Collision-Free Sensor Data Collection using LoRaWAN and Drones

Dimitrios Zorbas and Brendan O'Flynn
Tyndall National Institute,
Cork, Ireland
{dimitrios.zormpas, brendan.oflynn}@tyndall.ie

***Abstract*—An open issue in Wireless Sensor Networks (WSNs) is the data collection from nodes placed at distant positions where no Internet or fixed gateway coverage is available. In this paper, we propose a reliable and energy-efficient solution using drones as mobile gateways that periodically fly over the network and collect data. We consider a point-to-point communication model between the nodes and the drones using the LoRaWAN communication protocol. Due to the nature of the default LoRaWAN MAC protocol, we modify its ALOHA-style transmission policy introducing a more efficient time-scheduled transmission mechanism to eliminate potential packet collisions. Simulation results show that a single drone can collect the data of an entire day of an area of more than $1500 \times 1500m^2$ and 80 nodes while achieving 0% packet collisions.**

I. Introduction

The Internet of Things (IoT) technologies are cheap and efficient means for monitoring applications, such as infrastructure, air quality or smart agriculture. A large number of sensing systems such as the WSNs are deployed in large geographical areas and can provide significant information that can be exploited to improve the energy efficiency of buildings, the quality of crops in agriculture, and strengthen the environmental protection against hazards.

The WSN nodes periodically take measurements and send the data to a nearby central unit which acts as the gateway between the WSN and a core network like the Internet. The distance between the nodes and the gateway may vary according to the applied communication protocol. Recent advances in Low-Power Wide Area Networks (LPWANs) permit high communication distances of up to some kilometers depending the configuration and the environmental conditions. However, even that range may not be enough to interconnect networks deployed at distant rural areas with very limited 4/5G or other IoT technology coverage. The high cost of licensed networks and the lack of a seamless power supply for the gateway may also be an issue.

In this paper, we tackle this problem by proposing a data collection approach using drones. The drones play the role of the gateway; they periodically (e.g., once per day) travel to places with poor connectivity and collect the data flying over the nodes. Apparently, this solution can be applied to applications where real-time data collection is not of first priority. Examples of this type of applications are the agriculture field monitoring, the monitoring of water quality of rivers, air pollution monitoring etc.

Unlike most other works in the literature, in this paper we achieve data collection taking into account and focusing on point-to-point LoRaWAN communications [1]. LoRa is a proprietary spread spectrum modulation technique that trades data rate with distance and is one of the most promising technologies of LPWAN. LoRa operates on sub-GHz unlicensed bands, which makes its adoption very straightforward for IoT operators. However, considering the current ALOHA-style transmission policy of LoRaWAN MAC layer and the fact that the nodes have to send a considerable amount of data (i.e., accumulated data of one day) in a short amount of time, a high number of collisions may happen. To tackle this problem we propose a time and frequency scheduling algorithm to allow transmissions only at certain times. The algorithm takes into account potential clock desynchronization between the nodes and the drone arrival time. Our simulation results show that a single drone can cover an area of more than $1500 \times 1500m^2$ and more than 80 nodes.

The contribution of this paper is fourfold; (a) we introduce the minimum time data collection problem of a WSN using LoRaWAN and drones, (b) we present *Drone Data Collection Heuristic* (DDCH); an efficient solution to compute drone data collection points, (c) we enhance DDCH with *Spreading Factor Allocation Heuristic* (SFAH); a fast heuristic to efficiently schedule node transmissions and reduce the drone's flying time, and (d) we show through simulation that our approach is capable of gathering data from multiple nodes once their operation is properly scheduled and clock drift issues are considered.

II. Related research

The data collection from WSN nodes using mobile vehicles is not a new concept. An extensive literature review is provided by Di Francesco et al. [2]. Their work surveys 1-hop and multi-hop communication approaches, clustering solutions as well as mobility patterns and speed control methods to improve reliability and energy efficiency. Similar and recent works are those of Khan et al. [3], Yao et al. [4], and Zhao et al. [5].

Recently, Zhan et al. [6] have proposed a convex sub-optimal optimization solution to jointly optimize the nodes' wake-up schedule and drones' trajectory to minimize the maximum energy consumption of all nodes. The data collection problem using drones in case of emergency events is examined by Cao et al. [7]. The authors propose a cloud-assisted approach for deriving UAV's optimal flying and data acquisition sequence of a WSN cluster.

All these works focus on the vehicle trajectory planning and the node communication architecture while they neglect the effect of communication protocol parameters on the data collection time. The work closest to ours is that of Reynders et al. [8]. The authors propose a new MAC approach to schedule LoRaWAN transmissions so that multiple SF are used in parallel and the energy consumption is minimized by adjusting the nodes transmission power. However, synchronization is based on beacons which cannot be applied in our case.

III. Problem description

Given a set of nodes N with known locations, a drone with initially known position, and k data packets per node to be collected, we define the "Minimum Time Data Collection" (MTDT) problem as an optimization problem such as; *(a)* the total drone flying time is minimized, *(b)* all data packets from all nodes in N are collected, and *(c)* the nodes energy consumption is minimized.

The problem falls into the category of vehicle routing and facility location problems [9], [10] which are NP-Complete or NP-Hard to be solved to optimality[1].

The drone flying time is affected by many parameters such as the size of the network deployment, the number of sensors, the data transmission time, and possible re-transmissions. Besides, since LoRaWAN is used, the transmission time depends on many configuration parameters, mainly the Spreading Factor (SF) and the bandwidth (BW). For instance, the higher the SF the higher the communication range, which implies shorter drone moving times. However, the higher the SF, the lower the data rate and, thus, the higher the transmission time. So, a trade-off between the SF and drone movement exists.

Moreover, transmissions in LoRaWAN are vulnerable to extensive collisions when a number of conditions are met [11]. The number of collisions may be considerably high in our case considering that (a) the drone cannot stay more than a few minutes at each location due to its limited battery lifetime and (b) each node has to transmit a high volume of data (e.g., aggregated data of the day) in this short amount of time. In order to quantify this packet loss, we simulate a scenario with variable number of nodes randomly placed in a square terrain of 1500m side. A time window of 5 minutes was given to the nodes to transmit the entire volume of data. For this purpose we use the LoRa Simulator [11] considering the parameters described in Section V-A. As we can see from Fig. 1, the throughput decays notably as more nodes are added in the field. Based on this result and theoretical works [12], [13] we can conclude that a most sophisticated data transmission approach should be followed.

IV. Drone-assisted data collection heuristic

In order to collect the data we propose a two-phase method which first allows a drone or a fleet of drones to visit the network area and identify the maximum reliable distances for all the available spreading factors. During the second phase, we introduce the DDCH which computes drone data collection points as well as node radio active/sleep mode schedules.

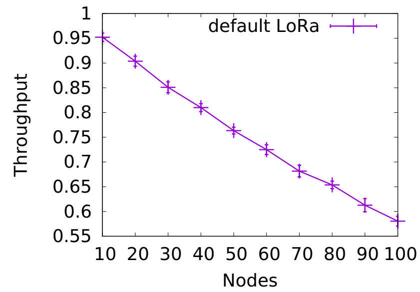

Fig. 1. Throughput of LoRaWAN with variable number of nodes transmitting 288 packets in 300 seconds.

In this paper, we focus on the second phase of the method and we give details on how the data collection process is achieved. DDCH performs as a greedy heuristic and consists of a main routine and two subroutines; the *data collection point* routine, the *optimal drone position*, and the *node scheduling* subroutines.

1) Data collection point routine: This is the outer loop routine which takes as input the location of the nodes and the maximum distance of a node per SF which has been experimentally found during the first phase. Using the two other encapsulated subroutines it computes one or more data collection points that minimize the distance between the nodes and the drone so the flying time of the drone is also minimized.

The data collection point routine picks the shortest to the initial drone position node and attempts to expand the drone coverage by adding more nodes in the range. To do so, it successively adds one node per time and it updates the drone position taking into account the new node transmission time and location. At every new position the algorithm checks if: (a) all the nodes in the range can reach the gateway (i.e., the drone), and (b) it is more efficient (less time consuming) to break the data collection time in two positions, P_1 and P_2. This is a decision that is made according to Eq. (1).

$$T_1' \leq T_1 + T_2 + M_{1\rightarrow 2}. \qquad (1)$$

This equation takes into account the data collection time at the current position T_1, the data collection time at the new position T_1', the data collection time at a separate drone position T_2 as well as the time to move from position P_1 to position P_2 (i.e., $M_{1\rightarrow 2}$). DDCH will finally include the last picked up node in the drone's range if Eq. (1) is true. Otherwise, it will split the drone's movement in two positions.

2) Optimal drone position subroutine: The objective of the *optimal drone position* subroutine is to compute drone positions that minimize the data collection time which mainly depends on the air packet time (i.e., transmission time) $T_{SF,BW}$ achieved by different LoRaWAN SF/BW combinations [14]. The transmission time is given by Eq. (2) and it increases with

[1]We omit the formal formulation of the problem due to the limited size of this paper. It will soon be done in a more extended version.

higher SFs and lower BWs.

$$T_{SF,BW} = (N_p + 4.25)\frac{2^{SF}}{BW} + \left(8 + \max(\lceil\frac{8PL-4SF+28+16-20H}{4(SF-2DE)}\rceil(CR+4),0)\right)\frac{2^{SF}}{BW}, \quad (2)$$

where N_p is the number of programmed preamble symbols, PL is the packet payload, H = 0 when the header is enabled and H = 1 when no header is present. DE = 1 when the low data rate optimization is enabled and DE = 0 for disabled.

Besides, SF and BW affect the transceiver sensitivity ($SEN^{SF,BW}$) which increases with higher and lower SF and BW values, respectively. Moreover, a signal can be decoded at the gateway (drone) only when its power (P_{rx}) is higher than a transceiver sensitivity as it is defined by the corresponding combination of SF/BW:

$$P_{rx} > SEN^{SF,BW}, \quad (3)$$

where P_{rx} is given by the following formula:

$$P_{rx} = P_{tx} + G - L - L_{pl}. \quad (4)$$

P_{tx} is the transmission power, G is the antenna power gains and L is the power losses at the transmitter. L_{pl} describes the attenuation of the signal in relation with the distance d:

$$L_{pl} = L_{pl}^{d_0} + 10\gamma \log \frac{d}{d_0} - \sigma^2 g, \quad (5)$$

$L_{pl}^{d_0}$ is the power at reference distance d_0, γ is the path-loss factor, σ^2 is the variance, and $g \in (0,1)$ a random real value.

Given the equations (2)-(5), the factor that mainly affects the received signal power and, thus, the SF is the distance between the nodes and the drone. Hence, the computation of the drone position is a classic Facility Location Problem (FLP) which is NP-Hard to solve to optimality. However, since in LoRaWAN the transmissions are orthogonal, the data collection time mainly depends on the transmission time of the nodes with higher SFs. Thus, the problem is transformed to a minimax FLP which can be solved to optimality by computing the smallest enclosing circle (SEC) in $O(n)$ time.

An example of the transition from the initial drone position to a new location or to a separate one is illustrated in Fig. 2. The big circles represent the SEC range (i.e., the distance to the most distant nodes) centered at the drone's X,Y coordinates and the dots the nodes. On the left subfigure, the drone's range expands to cover one more node, while on the right subfigure two separate locations are computed.

3) Node scheduling subroutine: This subroutine is used to compute and optimize the total transmission time within the drone's range at a certain location. As it is explained in Section III the throughput decreases considerably as more nodes are added in the drone's range due to the high data volume and the short window of the transmission time. To tackle this problem DDCH allows each node to wake up and transmit at certain time periods while nodes with the same SF configuration are scheduled at different time periods.

The issue of this approach is that the nodes need to be synchronized according to a global clock. It has been experimentally found that a node's clock can drift up to $30\mu s$

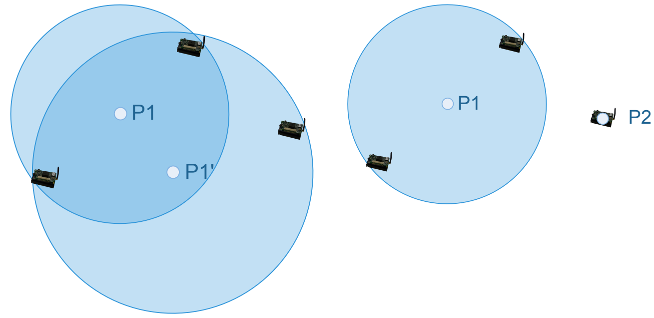

Fig. 2. Expanding drone range covering a 3rd node with a single position or splitting in two separate positions. The decision depends on the transmission times of the nodes and the distance between points P1 and P2.

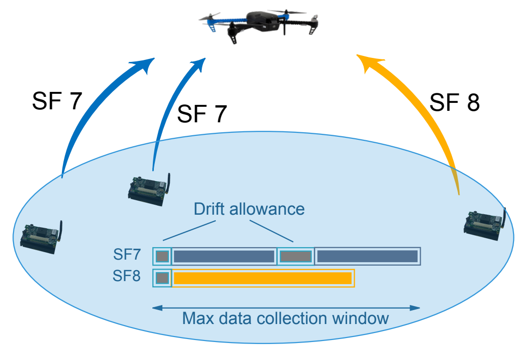

Fig. 3. Optimized data collection time with adjustable Spreading Factors.

per second [15] which in a period of one day this implies a total desynchronization of about 2.6 seconds. This practically means that the drone needs to be in the communication range of a particular node at least 2.6 seconds before the node's predefined wakeup scheduled time. Moreover, the algorithm must schedule the wakeup periods of two nodes with the same SF, allowing a gap of 2×2.6 seconds between them.

The purpose of the *node scheduling* subroutine is to adjust the nodes' SFs and schedule their active periods such as the data collection time is minimized. For example, as depicted in Fig. 3, if three nodes with SF=7 are in the drone's range, it is better to switch one of the nodes to a higher SF (e.g., SF=8), than using the same SF for all the nodes. Considering that 288 packets need to be sent per node and the transmission time per packet is 14.144ms and 25.728ms for SF=7 and SF=8 respectively (see Eq. (2) with BW=500), the data collection time in the latter case is $14.144ms \times 288 + 2 \times 2.6s + 14.144ms \times 288 + 2 \times 2.6s + 14.144ms \times 288 = 22.62secs$, while in the first case it is $\max(14.144 \times 288 + 2 \times 2.6s + 14.144ms \times 288, 25.728ms \times 288) = 13.35sec$.

In order to minimize the data collection time, we introduce the SFAH which solves the following optimization problem: Given a set of n nodes $S = \{s_1, \cdots, s_n\}$, a minimum SF value ϕ_i, $\forall\ s_i \in S$, and a set of k available SF subsets $F = \{f_1, \cdots, f_k\}$: $\forall\ f_j \in F\ \exists$ a given cost (transmission time) for each node $s_i \in S$ with $\phi_i \geq j$ described by the following formula,

$$C_{b_j,s_i} = \begin{cases} \psi_j, \text{if } |b_j| = \emptyset, \\ \psi_j + 2r, \text{if } |b_j| \neq \emptyset, \end{cases} \quad (6)$$

find the best SF allocation for the nodes in S such as the maximum sum of the cost of the SFs in F is minimized:

$$\min\left(\max\left(\sum_{i=1}^{|f_1|} C_{f_1,s_i}, \cdots, \sum_{i=1}^{|f_k|} C_{f_k,s_i}\right)\right) \quad (7)$$

subject to

$$\forall\ f_j\ \in F,\ \nexists\ s_i \in f_j : \phi_i > j, \quad (8)$$
$$|f_1| + \cdots + |f_k| = n. \quad (9)$$

Note that ψ_j of Eq. (6) corresponds to the transmission time for the given SF j described by Eq. (2). We assume an equal BW per node. r is the maximum drift allowance.

SFAH (see Algorithm 1) solves this problem by allocating first the nodes with the highest SF (i.e., ϕ value). It then checks all the available SFs for this particular node and it chooses the one with the minimum provisional sum of costs. The transmission time is computed as the maximum sum of costs of all the available SFs. The algorithm returns the total transmission time (i.e., $time$) as well as the collections of subsets F. Its time complexity is $O(n^2 + nk)$.

Algorithm 1: Spreading Factor Allocation Heuristic

require: S, F, and ϕ_i, $\forall\ s_i \in S$
foreach $f_j \in F$ **do** $f_j = \emptyset$;
$S' = \emptyset$;
$time = 0$;
while $|S'| < |S|$ **do**
  select node I in S but not in S' with $\phi_I = \max(\phi_1 \cdots \phi_n)$;
  if $time = 0$ **then**
    $J = \phi_I$;
    $f_J = f_J \cup \{I\}$;
  else
    $x = \phi_I$;
    select subset $f_J \in F$ with the minimum provisional cost
    $\min\left(\sum_{i=1}^{|f_x|} C_{f_x,s_i} + C_{f_x,I}, \cdots, \sum_{i=1}^{|f_k|} C_{f_k,s_i} + C_{f_k,I}\right)$
    $f_J = f_J \cup \{I\}$;
  $S' = S' \cup \{I\}$;
$time = \max\left(\sum_{i=1}^{|f_1|} C_{f_1,s_i}, \cdots, \sum_{i=1}^{|f_k|} C_{f_k,s_i}\right)$;
return $time, F$;

V. Evaluation & discussion of the results

A. Setup

In this section, we evaluate the proposed approach by conducting a set of simulations using a set of Perl scripts and a modified version of LoRaSim [11]. We consider two versions of DDCH, with and without the SFAH mechanism. In the latter case transmissions with the same SF are scheduled one after the other using the minimum possible transmission time according to the node-drone distance. We also present results of DDCH with the default LoRaWAN transmission policy for comparison purposes (appeared as "default"). We reduce the nodes transmission power to 7dBm for energy saving purposes. We also assume that the LoRaWAN Adaptive Data Rate mechanism is disabled. Table I summarizes the values of the simulation parameters. Sensitivity and signal attenuation values have been found experimentally. Each simulated scenario is executed 50 times using different random node placements and the average results are presented along with the 95% confidence intervals.

TABLE I
LoRaWAN and simulation parameters

Parameter	Value
Coding Rate (CR)	4/5
Channel Bandwidth (BW)	$500kHz$
Spreading Factor (SF)	7 – 12
Sensitivity per SF (SEN) and the given BW	-120.75, -124, -127.5, -128.75, -130, -132.25
Payload (PL)	20 bytes
Low data rate optimization (DE)	0
Header (H)	0
Preamble symbols (N_p)	8
Transmission power (P_{tx})	7 dBm
Gains minus losses ($G - L$)	0
Path loss exponent (γ)	4
Reference distance (d_0)	50 m
Power at reference distance ($L_{pl}^{d_0}$)	-80 dBm
Variance (σ^2) and g	8, $N(0,1)$
Packets to transmit (1 day's data)	288 (1 measurement / 5 min)
Clock drift per sec	$30\mu s$, $r = 30\mu \times 3600 \times 24s$
Drone average speed and height	4.9 m/s, 10m
Drone initial position	(0, 0)

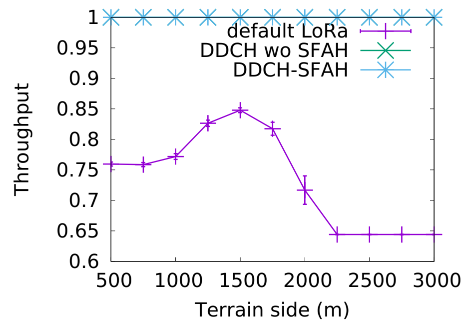

Fig. 4. Throughput for a scenario with 30 nodes and variable terrain size.

B. Results

Fig. 5 presents the total drone flying time for a scenario with variable terrain size and 30 nodes. We can observe that SFAH decreases considerably the flying time compared to the non-optimized version. Both DDCH versions present the same result when very high terrain sizes are considered since most of the nodes use very high SF values which can be only serially scheduled. Apparently, using the default LoRaWAN transmission policy, the flying time is minimized. However, due to the high number of collisions, the throughput is at least 15 to 35% less (see Fig. 4). Both DDCH approaches achieve zero collisions and thus minimize the energy consumption of the nodes since no retransmissions are required.

The results depicted in Fig. 6 reveal that most of the flying time is spent for movement. This movement includes the time

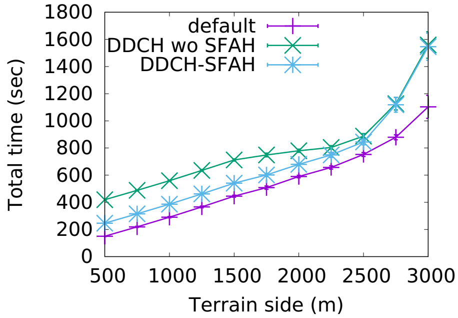

Fig. 5. Total drone flying time for a scenario with 30 nodes and variable terrain size.

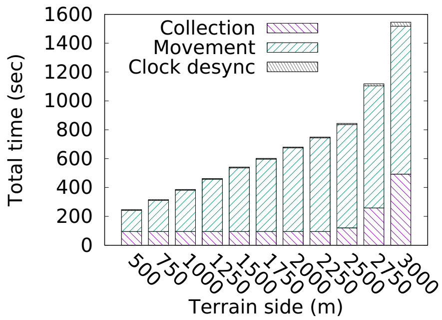

Fig. 6. Time reserved for each individual process (data collection, movement, and clock desynchronization) for the DDCH-SFAH algorithm.

required to move from the base station to the data collection point(s) and back to the base station. A very small part of the time is spent to ensure connectivity due to the clock drift.

In the last simulation we assess the behavior of the algorithms in a scenario with a fixed terrain size and variable number of nodes. The results are presented in Fig. 7 and show that SFAH reduces greatly the total flying time compared to the non-optimized approach. Taking into account that most commercial drones have a battery life of approximately 15 minutes, a single drone can cover about 80 to 90 nodes for the given area. The default LoRaWAN exhibits a constant trend since all the nodes can transmit in parallel.

VI. Conclusion & Future work

In this paper, we introduced the data collection problem using drones and scheduled LoRaWAN transmissions. We proposed an algorithm to compute data collection points taking into account the distance between the drone and nodes as well as LoRaWAN configuration parameters. We enhanced our approach by adjusting the spreading factors of the transmissions and allow as many parallel orthogonal transmissions as possible. Simulation results showed that a single drone can collect data of an area of more than 1500×1500m and 80 nodes without collisions.

Future work will focus on solving practical issues towards a real implementation of the system for smart agriculture purposes. More specifically, we will investigate the use of other SF/BW LoRaWAN configurations as well as different transmission power levels. We will also adapt our approach to fulfill the regional duty cycle regulations.

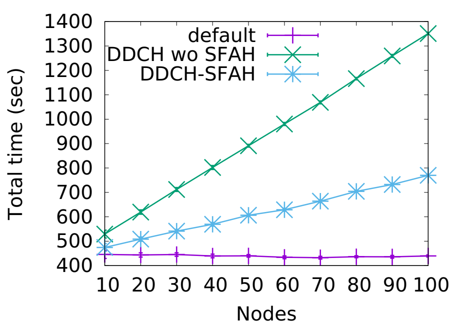

Fig. 7. Total drone flying time for a scenario with 1500×1500m terrain size and variable number of nodes.

Acknowledgments

This publication has emanated from research supported in part by a research grant from Science Foundation Ireland (SFI) and is co-funded under the European Regional Development Fund under Grant Number 13/RC/2077.

References

[1] "LoRa Alliance web page," 2016, https://www.lora-alliance.org/.
[2] M. Di Francesco, S. K. Das, and G. Anastasi, "Data collection in wireless sensor networks with mobile elements: A survey," *ACM Trans. Sen. Netw.*, vol. 8, no. 1, pp. 7:1–7:31, Aug. 2011.
[3] A. W. Khan, A. H. Abdullah, M. H. Anisi, and J. I. Bangash, "A comprehensive study of data collection schemes using mobile sinks in wireless sensor networks," *Sensors*, vol. 14, no. 2, pp. 2510–2548, 2014.
[4] Y. Yao, Q. Cao, and A. V. Vasilakos, "Edal: An energy-efficient, delay-aware, and lifetime-balancing data collection protocol for heterogeneous wireless sensor networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 23, no. 3, pp. 810–823, 2015.
[5] M. Zhao, Y. Yang, and C. Wang, "Mobile data gathering with load balanced clustering and dual data uploading in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 14, no. 4, pp. 770–785, April 2015.
[6] C. Zhan, Y. Zeng, and R. Zhang, "Energy-efficient data collection in uav enabled wireless sensor network," *IEEE Wireless Communications Letters*, vol. 7, no. 3, 2018.
[7] H. Cao, Y. Liu, X. Yue, and W. Zhu, "Cloud-assisted uav data collection for multiple emerging events in distributed wsns," *Sensors*, vol. 17, no. 8, 2017.
[8] B. Reynders, Q. Wang, P. Tuset-Peiro, X. Vilajosana, and S. Pollin, "Improving reliability and scalability of lorawans through lightweight scheduling," *IEEE Internet of Things Journal*, 2018.
[9] P. Toth and D. Vigo, *Vehicle routing: problems, methods, and applications*. SIAM, 2014.
[10] R. Z. Farahani and M. Hekmatfar, *Facility location: concepts, models, algorithms and case studies*. Springer, 2009.
[11] M. C. Bor, U. Roedig, T. Voigt, and J. M. Alonso, "Do lora low-power wide-area networks scale?" in *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '16. New York, NY, USA: ACM, 2016, pp. 59–67.
[12] D. Zorbas, G. Z. Papadopoulos, P. Maille, N. Montavont, and C. Douligeris, "Improving lora network capacity using multiple spreading factor configurations," in *25th International Conference on Telecommunication (ICT)*. IEEE, 2018.
[13] K. Q. Abdelfadeel, V. Cionca, and D. Pesch, "Fair adaptive data rate allocation and power control in lorawan," *arXiv preprint arXiv:1802.10338*, 2018.
[14] "Semtech. an1200.13 sx1272/3/6/7/8: Lora modem – designer's guide, revision 1," July 2013.
[15] D. Stanislowski, X. Vilajosana, Q. Wang, T. Watteyne, and K. S. Pister, "Adaptive synchronization in ieee802. 15.4 e networks," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 795–802, 2014.