Heuristic Algorithm for Gateway Location Selection in Large Scale LoRa Networks

Krzysztof Grochla, Konrad Połys

Abstract—**The LoRa networks allow building low-cost long-range networks. The selection of LoRa access point locations in large scale networks requires taking into consideration the spatial distribution of clients, radio signal propagation and the limitation of the number of devices served by a single point of access. We propose a heuristic algorithm for the selection of access point locations within for a given set of candidate locations, based on gradient optimization. The proposed method allows selecting a sub-optimal set of locations which provide full coverage and take into consideration the capacity dimensioning based on spreading factor and expected channel utilization. The algorithm is evaluated in random topologies and in real-life scenario based on city-wide smart meters locations. The analysis shows that it outperforms a manual selection of access points locations, allowing to decrease the number of access points providing the full coverage.**

Index Terms—**LoRa, Capacity, Internet of Things, dimensioning, radio planing**

I. Introduction

The introduction of Low Power WAN network concept, with devices communicating over distances of tens of kilometres using low power radio interface, has generated a multitude of novel use cases based on long life battery-powered devices. The LoRa is one of the most widely adopted LP WAN standards, which is based on ISM frequencies and chirp spectrum modulation. The LoRa communication now is being deployed in devices such as smart meters, sensors or actuators executing smart city functions[1]. The LoRa can provide cheap and very energy efficient communication with thousands of devices per one access point, at the cost of low data rate. To facilitate the deployment of LoRa devices, the LoRaWAN standard has been proposed, which defines the packet format and communication architecture allowing to forward the information from the devices to the internet servers.

LoRa is a wireless data transmission technology. It allows to transmit data on long distances with low battery usage, but the data rate is relatively low. LoRa suits for different sensor devices. In this technology modulation of electromagnetic waves is called CSS – Chirp Spread Spectrum. So far it was used in military and space industry because its key feature is long range and high interference robustness. To orthogonalize transmissions LoRa is not bonded to a particular carrier frequency, bandwidth, coding rate and spreading factor, however, it is typically deployed in industrial, scientific and medical (ISM) bands: 868 Mhz in Europe and 902 MHz in US.

Krzysztof Grochla and Konrad Połys were with the Institute of Theoretical and Applied Informatics Polish Academy of Sciences, 5 Bałtycka Str., 44-100 Gliwice, Poland, e-mail: [kgrochla|kpolys]@iitis.pl

Besides LoRa modulation there is a LoRaWAN protocol[2]. LoRaWAN is a Media Access Protocol (MAC) which is developed to achieve a high efficiency using LoRa as a lower layer. This protocol is bi-directional which allows receiving a confirmation packet. LoRaWAN also supports data encryption, a device to network registration and multicast data transmission. This protocol was developed to provide compatibility with all LoRaWAN devices. The architecture of LoRaWAN consist of four main components:

- end device
- gateway
- application server
- network server

An end device (end node) is a small computing power device equipped with a radio interface and usually with a sensor or actuator. Such node is able to send and receive data and are battery powered. When a node transmits data to network it's called an uplink, when the network sends packets to node it's downlink. The LoRa gateways, also called access points, are devices which listen for the packets sent by end devices. The received data packets are transmitted as IP packets to a network server. The gateways also handle downlink traffic to the end devices. A network server is responsible for packet forwarding to applications, which gateway is the best candidate to send a downlink data to a node, removing duplicated messages, decrypting data. An application server is an end where data from a node are processed.

The LoRa devices use different Spreading Factor (SF) depending on the radio signal propagation conditions. The SF value can be between 7 and 12 and the higher SF means the higher transmission time but greater resiliency to interferences. The SF can be selected automatically, by an Adaptive Data Rate (ADR) algorithm [3] or can be configured manually to a constant value.

A. Gateway Location Selection

The problem of gateways' location selection in LoRa networks is similar to the radio network planning problem in cellular networks. Traditionally, this was solved by experienced engineers, with the use of radio signal propagation modelling software, digital maps and drive tests for selected locations [4], [5]. This process was both time consuming and costly. Because of the low cost of LoRa gateways, it is more cost-efficient to place more devices than use complex solutions to select the proper location for a gateway. Now it makes a need for tools supporting gateway location selection that are more simplified and easier to use. The number and locations of gateways need to be adjusted not only to allow the reception of LoRa signal within the predefined area, but also need to be adjusted to the endpoints' density, as more end nodes generate more traffic which needs to be served. In LoRaWAN the ALOHA channel access is used, thus the main limiting factor of the number of nodes is the packet collision probability. The increase of the number of gateways in an area allows using lower spreading factor, which shortens the transmission time and lowers the collision probability, increasing the network capacity. However, radio planning is well known problem but in the context of LoRa network it is still not well described and needs to be further investigated.

B. Problem formulation

In this work, we analyze the problem of capacity planning and selection of location for LoRa network access point. We propose an algorithm to select the suboptimal location for the LoRa gateway location for a given set of endpoint location. The gateway location is selected from a set of location candidates, which may be the same as the set of end node location. The selection of a set of gateways must meet the requirement to provide connectivity to all end nodes. Assuming we have a connectivity graph estimated from the distances between the nodes and the radio signal propagation model, this may be seen as a problem of selection of a dominating set, which is an NP-complete decision problem in computational complexity theory[6]. Thus the problem is complex and it is hard to solve the capacity planning jointly with the location selection, we divide the problem into two parts: first we discuss the maximum number of nodes that can be served by a gateway, next we propose an algorithm to select gateways locations which sustain the condition that each gateway is serving at maximum previously selected number of nodes.

There are a few differences between other popular radio networks (e.g. LTE) and LoRa. LoRa end devices are typically battery powered devices so they transmit data rarely, but usually in a constant time window. LoRa uses ALOHA media access control so collisions are more probable. LoRa devices also use all possible channels uniformly, there is no "colouring" of areas and LoRa uses spreading factors, thus the capacity of the gateway may vary depending on the spatial distribution of the nodes. We assume the set of client locations and the traffic characteristic is known and is an input for the proposed algorithm.

The rest of the paper is organized as follows: the second section describes the state of the art regarding the planning and dimensioning of LoRa networks and the selection of AP location. Next, we describe the network model used in the study together with the analysis of the maximum number of LoRa devices serviced by a single gateway. In the following section, we show the pseudocode of the proposed optimization algorithm. Next we present the analysis of the performance of the algorithm in comparison to the selection based on a regular placement of nodes. We finish the paper with a short conclusion.

II. State of the Art

The resiliency of the chirp spread spectrum to the interferences has been thoughtfully analyzed in [7]. This work provides a quantitative confirmation of the general rules regarding recommendable settings spreading factors, bandwidth and coding rate as a function of distance and interferences. It provides a good basis for the LoRa capacity estimation. The coverage and capacity of LoRa network basing on real life deployment are also discussed in [1], however the results are based on small number of experiments. The analysis of the scalability of the LoRa technology and the determination of the maximum number of sensors that can be served by an access point is given in [8]. This paper shows calculation of the packet delivery probability for given LoRa network configuration, basing eg. on the payload size or number of LoRa nodes. The analysis is based on fixed percentage of nodes per each spreading factor, which is true only for a dense networks with regular placement of end devices.

The radio network planning has been investigated since more than twenty years, however most of the work is dedicated to cellular networks - see e.g. [4] or [5]. However, there is very little research results available considering the specifics of the LP WAN radio network planning, where the cost of network deployment is much smaller, thus it does not justify the costly drive tests and radio signal propagation analysis. One of the few tools which allow to estimate the coverage of a LoRa gateway on a map is CloudRF [9]. There are some other tools allowing to simulate radio signal propagation, of which most support ISM frequencies and allow to estimate the range of LP WAN access points, such as e.g. Radio Mobile Online [10]. These tools are very useful in analysis of network coverage for a given set of access points, but require large knowledge of RF signal propagation and do not select automatically the gateway location, requiring the experience of the operator. While this is effective for a small number of access points, for large networks which need to match the restraints of number of nodes services by an access point use of such tools is ineffective.

III. Model Description

LoRa uses Chirp Spread Spectrum what means that there is used fixed amplitude modulation. It uses the entire allocated spectrum by creating a signal that "sweeps" across the channel. If frequency moves upward it is a "upchirp" and if it moves downward it is "downchirp". LoRa can use one of three bandwidths – 125 kHz, 250 kHz and 500 kHz. Decreasing the Spreading Factor means a duration of chirp. LoRa operates with spreading factors from 7 to 12. Lower number causes a shorter transmission time. Higher spreading factor takes more time but is more robust. In general the spreading factor rises with distance between node and gateway. At [11] a spreadsheet based on Semtech documentation is shown, which allows to obtain a time duration to transmit LoRa packed with given parameters, such as payload size, header size, coding rate, bandwidth and Spreading Factor.

A. Limit of the number of devices supported by the LoRa Gateway

The transmission time of a single LoRa packet can be calculated using the following equation:

$$\begin{gathered}
T_{symbol} = \frac{2^{SF}}{BW} \\
T_{preamble} = (n_{preamble} + 4.25) * T_{symbol} \\
Payload = 8+ \\
Max\left(\left\lceil \frac{8PL - 4SF + 44}{4(SF - 2DE)} \right\rceil * (CR + 4), 0\right) \\
T_{payload} = Payload * T_{symbol} \\
T_{packet} = T_{preamble} + T_{payload}
\end{gathered} \tag{1}$$

where

- BW – bandwidth (125 used),
- $n_{preamble}$ – symbols in preamble (8 used),
- PL – payload size (51 used),
- DE – equal 1 for SF11 and SF12, equal 0 for the rest of SF,
- CR – coding rate (5 used),
- T_packet - packet transmission time.

We assume that all the nodes transmit packets of the same size within a specific time window, denoted as T_{int}. The moment of the transmission start is selected randomly with uniform distribution between the start of the time window and the $T_{int} - T_{packet}$ to finish the packet transmission before the end of the transmission window. Thus the probability of n packets being transmitted at the overlapping time is equal:

$$P = \left(1 - \frac{2T_{packet}}{T_{int}}\right)^{n-1} \tag{2}$$

it can also be transformed to:

$$\begin{gathered}
P = \left(\left(1 - \frac{2T_{packet}}{T_{int}}\right)^{\frac{2T_{packet}}{T_{int}}}\right)^{\frac{T_{int}*(n-1)}{2T_{packet}}} \\
P = e^{\frac{T_{int}*(n-1)}{2T_{packet}}}
\end{gathered} \tag{3}$$

The capacity n which is a maximum number of packets transmitted in a given time window T_{int} for a defined collision probability can be calculated as follows:

$$\begin{gathered}
ln(P) = \frac{T_{int}(n-1)}{2T_{packet}} \\
n = 1 + \frac{ln(P) * 2T_{packet}}{T_{int}}
\end{gathered} \tag{4}$$

after the simplification:

$$n = 1 + \frac{ln(P)}{ln\left(1 - \frac{2T_{packet}}{T_{int}}\right)} \tag{5}$$

where

- n - number of packet transmitted in a time interval
- T_{int} - interval used for packet transmission.

Using the above equation the maximum number of nodes can be calculated per given collision probability and for selected LoRa network parameters. The calculation have been executed for a single channel and single spreading factor - in the networks using multiple channels the load per single channel is equal to $\frac{1}{c*s}$, where c is the number of channels and s is the number of spreading factors.

IV. Gateway location selection algorithm

A. General description

To solve the problem of finding the location of gateways for a given set of points we have created a heuristic optimization algorithm. The algorithm starts with uniform distribution of gateways on a regular grid and tries to optimize it by moving the proposed location for a gateways towards areas with nodes wich have no coverage. When the algorithm is unable to assure full coverage only by shifts of the gateway location a new gateway is added, and when two gateways have overlapping coverage one of them is removed. To optimize the process of finding end points in range a bucket data structure is introduced.

B. Buckets

Buckets represent rectangular regions of space. They are created to reduce a number of operations taken by an algorithm. Amount of buckets depends on dimension of space covered by a gateways and its size. The size is an algorithm parameter and in described case it is calculated as a range where communication between gateway and node should be possible. To calculate this value the radio sensitivity, transmit power and propagation model are needed. After calculation of bucket size there is created a matrix where number of columns and rows is respectively x dimension and y dimension divided by bucket size. All nodes and gateways are assigned to corresponding bucket. Some bucket could be empty. Due to use of these buckets for example if algorithm needs all nodes with received signal strength above exact value of some gateway there is no need to iterate for every node in the network.

C. Algorithm Elements

The algorithm starts with placing the initial arrangement of proposed gateway locations in regular distances. This initialization procedure puts one gateway in the localization of middle node of all nodes in every bucket. After saving initial state of network, the algorithm starts the loop which can end if one of two conditions occurs: either the number of nodes without connection (function *no_connection()*) is equal or lower than a number set in configuration (by default 0, but it may be set to higher number to allow finishing the algorithm without full coverage, but lower number of gateways) or main loop counter exceeds the maximum number of steps threshold. This threshold can set in configuration, the higher number leads to longer computation but could give better results.

The loop contains a *move_gateways()* function which is responsible for choose better localization of gateway and better resources utilization. If *move_gateways()* function achieve better results than in saved state, the saved state is updated.

This step may be repeated a few times in a sub-loop. Next it is verified if the number of nodes without connection is higher than in previous saved state, if true it restore network to the best result of move_gateways() function. Next step of main loop is to invoke function which removes gateways from places where they are too close to each other. The last step is to add a gateway to the region where are nodes without coverage by any access point. This is realized by the three functions: *move_gateways()*, *remove_excess_gateways()* and *add_gateway_in_uncovered_region()*. In more details they work as follows:

- *move_gateways()* function for each gateway in the whole network finds the nodes which could be in its range and plus – in our case – 10% more. Range is calculated on base of transmit power, receiver sensitivity and propagation model. Next, all nodes in range+ are checked against if exists association to any gateway, if not the location of such node is taken into calculation of average location of unconnected nodes. If average location is different more than one tenth of range of current gateway then new serving gateway is moved by average location times random percentage. Random value helps to avoid returning to same state as in previous iterations.
- *remove_excess_gateways()* iterate for each gateway and then iterates for every connected node. The numeber of nodes connected to a gateway and the number of nodes connected to gateway within range of others is compared. If nodes in range of others plus random value between zero and maximum nodes per gateway divided by parameter is higher than the nodes count for current gateway than the current gateway and all references to it are deleted.
- *add_gateway_in_uncovered_region()* aims to depute a node to become access point for others in region where is lack of coverage. Every bucket is checked if there nodes without connection. If there is more than one node without connection in bucket a new gateway is placed in randomly selected location. Due to the simplicity of this function we have omitted its pseudocode in the paper.

```
save_current_state()
while (no_connection() >= maxUnconnected)
{
  if (ml_counter > maxSteps)
    break;
  else
    ml_counter++
  move_gateways()
  if (no_connection() < no_conn_saved() )
    save_current_state()
  else
    restore_saved_state()
  remove_excess_gateways()
  add_gateway_in_uncovered_region()
}
```

```
move_gateways() {
  foreach(gw in gateways) {
    find_not_connected_in_range_plus()
    foreach(node in not_connected)
      average+=node
    if (distance(average, gw_location)
                      < range/rangeDenomin )
      move_by(average*random x%)
    }
  }
}
```

```
remove_excess_gateways() {
  foreach(gw in gateways)  {
    count = 0;
    foreach(node connected to gw) {
      count += 1;
      if (node_in_range_of_other(gw))
        count_duplicates+=1;
    }
    if (count_duplicates+random >= count)
      delete_current_gw();
  }
}
```

V. Results and Discussion

The proposed gateway location selection algorithm was evaluated through an implementation in simulated environment. The end nodes were placed randomly or on a regular grid (uniformly) within a rectangular area. The radio signal propagation was modeled using SUI (Stanford University Interim) presented in [12]. We have implemented the proposed algorithm by extending the PyLTEs optimization tool [13] in Python, replacing the LTE base station model with LoRa gateway. The code of the algorithm has been made available at GitHub [14].

The following LoRa radio parameters were used:

- 125 kHz bandwidth,
- 8 symbols preamble,
- 51 bytes of payload,
- coding rate equal 5,
- 14 dBm of transmit power
- -133 dBm of receiver sensitivity,
- 8 channels, 5 spreading factors.

For the sake of evaluation the algorithm has been parametrized to use the transmission window time of 1h, minimum delivery probability 60% and both rangeDenominator and capacityDenominator parameters equal 10. The figures 1 – 6 depicts some examples which differentiate on nodes count and area size. The execution of the algorithm on different sets of points showed that it is always capable of finding a near-optimal solution, with small number of gateways providing full coverage and taking into account the capacity limitation of a maximum number of nodes per gateway. The colors on the plots have been added only to represent the coverage of different gateways and make the plot more readable.

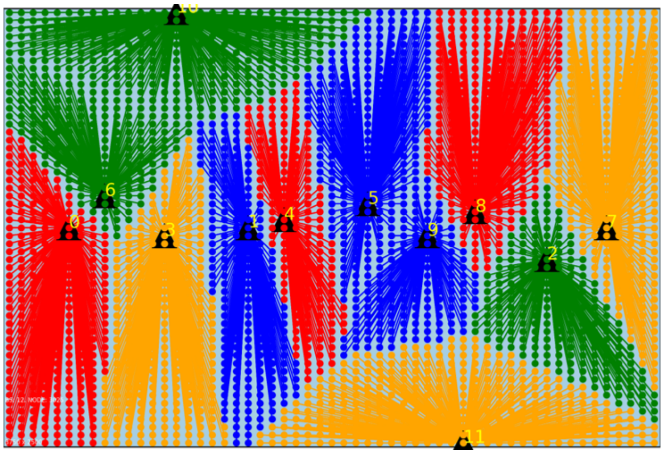

Fig. 1: Gateway locations for 3 750 x 2 500 m area, 3 000 nodes, 12 gateways, 20 iterations of main loop, 50 iterations of move function, uniform nodes' distribution.

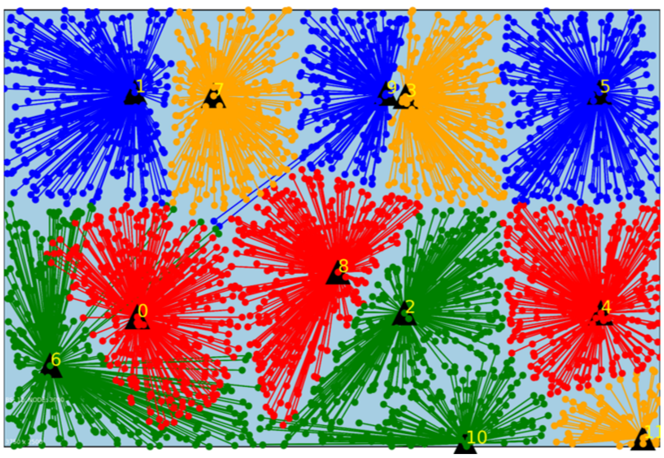

Fig. 2: Gateway locations for 3 750 x 2 500 m area, 3 000 nodes, 12 gateways, 20 iterations of main loop, 50 iterations of move function, random nodes' distribution.

The results for smaller networks, with 3000 nodes, shown on figures 1 and 2 show large degree of irregularity of the selected locations of gateways, which is caused by the fact that the algorithm minimizes the number of gateways that provide full coverage, without taking into account the distances between them. In larger networks the maximum number of nodes served by a gateway is the main limiting factor and the selection of locations is more regular, what can be seen on figures 3 and 4. It is in some cases disrupted by some randomness of the algorithm, which is a case e.g. for a gateway 74 and 75 on figure 6. This is caused by the randomness and finding some local minimum.

The tables 1 - 3 show the comparison of the number of gateways selected by the algorithm for random topology (table 1), topology with all nodes placed uniformly on a grid (table 2). For a reference we have calculated a minimum number of gateways which can provide provide full coverage assuming

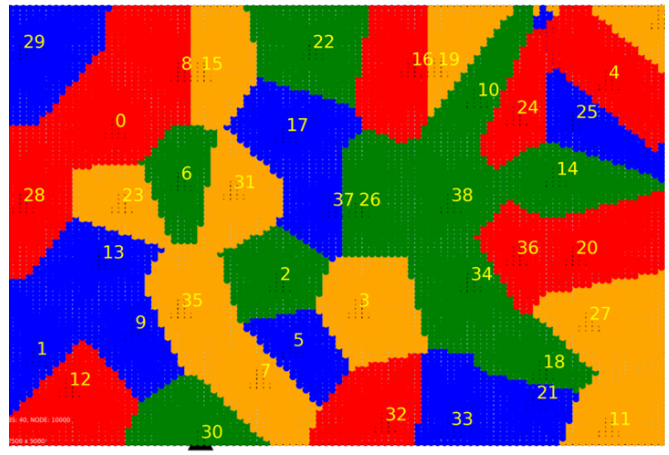

Fig. 3: Gateway locations for 7 500 x 5 000 m, 10 000 nodes, 40 gateways, 10 iterations of main loop, 5 iterations of move function, uniform nodes' distribution.

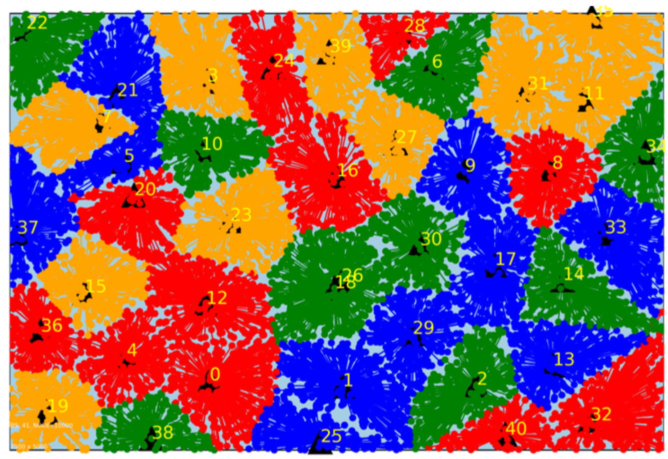

Fig. 4: Gateway locations for 7 500 x 5 000 m, 10 000 nodes, 40 gateways, 10 iterations of main loop, 5 iterations of move function, random nodes' distribution.

	1000 nodes	2500 nodes	10 000 nodes	20 000 nodes	30 000 nodes
3750 x 2 500 m	5	10	37	71	110
7500 x 5 000 m	14	15	40	76	110
15 000 x 10 000 m	46	52	54	76	127
30 000 x 20 000 m	158	181	196	197	189

TABLE I: Number of gateways, random nodes' distribution.

they are located in a regular grid, what is shown in table 3. The outcome of the algorithm is very similar for the regular and the random placement of the nodes, with differences of only a few gateways. The proposed gateway location selection algorithm allows to find a solution with lower number of gateways for most of the cases comparing to the regular location listed in table 3, except of a very large networks with large number of nodes (20 000 or 30 000), where finding the optimal solution is very time consuming and due to the nature of the test

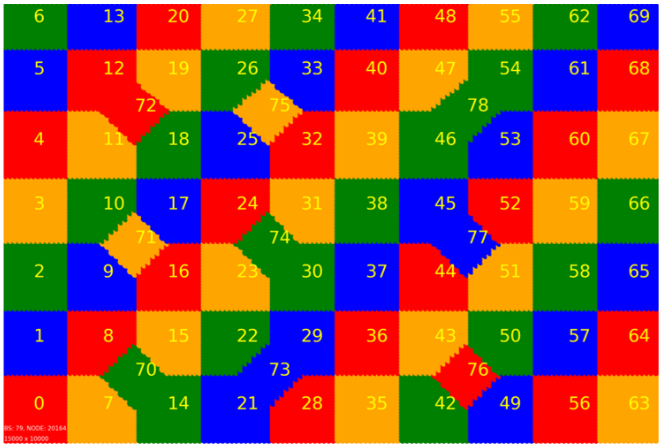

Fig. 5: Gateway locations for 15 000 x 10 000 m, 20 000 nodes, 79 gateways, 50 iterations of main loop, 5 iterations of move function, uniform nodes' distribution.

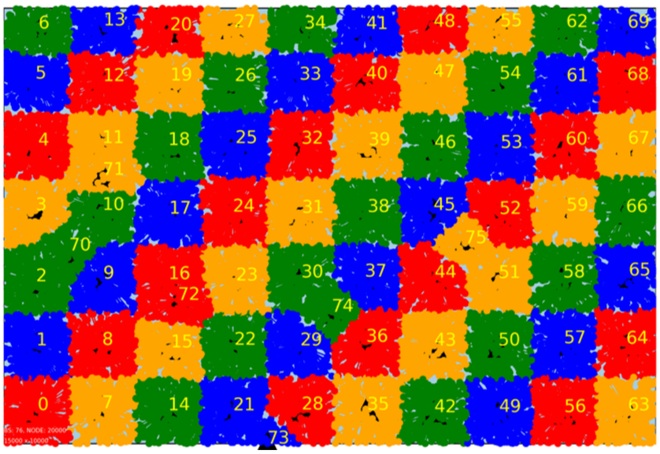

Fig. 6: Gateway locations for 15 000 x 10 000 m, 20 000 nodes, 76 gateways, 50 iterations of main loop, 5 iterations of move function, random nodes' distribution.

scenario (rectangular area with uniform distribution of nodes), the placement of the gateways on the regular grid is very effective. But in cases where the irregularities in node location can be exploited by the algorithm, the number of gateways selected is lower, which can be seen by the lower values in table 1 for the 3750x2500 m area.

	1000 nodes	2500 nodes	10 000 nodes	20 000 nodes	30 000 nodes
3750 x 2 500 m	5	11	38	71	109
7500 x 5 000 m	14	15	39	74	113
15 000 x 10 000 m	52	53	53	79	126
30 000 x 20 000 m	176	175	198	200	190

TABLE II: Number of gateways, uniform nodes' distribution.

VI. Conclusions

We have presented a novel heuristic algorithm for LoRa network gateway location selection. The results of the evaluation for random and regular topologies show that it gives

	1000 nodes	2500 nodes	10 000 nodes	20 000 nodes	30 000 nodes
3750 x 2 500 m	6	12	60	104	117
7500 x 5 000 m	15	15	60	77	117
15 000 x 10 000 m	54	54	54	77	117
30 000 x 20 000 m	176	176	187	187	176

TABLE III: Number of gateways without algorithm with full coverage, uniform nodes' distribution.

near-optimum selection of location, maintaining the limit of the maximum nodes per gateway to meet the required packet collision probability limit.

Acknowledgment

This work has been partially funded by the Polish National Center for Research and Development grant no. POIR.04.01.04-00-0005/17.

References

[1] A. Lavric and V. Popa, "Internet of things and lora^{TM} low-power wide-area networks: a survey," in *2017 International Symposium on Signals, Circuits and Systems (ISSCS)*. IEEE, 2017, pp. 1–5.
[2] I. LoRa Alliance, "Lorawan 1.1 specification," 2017.
[3] "Lorawan adaptive data rate," https://www.thethingsnetwork.org/docs/lorawan/adaptive-data-rate.html, accessed: 2020-01-07.
[4] J. Laiho, A. Wacker, and T. Novosad, *Radio network planning and optimisation for UMTS*. Wiley Online Library, 2002, vol. 2.
[5] S. Hurley, "Planning effective cellular mobile radio networks," *IEEE Transactions on Vehicular Technology*, vol. 51, no. 2, pp. 243–253, 2002.
[6] J. Alber, M. R. Fellows, and R. Niedermeier, "Polynomial-time data reduction for dominating set," *Journal of the ACM (JACM)*, vol. 51, no. 3, pp. 363–384, 2004.
[7] K. Staniec and M. Kowal, "Lora performance under variable interference and heavy-multipath conditions," *Wireless communications and mobile computing*, vol. 2018, 2018.
[8] B. Vejlgaard, M. Lauridsen, H. Nguyen, I. Z. Kovács, P. Mogensen, and M. Sorensen, "Interference impact on coverage and capacity for low power wide area iot networks," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2017, pp. 1–6.
[9] "Cloudrf," https://cloudrf.com/LoRa_planning, accessed: 2020-02-07.
[10] "Radio mobile online," https://www.ve2dbe.com/rmonline_s.asp, accessed: 2020-02-07.
[11] "Lora(wan) airtime calculator," https://docs.google.com/spreadsheets/d/1QvcKsGeTTPpr9icj4XkKXq4r2zTc2j0gsHLrnplzM3I/edit, accessed: 2020-02-07.
[12] V. Erceg, L. J. Greenstein, S. Y. Tjandra, S. R. Parkoff, A. Gupta, B. Kulic, A. A. Julius, and R. Bianchi, "An empirically based path loss model for wireless channels in suburban environments," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 7, pp. 1205–1211, 1999.
[13] M. Slabicki and K. Grochla, "Pyltes—python lte evaluation framework for quick and reliable network optimization," in *2016 39th International Conference on Telecommunications and Signal Processing (TSP)*. IEEE, 2016, pp. 64–67.
[14] "Pylores on github," https://github.com/iitis/pylores, accessed: 2020-02-07.