Optimal Fault-Tolerant Placement of Relay Nodes in a Mission Critical Wireless Network

Toni Mancini[1], Enrico Tronci[1], Agostino Scialanca[2], Filiberto Lanciotti[2], Alberto Finzi[3], Riccardo Guarneri[4], and Silvia Di Pompeo[5]

[1] Computer Science Department, Sapienza University of Rome
[2] Log.In s.r.l. Servizi e Sistemi Avanzati per l'Elettronica (Italy)
[3] University of Naples Federico II, Department of Electrical Engineering and Information Technology
[4] Ministero dell'Interno, Dipartimento della Pubblica Sicurezza (Italy)
[5] Roma Capitale, Dipartimento dei Servizi Educativi e Scolastici (Italy)

Abstract. The operations of many *critical infrastructures* (*e.g.*, airports) heavily depend on proper functioning of the radio communication network supporting operations. As a result, such a communication network is indeed a *mission-critical communication network* that needs adequate protection from external electromagnetic interferences. This is usually done through *radiogoniometers*. Basically, by using at least three suitably deployed radiogoniometers and a gateway gathering information from them, sources of electromagnetic emissions that are not supposed to be present in the monitored area can be localised. Typically, *relay nodes* are used to connect radiogoniometers to the gateway. As a result, some degree of fault-tolerance for the network of relay nodes is essential in order to offer a reliable monitoring. On the other hand, deployment of relay nodes is typically quite expensive. As a result, we have two conflicting requirements: minimise costs while guaranteeing a given fault-tolerance. In this paper address the problem of computing a deployment for relay nodes that minimises the relay node network cost while at the same time guaranteeing proper working of the network even when some of the relay nodes (up to a given maximum number) become faulty (*fault-tolerance*). We show that the above problem can be formulated as a Mixed Integer Linear Programming (MILP) as well as a Pseudo-Boolean Satisfiability (PB-SAT) optimisation problem and present experimental results comparing the two approaches on realistic scenarios.

1 Introduction

The operations of many *critical infrastructures* (*e.g.*, airports) heavily depend on proper functioning of the radio communication network supporting operations. As a result, such a communication network is indeed a *mission-critical communication network* that needs adequate protection from external electromagnetic interferences.

For example, in an airport, *ground movement control* is responsible for areas such as: taxiways, inactive runways, holding areas, and some transitional aprons or intersections where aircraft arrive, having vacated the runway or departure gate. In such a case, communication between the air traffic control tower, vehicles and personnel moving in the airport area takes place through VHF/UHF radio. In such a situation, an electromagnetic attack (*e.g.*, through *jamming*) preventing communication between the air traffic control tower and aircraft will degrade the operations of the airport itself.

Accordingly, the area where the mission-critical communication network is deployed is typically *monitored* in order to *detect* spurious electromagnetic emissions and *localise* the source of such emissions (*transmitter*). This is usually done using *radiogoniometers*. Basically, by using at least three suitably deployed radiogoniometers and a gateway gathering information from them, sources of electromagnetic emissions that are not supposed to be present in the monitored area can be localised.

1.1 Motivations

Unfortunately, effective deployment of radiogoniometers is all but easy. In fact, communication between the radiogoniometers and the gateway must use a spectrum that does not interfere with that used within the infrastructure under protection. For this purpose, often communication between radiogoniometers and the gateway rests on an unlicensed spectrum such as WiFi.

This has far-reaching implications. In fact, because of the limitations on the transmission power for WiFi channels, for a large area like a big airport, communication between the radiogoniometers and the gateway is typically not possible with a single hop. Accordingly, *relay* nodes need to be deployed in the area to ensure that radiogoniometers can always communicate with the gateway, possibly through several hops.

Of course in such a situation, reliability of the network of relay nodes becomes essential. This means that relay nodes must be deployed by aiming at achieving a given *fault-tolerance* for the relay network. Namely, we must ensure that even if a maximum given number of relay nodes become faulty, all radiogoniometers are still able to send their information to the gateway.

Installing a relay node entails two kinds of costs. First, that of the devices of which it consists (antennas, amplifiers, etc.) Second, and perhaps more important, the installation cost. The latter may be quite relevant for a critical infrastructure. For example, installing a relay node in an airport may entail installing antennas in a ground movement area.

The above considerations motivate research on effective methods to compute placements for relay nodes that minimise cost while at the same time guaranteeing a given fault tolerance for the network of relay nodes.

1.2 Contributions

In this paper we present Mixed Integer Linear Programming (MILP)–based and Pseudo-Boolean Satisfiability (PB-SAT) optimisation algorithms for optimal and fault-tolerant placement of relay nodes.

The main obstacle to be overcome is the huge number of constraints needed to model the problem. In fact, in order to maximise the transmission distance (without exceeding WiFi transmission power limits) we use directional antennas on relay nodes. As a result, to compute radio-visibility between two network nodes, we need to represent terrain orography. Since the monitored area is of several squared kilometres, we will have many cells to consider in the discrete grid representing orography for the monitored area.

Our algorithm takes as input: the technical (*e.g.*, max transmission power, antenna specifications, etc.) and economical (*e.g.*, cost of devices and installation) characteristics of the relay nodes; the orography of the area to be monitored (*e.g.*, hills, buildings, runways, etc.); constraints on where relay nodes may be installed; desired fault-tolerance F.

Our algorithm returns as output, for each relay node, a routing policy and a placement that minimises the network cost and guarantees that radiogoniometers are connected with the gateway even if F relay nodes become faulty.

We present:

1. Formulations for our MILP as well as PB-SAT optimisation problems;
2. Implementation for our tools;
3. Experimental results comparing the two approaches on a real case study from the geographic area of the Leonardo da Vinci Airport in Rome (Italy).

Our experimental results show that covering an area of several squared kilometres with hundreds of cells generates a MILP problem that can be solved within hours of computation on a commodity PC.

2 State of the art

A survey on node placement in wireless sensor networks is in [37]. None of the methods discussed in [37] focus on minimising network cost while at the same time guaranteeing network connectivity notwithstanding loss (*faults*) of up to a given number of relay nodes. We note that random deployment structurally achieves some degree of fault tolerance, but on one side the network cost is far from being minimal and on the other side no (deterministic) guarantee on the number of faults that the network can withstand can be given *a priori*.

An artificial bee colony algorithm for optimal placement of relay nodes in wireless sensor networks has been presented in [12]. The goal of [12] is that of reducing the communication holes stemming from randomly-placed relay nodes. The results in [12] show that indeed a clever positioning of relay nodes decreases the network energy consumption. Unfortunately, the methods in [12] cannot be directly used in our context since energy is not a concern for us, whereas our goal is to minimise network cost while guaranteeing a given fault tolerance.

The relationship between energy consumption of network relay nodes and routing strategies is studied in [6] as a maximum flow problem. There, fault tolerance is addressed from a statistical point of view: as relay nodes are inexpensive, a sufficiently high number of them is deployed randomly to guarantee high enough probability of the existence of multiple routes from each node to the gateway. On the other hand, in our setting relay nodes are expensive, hence we are required to guarantee fault tolerance under minimum overall placement cost. Also, our nodes are connected to the electrical grid. This means they do not have energy problems, although they must be deployed in areas where connection to the grid is available.

In [9] a 2D model for directional antennas is presented. Although we are also using directional antennas, we need to take into proper account also ground elevation and the presence of obstacles in the area of interest.

Much research work has focused on routing through relay nodes with a known given position (*e.g.*, see [1,32,4,31,8]) or with a given network structure (*e.g.*, see [11,5,34,7,17,36]). Our scenario is quite different, since we have to compute at the same time the position of the relay nodes as well as the routing strategy for each relay node. Note that our solution also needs to define the orientation of the unidirectional antennas hosted by each relay node.

The transmission capacity of ad-hoc networks has been studied in [16]. In our context, such results can be used to model transmission capacity between sensor nodes (*i.e.*, *radiogoniometers*) and gateway nodes.

Modelling of radio communication obstacles in a real environment has been studied in [14]. Such results can be also be used in our context to model static transmission obstacles such as walls, hills, etc.

Network traffic, congestion and interference have been analysed in [15] using a Mixed Integer Linear Programming (MILP)-based approach. Genetic algorithms for optimal node placement with omnidirectional antennas have been studied in [10]. Such results cannot be used in out context since we are using unidirectional antennas.

Optimal positioning of relay nodes in a WiFi network (our setting) has been studied *e.g.*, in [32,3,2]. We note however that [32] does not address fault-tolerant positioning, and [3,2] focuses on energy saving fault tolerant routing of hierarchical networks.

The work in [4], gives up optimality and presents heuristic methods for fault-tolerant positioning of relay nodes. Although we also ask for optimality, we can use some of the considerations in [4] to restrict our search space.

The work in [33] presents methods for optimal and robust (wrt. throughput degradation) positioning of relay nodes. Although in our setting *robustness* in [33] is not the same as fault-tolerance, clearly the work in [33] is aiming towards the same direction as ours.

Summing up, although the are papers addressing optimal positioning of relay nodes as well as papers addressing fault-tolerant positioning of relay nodes, to the best of our knowledge there is no previously published work addressing optimal fault-tolerant positioning of relay nodes.

3 Problem requirements

Our problem is to enable reliable and high-throughput communication between radiogoniometer sensors placed throughout the Monitored Area (MA) of an airport to a gateway, typically located at the air traffic control tower.

As the MA is large (with edges that could be several kilometres long) and because of several logistic constraints, a wired communication network is not viable neither technically nor economically, and wireless communication must be used. Also, given the long distances involved, and the need, due to regulatory

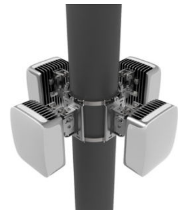

Fig. 1: An array of 4 directional antennas mounted on a pole.

constraints, to use standard WiFi frequencies, *intermediate relay antennas* need to be placed to enable multi-hop communication.

The aim of our *decision support* tool is to compute an *optimal placement* of relay antennas in the MA, in order to create the necessary infrastructure and allow proper routing of the required network traffic from the radiogoniometer sensors (placed near the runways) to the gateway (the air traffic control tower).

Relay antennas are *directional* antennas. Hence, they are deployed in *arrays* mounted on poles, as shown in Figure 1. Antennas mounted on the same pole are pointed to different directions. Each antenna array is positioned at a given elevation L from the ground (the height of the poles).

The problem amounts to decide where in the MA and in which direction to deploy relay antennas in order to satisfy all given requirements and minimising the overall cost due to the devices and their deployment.

In the following sections we describe the high-level problem requirements.

3.1 Monitored Area definition

The MA area typically has an irregular shape and a size of several squared kilometres, going from, *e.g.*, the runways to the air traffic control of an airport. Also the terrain orography is irregular: different regions of the MA might show different ground elevations, and obstacles of various kinds (*e.g.*, buildings like terminals or hangars, etc.) might exist.

Radio visibility between points in the MA cannot be determined by simply considering their Euclidean distance, because obstacles (*e.g.*, buildings) and the terrain orography (*i.e.*, different elevation) must be considered as well. To this end, we need to wisely model the *terrain orography* for our scenario.

Terrain orography information is stored in available databases. In particular, in our database, the MA is partitioned into *convex polygons* and a *terrain elevation* value is associated to each such polygon.

3.2 Forbidden placement and link areas

Forbidden placement areas define those portions of the MA in which no relay node can be placed. *Forbidden link areas* define those portions of the MA that should not be traversed by radio transmission links.

Forbidden placement areas stem from the existence of runways, taxiways, holding areas, etc. in the MA in which no antenna can be installed, as well as other requirements that impose that some regions are kept clear, due to technical or regulatory constraints. Forbidden link areas stem from, *e.g.*, technical or safety requirements (for example, to avoid interference with other radio networks in the airport).

Both forbidden placement and forbidden link areas are defined in our database as a set of *convex polygons* lying within the MA.

3.3 Deployment costs

The cost of placing a pole to enable installation of an antenna array is not constant throughout the MA, as it might depend on several parameters (*e.g.*, the accessibility of the chosen point for maintenance).

Our database defines such costs by associating pole placement costs to *convex polygons* contained in the MA. On the other hand, once a support pole has been deployed, installing a single directional antenna on it has a fixed cost.

3.4 Radio visibility requirements

Radio visibility requirements state the conditions that must be met by the placement of pairs of directional relay antennas for their network links to work satisfactorily.

In particular, radio communication between two (properly-oriented) antennas in points (x, y) and (x', y') of the MA (at elevation points h and h' as stated by the terrain orography and the height of the holding poles) is considered impossible/unsatisfactory not only if the 3D Euclidean distance between them is longer than a given threshold, but also if the relevant Fresnel Ellipsoid (FE) computed for the two antennas intersects obstacles (due to, *e.g.*, terrain elevation changes, buildings, etc.).

Given two antennas deployed in two points (possibly at different elevations), the relevant FE associated to them is a 3D ellipsoid whose major radius is the straight-line segment connecting the two points, and whose minor radius is the greatest value such that, if a stray component of the transmitted signal bounces off an object within the FE and then arrives at the receiving antenna, the resulting phase shift will be considered to have an unacceptably negative impact on the signal quality. Given the transmitting/receiving power of the employed antennas, their maximum transmission distance and the band used, the value of the minor radius of the relevant FE for the relay placement task at hand can be computed with specialised algorithms.

3.5 Fault tolerance requirements

Fault tolerance requirements ask that the network to be designed guarantees successful routing from each sensor to the gateway also in case of malfunctioning of at most $F \geq 1$ relay nodes.

To guarantee that fault tolerance requirements are satisfied, the computed relay node placement is required to be such that at least $F + 1$ distinct routes from each sensor to the gateway exist, and that no two distinct routes from the same sensor share a relay node.

3.6 Performance requirements

Performance requirements ask that the network to be designed ensures that communication from each sensor to the gateway is not significantly impacted by degradation effects, which arise when using too many relay nodes.

To guarantee that performance requirements are met, it is requested that each sensor-to-gateway route in the designed network consists of at most $H \geq 1$ hops (*i.e.*, traverses at most $H - 1$ intermediate relay nodes).

3.7 Capacity requirements

Network capacity constraints ask that the network is able to convey the needed traffic from all the sensors to the gateway. This means that there is an upper bound to the number of messages (from sensor nodes) that can be routed through a given radio link (*i.e.*, a pair of antennas mounted on different poles each one lying in the radio visibility cone of the other). This limits depend on the transmission band available on radio links and on the band needed to transmit a single message from a sensor.

4 Computing optimal fault-tolerant relay placement

The complex problem requirements outlined in Section 3 make it impossible to immediately use standard off-the-shelf Artificial Intelligence (AI) reasoners. In particular, requirements that ask to consider the terrain orography and the presence of obstacles and radio-visibility conditions between pairs of relay antennas demand for a complex preprocessing of the source data before giving it as input to a standard AI reasoner to solve the problem.

In this section we briefly describe how our algorithm, by querying the external data sources defining the scenario at hand (*i.e.*, the geometry and orography of the Monitored Area, MA, and the forbidden placement and link areas), and by taking all the other user parameters as input, processes such data by defining multiple computational geometry problems similar to those addressed in [18], which are solved using a Mixed Integer Linear Programming (MILP) solver. After this preprocessing is completed, standard AI reasoners can be used to solve our main problem. In this paper we experiment with off-the-shelf MILP and Pseudo-Boolean Satisfiability (PB-SAT) solvers.

4.1 Input data

From external sources, the following data are available for the scenario at hand:

1. The geometry and terrain orography of the MA, in terms of a set of convex polygons, each one associated with an elevation value (in meters).
2. The set of areas in the MA where no relay node can be placed (forbidden placement areas), in terms of a set of convex polygons.
3. The set of areas in the MA that cannot be traversed by radio links (forbidden link areas), in terms of a set of convex polygons.

4. A map defining the placement cost of a relay node throughout the MA, again in terms of a set of convex polygons, each one associated with a cost value (in Euro).
5. The number and positions of the sensor nodes (radiogoniometers) in the MA.
6. The position of the gateway node in the MA.

Moreover, our algorithm takes the following additional inputs:

7. The maximum transmission rate of relay antennas (all equal).
8. The transmission rate of each sensor.
9. The maximum distance D between two relay nodes for their direct radio link to be reliable.
10. The value r of the minor radius of the Fresnel Ellipsoid (FE) to be considered when performing quality assessments of the potential radio links.
11. The maximum number of allowed hops H for all sensors-to-gateway routes.
12. The maximum number of relay node faults F that the network must tolerate.
13. The maximum number of antennas that can be installed in a relay node.
14. The finite set of possible orientations O of antennas in a relay node, in terms of a set of values in degrees from a fixed direction (*e.g.*, North).
15. The fixed cost of installing a single antenna in a (already existing) relay node.

Finally, in order to make the set of candidate relay antenna placements finite, our algorithm considers the MA partitioned in rectangular cells. Each relay node will be placed in a distinct cell. The user is required to provide the desired number of such cells, in terms of:

16. I and J (positive integers), the number of cells in which area A must be split along the X and Y axes, respectively.

4.2 Monitored Area discretisation

The geometry of the MA is over-approximated by a rectangular bounding box A. A suitable Cartesian orthogonal coordinate system is defined as to make the edges of rectangle A parallel to the X and Y axes. As a result of this coordinate system transformation, area A is defined by points having X and Y coordinates ranging within $[0, X_{\max}]$ and $[0, Y_{\max}]$ resp., for some $X_{\max}, Y_{\max} \in \mathbb{R}^+$. From now on, we assume that whenever we need to use geographic coordinates, they will be first transformed accordingly into coordinates of the above system. Any region in A not belonging to the MA will be managed by defining *forbidden placement* (akin to *unadmissible scenarios* in [24]) and link areas in A (see Section 3.2).

Area A is then partitioned in $I \times J$ *cells*, each one being a rectangle having size W_X by W_Y (in metres), where $W_X = \frac{X_{\max}}{I}$ and $W_Y = \frac{Y_{\max}}{J}$.

Cells are denoted by $C_{i,j}$, for $i \in [0, I-1]$ and $j \in [0, J-1]$. Hence, for each i, j, $C_{i,j}$ defines the rectangle consisting of points:

$$C_{i,j} = \{(x, y) \in A \mid iW_X \leq x \leq (i+1)W_X,\ jW_Y \leq y \leq (j+1)W_Y\}.$$

4.3 Relational database population

The first step of our algorithm is to run helper feasibility and optimisation MILP problems in order to populate a relational database, starting from the input data about terrain orography and pole placement costs. This pre-processing follows the same strategy used in [20] to generate admissible (simulation) scenarios through a pre-processing, basically computing all feasible solutions to a constraint problem.

We proceed as follows. For each cell $C_{i,j} \in A$ we first compute whether it intersects a forbidden antenna placement area. This is done by solving a simple feasibility MILP problem defined over the linear constraints stemming from the cell definition and those stemming from each of the forbidden placement area convex polygons. The MILP problem also defines one boolean decision variable per such polygon. A feasible solution is enforced to set at least one of the boolean variables to true, which in turn imposes that at least one forbidden placement area polygon intersects the cell at hand.

MILP problems as the one outlined above are very easy to solve and define our general approach to check, using standard MILP technology, whether a polygon (or, in some cases, a 3D polyhedron) intersects a union of polygons (3D polyhedra). This is done using the same MILP based approach used in [29] to generate control abstractions.

For cells $C_{i,j}$ not intersecting a forbidden placement area (hence, whose associated MILP problem is *infeasible*), we also compute: (i) The minimum elevation of a point in $C_{i,j}$; (ii) The maximum pole placement cost within $C_{i,j}$. For such computations, two additional (very simple) optimisation MILP problems are generated and solved. Constraints of such problems are similar to those of the feasibility MILP problem described earlier to detect intersection between a convex polygon and the union of convex polygons. In this case, also an objective function is defined, which searches for the intersection point having minimum elevation or maximum cost respectively.

At the end of this pre-processing stage, much as in [30], our relational database is populated with relation *Cell*, whose tuples (i, j, e, c) define the cells $C_{i,j}$ covering A not intersecting a forbidden placement area together with their minimum elevation value and maximum antenna placement cost (positive reals).

4.4 Radio visibility graph computation

The next step performed by our algorithm is to compute the radio visibility graph G of A using an approach similar to that used in [22,25,21] to generate sets of simulation scenarios. The radio visibility graph is an undirected graph $G = (V, E)$ where:

- The set of nodes V is the set of possible placements of poles within cells of A (*pole nodes*) and possible installations of directional antennas on such poles (*antenna nodes*).
- The set of edges E is the set of feasible and reliable communication links that could be established between two antenna nodes installed on different poles and the communication links between each pole and antennas installed on it.

In order to compute the radio visibility graph G, a set of helper feasibility/optimisation MILP problems are solved using data from the input sources.

Computation of nodes The set of nodes V of G is computed, much as admissible scenarios are computed in [23], starting from the set of cells $C_{i,j}$ already proved not to intersect a forbidden placement area (hence such that the pair (i, j) occurs in relation *Cell* of our database, Section 4.3).

For each such cell, $|O| + 1$ nodes of G are generated: one node (*pole node*) representing the potential installation of a pole in that cell and $|O|$ nodes (*antenna nodes*) representing the potential installation of a single antenna mounted on the pole installed in the cell and with each possible orientation.

More in detail, for cell $C_{i,j}$, the following nodes are generated:

- $|O|$ *antenna nodes* of the form (i, j, o), one for any possible antenna orientation $o \in O$. The cost of installing a single antenna on an already deployed pole is constant and given as input (Section 4.1).
- A single *pole node* of the form (i, j), representing the installation of a pole in that cell. For each pole node, the maximum cost for installing a pole within its cell is available in relation *Cell* of our database (Section 4.3).

Computation of edges Each edge defined in the radio visibility graph G will represent a possible communication link.

First, an edge is generated connecting each pole node with any antenna node in the same cell, thus defining the possibility of a (wired) direct connection between each single antenna and the pole on which it is mounted (and hence, indirectly, with all the other relay antennas mounted on the same pole).

Second, for each pair of antenna nodes in different cells, an edge is generated if there is good radio visibility between the two antennas (considering their respective orientations). Algorithm 1 outlines the algorithm used to check if radio visibility between antenna nodes (i, j, o) and (i', j', o') is satisfactory or not. The algorithm first computes the centre positions of the given cells and the minimum heights of the antennas that could be placed on it and then returns true if and only if: (i) the distance between the two positions is not greater than the maximum transmission distance D; (ii) the two antennas (given their orientations) are visible to each other; (iii) no interference is expected between the two antennas and no forbidden link area is traversed by the prospective radio link. Double visibility between the two (oriented) antennas is delegated to function *double_visibility*() that uses standard trigonometry-based computations to check if the second antenna is within the visibility cone of the first one and vice-versa. As antennas, although, directional, might have a non-negligible visibility cone, parallel radio links can be established between distinct antennas belonging to the same pair of poles.

Computation of absence of interference is delegated to function *radio_no_interference*() which also takes into account the height of the holding poles L. Along the lines of the MILP based over-approximation approaches in [26,27,13], function *radio_no_interference*() computes the 3D bounding box of the FE with minor radius r in terms of linear constraints (hence a safe over-approximation of it), and solves additional helper feasibility MILP problems (along the lines of those outlined in Section 4.3) to check whether such a bounding box intersects at least one polyhedron defined from the terrain orography data or whether its XY projection overlaps a forbidden link area polygon.

The pseudo-code of these functions is omitted for space reasons.

```
function radio_visibility( (i,j,o), (i',j',o'))
  x ← W_X ∗ (i + 1/2);
  y ← W_Y ∗ (i + 1/2);
  h ← min. elevation of cell C_{i,j} as stored in relation Cell;
  x' ← W_X ∗ (i' + 1/2);
  y' ← W_Y ∗ (i' + 1/2);
  h' ← min. elevation of cell C_{i',j'} as stored in relation Cell;
  return sqrt((x' − x)^2 + (y' − y)^2 + (h' − h)^2) ≤ D and
        double_visibility((x, y, o), (x', y', o')) and
        radio_no_interference((x, y, h + L), (x', y', h' + L))
```

Algorithm 1: Checking feasibility of a radio link between to antennas.

4.5 The main MILP and PB-SAT optimisation problems

Thanks to the preprocessing outlined in the previous sections, our radio visibility graph already provides a suitable representation of requirements about forbidden placement and link areas (Section 3.2) and radio visibility (Section 3.4).

Our main optimisation problem described next will decide where to deploy antenna arrays (over holding poles placed in the centre of some of the cells defined as pole nodes in the radio visibility graph) and the number and orientations of the associated antennas, enforcing the remaining requirements (*i.e.*, fault tolerance, performance and capacity requirements, Sections 3.5 to 3.7), while minimising overall deployment costs (as required by the requirement in Section 3.3).

We decided to model our main optimisation problem both as a MILP and a PB-SAT problem and compare the performance of off-the-shelf solvers of both technologies on a set of real case studies. It turns out that the problem specifications in the two paradigms is quite similar, as a natural choice for our MILP problem is to define only 0-1 decision variables. Such 0-1 variables can be immediately converted into Boolean variables in our PB-SAT specification. Also, thanks to the preprocessing computation of the radio visibility graph, the requirements not already handled can be quite conveniently defined as linear constraints.

Below we outline our main modelling ideas, which are common to the two solving technologies.

Variables For each sensor s and for each of the $F+1$ distinct routes r from s to the gateway (as required by the fault tolerance constraints), the following set of 0-1 (as for MILP) or Boolean (as for PB-SAT) variables is defined: $\{x_{s,r,i,e} \mid i \in$

$[1, H], e \in E_{\text{ant}}\}$, where E_{ant} is the subset of edges in the radio visibility graph connecting two antenna nodes. In a solution, assignment 1 to each variable $x_{s,r,i,e}$ denotes that the r-th route from sensor s to the gateway uses the radio link e for its i-th hop.

Additional (redundant) 0-1/Boolean variables are defined to ease the specification of some of the constraints. Such additional variables proved convenient both for the MILP and the PB-SAT specifications. For example, additional variables are introduced to denote the set of poles and antennas that need to be deployed (their values functionally depends on the assignment to the $x_{s,r,i,e}$ variables defined above). Suitable channelling constraints are added in both specifications to keep the values of such variables in sync with those of the variables on which they depend.

Constraints The MILP and the PB-SAT problems have several constraints, generated starting from the radio visibility graph and the other inputs. Although modelled differently, constraints in both specifications can be classified in the following common high-level way:

Routes' connectedness constraints For each sensor s, its $F + 1$ required routes to the gateway are made of a sequence of adjacent edges of E_{ant}. Connectedness constraints also enforce the use of the proper antenna-to-pole and pole-to-antenna edges for routes traversing two antennas installed on the same pole (such explicit links are needed to enforce capacity constraints).

Fault tolerance constraints For each sensor s, its $F + 1$ required routes to the gateway do not share any relay node.

Number of antennas per relay node The number of antennas that need to be deployed on each pole does not exceed the upper limit given as input.

Number of antennas at the gateway The number of antennas that need to be deployed at the gateway does not exceed the upper limit given as input.

Network flow and capacity constraints These constraints model a flow problem in the graph induced by the computed set of routes, ensure that each relay node has sufficient transmission capacity to handle the overall network traffic passing through it and that the necessary flow constraints are satisfied.

Objective function The (linear) objective function for both our MILP and PB-SAT specifications asks to minimise the overall placement cost of the relay node, which is given by the sum of the cost of placing each pole (these costs depend on the cell where each pole if deployed and have been computed during preprocessing), plus the cost of installing all the required antennas on the deployed poles (the cost of installing each antenna is fixed). An additional part in the objective function is defined in order to minimise also the overall number of radio links used. A proper weighting of this last part of the objective function avoids the risk that a solution is computed having non-minimal cost (but envisioning a lower number of used radio links).

5 Experimental Results

In this section we present our experimental results. We considered 8 realistic relay node placement scenarios in various areas of the "Leonardo da Vinci" Fiumicino airport (Rome, Italy). Table 1 describes our scenarios in terms of:

1. the size of the Monitored Area (MA) in meters (columns $X_{\max}$ and $Y_{\max}$);
2. the size of the (always square) cells in meters (column $W_X = W_Y$);
3. the resulting overall number of cells (column "#cells");
4. the maximum number H of hops of each sensor-to-gateway route (column "#hops");
5. the maximum number of relay node faults F that the network must tolerate (column "#toler. faults");
6. the maximum transmission distance D between two pairs of antennas in meters (column "max tx dist.")

All scenarios have been defined with using following constant values for the parameters not mentioned in Table 1:

7. minor radius r of the Fresnel Ellipsoid (FE): 4 meters;
8. maximum number of antennas per node (sensor, relay or gateway): 6;
9. possible antenna orientations: $O = \{\text{N, NE, E, SE, S, SW, W, NW}\}$;
10. cost for a directional antenna installation on an existing pole: Eur 50.00;

The 8 scenarios above have been solved on a personal computer equipped with an Intel Core 2duo Processor P8400@2.26 GHz and 3GB RAM, using both the Glpk Mixed Integer Linear Programming (MILP) solver and the MiniSat+ Pseudo-Boolean Satisfiability (PB-SAT) solver. For each scenario, Table 1 reports the computation time of the main optimisation problem using both solvers and the objective value of the optimal solution (minimum relay node placement cost). Time for performing the preprocessing phase, dominated by the creation of the radio visibility graph, is not reported, as this phase is identical for both solving technologies. Although the preprocessing time can be substantial (≤ 1 hour in the worst case), this is not a critical issue, because:

1. Preprocessing is a one-time task and does not need to be repeated if relay node placement costs or other problem parameters like the required fault tolerance, network capacity and performance requirements change.
2. Preprocessing is amenable to massive (*embarrassing*) parallelisation. In particular, the helper MILP problems are independent of each other, and the computation speed of the radio visibility graph can be increased linearly with the number of available computation cores (hence, reduced to very low values with a reasonably small number of processors).

Table 1 shows that MiniSat+ (PB-SAT) is always outperformed by Glpk (MILP). However, it is worth noting that, thanks to the wise preprocessing phase outlined in Section 4, we were able to define our optimisation problem specification in a natural way, using modelling patterns very well known in the MILP solving community. The PB-SAT specification has been derived by straightforward translating the MILP specification, with no specific optimisations explicitly thought to boost the performance of the PB-SAT solver. Notwithstanding this, MiniSat+ was actually able to solve 5 out of our 8 complex instance scenarios in very reasonable time.

id	scenario definition							Glpk		MiniSat+	
	$X_{\max}$ (m)	$Y_{\max}$ (m)	$W_X = W_Y$ (m)	#cells	#hops (H)	#toler. faults (F)	max tx dist. (m) (D)	CPU time (min)	depl. cost (Eur)	CPU time (min)	depl. cost (Eur)
1	50	50	10	25	6	1	16	0.02	7100	0.30	7100
2	3660	1270	100	465	5	1	400	–	–	–	–
3	3660	1270	200	117	5	1	400	0.31	12600	6.41	12600
4	3590	1560	200	141	5	1	400	2.96	9750	33.15	9750
5	3500	1500	200	132	5	1	400	7.25	10550	51.97	10550
6	4000	3600	200	360	5	1	1000	–	–	–	–
7	1500	1500	100	225	2	1	120	0.01	2100	0.07	2100
8	3660	1070	150	175	5	1	350	1146	9800	–	–

Table 1: Experimental results ("–": time-out occurred after one week).

6 Conclusions

We addressed the problem of computing a deployment for relay nodes in a wireless network that minimise the relay node network cost while at the same time guaranteeing proper working of the network even when some of the relay nodes (up to a given max number) become faulty (*fault-tolerance*).

For such a problem we presented a Mixed Integer Linear Programming (MILP) as well as a Pseudo-Boolean Satisfiability (PB-SAT) formulation along with experimental results comparing the two approaches on realistic scenarios.

Our experimental results show that with less than one day of computation we can position with a precision of a few hundred meters relay nodes in areas of about 5 squared kilometres. Furthermore, thanks to our pre-processing, even though the problem formulation was designed for a MILP solver, the MiniSat+ PB-SAT solver was able to solve most of the problem instances we considered within reasonable time.

Scenarios involving areas partitioned in more than 300 cells seem to be particularly challenging for both our approaches. In such cases, a possibility is to relax the completeness/optimality constraints (along the lines of, *e.g.*, [19], possibly keeping statistical guarantees of coverage as in, *e.g.*, [35,28]) in order to converge to a (possibly sub-optimal) solution in reasonable time.

Acknowledgements. This work was partially supported by the Italian Ministry of University and Research (MIUR) under grant "Dipartimenti di eccellenza 2018–2022" of the Department of Computer Science of Sapienza University of Rome and by the EC FP7 project SmartHG (Energy Demand Aware Open Services for Smart Grid Intelligent Automation, 317761).

References

1. A.D. Amis, R. Prakash, T.H.P. Vuong, and D.T. Huynh. Max-min d-cluster formation in wireless ad hoc networks. In *Proc. IEEE INFOCOM 2000*, volume 1, pages 32–41, 2000.

2. M.D. Azharuddin, P. Kuila, and P.K. Jana. Energy efficient fault tolerant clustering and routing algorithms for wireless sensor networks. *Computers & Electrical Eng.*, 41:177–190, 2015.
3. A. Bari, A. Jaekel, J. Jiang, and Y. Xu. Design of fault tolerant wireless sensor networks satisfying survivability and lifetime requirements. *Computer Comm.*, 35(3):320–333, 2012.
4. E.S. Biagioni and G. Sasaki. Wireless sensor placement for reliable and efficient data collection. In *Proc. HICSS 2003*, page 127.2. IEEE Comp. Soc., 2003.
5. J. Cao, G. Wang, M.Z. Bhuiyan, and J. Wu. Deploying wireless sensor networks with fault-tolerance for structural health monitoring. *IEEE Trans. on Computers*, 2013.
6. M. Cardei and D.-Z. Du. Improving wireless sensor network lifetime through power aware organization. *ACM Wireless Netw.*, 11, 2005.
7. A. Chelli, M. Bagaa, D. Djenouri, I. Balasingham, and T. Taleb. One-step approach for two-tiered constrained relay node placement in wireless sensor networks. *IEEE Wireless Comm. Letters*, 5:448–451, 2016.
8. F. Deniz, H. Bagci, I. Korpeoglu, and A. Yazıcı. An adaptive, energy-aware and distributed fault-tolerant topology-control algorithm for heterogeneous wireless sensor networks. *Ad Hoc Networks*, 44:104–117, 2016.
9. D. Ganesan, R. Cristescu, and B. Beferull-Lozano. Power-efficient sensor placement and transmission structure for data gathering under distortion constraints. In *Proc. IPSN 2004*, pages 142–150. ACM Press, 2004.
10. S.K. Gupta, P. Kuila, and P.K. Jana. Genetic algorithm approach for k-coverage and m-connected node placement in target based wireless sensor networks. *Computers & Electrical Eng.*, 56:544–556, 2016.
11. X. Han, X. Cao, E.L. Lloyd, and C.-C. Shen. Fault-tolerant relay node placement in heterogeneous wireless sensor networks. *IEEE Trans. Mobile Computing*, 9(5), 2010.
12. H.A. Hashim, B.O. Ayinde, and M.A. Abido. Optimal placement of relay nodes in wireless sensor network using artificial bee colony algorithm. *Journal Netw. Comp. Appl.*, 64:239–248, 2016.
13. B.P. Hayes, I. Melatti, T. Mancini, M. Prodanovic, and E. Tronci. Residential demand management using individualised demand aware price policies. *IEEE Trans. Smart Grid*, 8(3), 2017.
14. A. Krause, C. Guestrin, A. Gupta, and J.M. Kleinberg. Near-optimal sensor placements: Maximizing information while minimizing communication cost. In *Proc. IPSN 2006*, pages 2–10. ACM, 2006.
15. Y. Lee, K. Kim, and Y. Choi. Optimization of AP placement and channel assignment in wireless LANs. In *Proc. IEEE LCN 2002*. IEEE, 2002.
16. J. Li, C. Blake, D.S.J. De Couto, H.I. Lee, and R. Morris. Capacity of ad hoc wireless networks. In *Proc. MobiCom 2001*, pages 61–69. ACM Press, 2001.
17. R. Magán-Carrión, R.A. Rodríguez-Gómez, J. Camacho, and P. Garcìa-Teodoro. Optimal relay placement in multi-hop wireless networks. *Ad Hoc Networks*, 46:23–36, 2016.
18. T. Mancini. Now or Never: Negotiating efficiently with unknown or untrusted counterparts. *Fundam. Inform.*, 149(1-2):61–100, 2016.
19. T. Mancini, P. Flener, and J. Pearson. Combinatorial problem solving over relational databases: View synthesis through constraint-based local search. In *Proc. SAC 2012*, pages 80–87. ACM, 2012.
20. T. Mancini, F. Mari, A. Massini, I. Melatti, F. Merli, and E. Tronci. System level formal verification via model checking driven simulation. In *Proc. CAV 2013*, volume 8044 of *LNCS*, pages 296–312. Springer, 2013.

21. T. Mancini, F. Mari, A. Massini, I. Melatti, I. Salvo, and E. Tronci. On minimising the maximum expected verification time. *IPL*, 122:8–16, 2017.
22. T. Mancini, F. Mari, A. Massini, I. Melatti, and E. Tronci. Anytime system level verification via random exhaustive hardware in the loop simulation. In *Proc. DSD 2014*, pages 236–245. IEEE, 2014.
23. T. Mancini, F. Mari, A. Massini, I. Melatti, and E. Tronci. System level formal verification via distributed multi-core hardware in the loop simulation. In *Proc. PDP 2014*, pages 734–742. IEEE, 2014.
24. T. Mancini, F. Mari, A. Massini, I. Melatti, and E. Tronci. SyLVaaS: System level formal verification as a service. In *Proc. PDP 2015*, pages 476–483. IEEE, 2015.
25. T. Mancini, F. Mari, A. Massini, I. Melatti, and E. Tronci. Anytime system level verification via parallel random exhaustive hardware in the loop simulation. *Microprocessors and Microsystems*, 41:12–28, 2016.
26. T. Mancini, F. Mari, I. Melatti, I. Salvo, E. Tronci, J. Gruber, B. Hayes, M. Prodanovic, and L. Elmegaard. Demand-aware price policy synthesis and verification services for smart grids. In *Proc. SmartGridComm 2014*, pages 794–799. IEEE, 2014.
27. T. Mancini, F. Mari, I. Melatti, I. Salvo, E. Tronci, J.K. Gruber, B. Hayes, M. Prodanovic, and L. Elmegaard. User flexibility aware price policy synthesis for smart grids. In *Proc. DSD 2015*, pages 478–485. IEEE, 2015.
28. T. Mancini, E. Tronci, I. Salvo, F. Mari, A. Massini, and I. Melatti. Computing biological model parameters by parallel statistical model checking. In *Proc. IWBBIO 2015*, volume 9044 of *LNCS*, pages 542–554. Springer, 2015.
29. F. Mari, I. Melatti, I. Salvo, and E. Tronci. Synthesis of quantized feedback control software for discrete time linear hybrid systems. In *Proc. CAV 2010*, pages 180–195. Springer, 2010.
30. F. Mari, I. Melatti, I. Salvo, and E. Tronci. Model-based synthesis of control software from system-level formal specifications. *ACM Trans. Softw. Eng. Meth.*, 23(1):6:1–6:42, 2014.
31. S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M.B. Srivastava. Coverage problems in wireless ad-hoc sensor networks. In *Proc. IEEE INFOCOM 2001*, pages 1380–1387 vol.3, 2001.
32. R. Prasad and H. Wu. Gateway deployment optimization in cellular wi-fi mesh networks. *Journal of Netw.*, 1(3):31–39, 2006.
33. L. Qiu, R. Chandra, K. Jain, and M. Mahdian. Optimizing the placement of integration points in multi-hop wireless networks. In *Proc. IEEE ICNP 2001*. IEEE, 2004.
34. L. Sitanayah, K.N. Brown, and C.J. Sreenan. A fault-tolerant relay placement algorithm for ensuring k vertex-disjoint shortest paths in wireless sensor networks. *Ad Hoc Networks*, 23:145–162, 2014.
35. E. Tronci, T. Mancini, I. Salvo, S. Sinisi, F. Mari, I. Melatti, A. Massini, F. Davì, T. Dierkes, R. Ehrig, S. Röblitz, B. Leeners, T. H. C. Krüger, M. Egli, and F. Ille. Patient-specific models from inter-patient biological models and clinical records. In *Proc. FMCAD 2014*, pages 207–214. IEEE, 2014.
36. C.M. Wei Liang, M. Zheng, and H. Sharif. A connectivity-aware approximation algorithm for relay node placement in wireless sensor networks. *IEEE Sensors Journal*, 16, 2016.
37. M. Younis and K. Akkaya. Strategies and techniques for node placement in wireless sensor networks: A survey. *Ad Hoc Networks*, 6(4):621–655, 2008.