other neighborhoods could be defined as suggested in the conclusion of [1]. Each metaheuristic search could also be wrapped in a GRASP procedure which initializes the local searches multiple times from greedy randomized solutions.

The Tabu Beam Search can also be accelerated with the use of Machine Learning. In [7], edge prediction by a GNN is used to predict which neighbor topologies are worth evaluating in every step of the algorithm. Similarly, GNN node prediction could also be used to predict which master hubs are worth evaluating.

**Declaration of Competing Interest** In accordance with Elsevier policy and our ethical obligation as researchers, we are reporting that we do not have received funding from any company that may be affected by the research reported in the enclosed paper. We have disclosed those interests fully to Elsevier, and we have in place an approved plan for managing any potential conflicts arising from that involvement.

**Credit authorship contribution statement** This paper is the summary of the results in **Vincent Perreault**'s Master dissertation [1], so Vincent Perreault is the leading contributor of this work both in terms of algorithmic design (50%) and experimental evaluation (100%). **Alain Hertz** and **Andrea Lodi** contributed equally (25% each) to the algorithmic design on their role of supervisors. The three authors contributed equally to the writing of the paper.

**Data availability** All code and generated instances can be found at https://github.com/VincentPerreault0/MultiWaveformMultiBeam

# References


[1] V. Perreault, Tactical wireless network design for challenging environments, Master's thesis, Polytechnique Montreal (August 2022).

[2] N. S. Hamami, T. C. Chuah, S. W. Tan, Joint resource allocation in multi-radio multi-channel wireless mesh networks with practical sectored antennas, in: 2010 International Conference on Computer Applications and Industrial Electronics, 2010, pp. 316–321. doi:10.1109/ICCAIE.2010.5735096.

[3] L. Zhou, X. Cao, L. Liu, L. Cai, X. Tian, Y. Cheng, On capacity optimization in multi-radio multi-channel wireless networks with directional antennas, in: 2015 IEEE International Conference on Communications (ICC), 2015, pp. 3745–3750. doi:10.1109/ICC.2015.7248907.

[4] F. Glover, M. Laguna, Tabu Search, Kluwer Academic Publishers, 1997.

[5] T. Berthold, Measuring the impact of primal heuristics, Oper. Res. Lett. 41 (6) (2013) 611–614. doi:10.1016/j.orl.2013.08.007.

[6] H. L. Kedane, Comparison of algorithms for the optimization of multi-waveform networks, Master's thesis, Polytechnique Montreal (May 2023).

[7] D. Liu, V. Perreault, A. Hertz, A. Lodi, A machine learning framework for neighbor generation in metaheuristic search, Frontiers in Applied Mathematics and Statistics 9 (2023). doi:10.3389/fams.2023.1128181.