Efficient Data Collection using Aerial Mobile Sinks for Delay Tolerant Wireless Sensor Networks

Djibrilla Incha Adamou, Véronique Vèque, Alexandre Mouradian, Sahar Hoteit, Lynda Zitoune
Université Paris-Saclay, CNRS, CentraleSupélec,
Laboratoire des Signaux et Systèmes, 91190, Gif-sur-Yvette, France
Email: {djibrilla.adamou, veronique.veque, alexandre.mouradian, sahar.hoteit,lynda.zitoune}@universite-paris-saclay.fr

***Abstract*—To overcome the lack of connectivity for collecting data in large white areas, we propose to provide long-range communication between aircraft and sensors on land. Our study is based on the use of commercial flights, with aircraft acting as mobile sinks, to collect data as they fly over an area of interest where sub-sinks are disseminated. Using relay nodes, the data can be collected even if the sensors are not located in the aircraft corridors. We investigate the feasibility of such a scheme by efficiently placing relay nodes to connect sources to sub-sinks which are covered by aerial corridors. The sources reach sub-sinks using a shortest path routing algorithm that reduces delays and balances the load among the sub-sinks. Simulation results and comparisons with well-known algorithms show the effectiveness of our algorithms which improve load-balancing by considering the throughput, the buffer capacity and the sub-sinks position.**

***Index Terms*—Wireless Sensor Network; Wide White Area; Delay Tolerant Network; Opportunistic Data Collection; MULEs ; Aircraft collector; LPWAN.**

I. INTRODUCTION

Wide White areas (WWA) are vast regions with harsh climate, difficult access such as deserts, pack ice or large forests. Although they are sparsely or even not equipped with infrastructure (energy, roads, etc), strategic human activities are being carried out such as mines, pipeline or, border security monitoring. As a matter of fact, all these activities can benefit from an infrastructure of sensors connected by a network in order to collect data. To tackle the problem of deploying networks in a very large area, Wireless Sensor Networks (WSN) have been proposed [1].

The deployment of WSN in WWA where wideness may be of thousands of kilometer squares is constrained by the large inter-node distances, the lack of energy supply, or problems of accessibility. An obvious solution is to use satellite communications because they cover a large area of the earth. However, their use suffers from a prohibitive operating costs, and energy supply is still an issue compared to the generated traffic. Moreover, in WWA, sensing applications are said to be delay-tolerant which means that they accommodate temporary disconnections, and can profit from any opportunistic access to send their data [2]. In opportunistic networks, data MULEs (Mobile Ubiquitous LAN Extensions) [3] are used to overcome the lack of network connectivity. MULEs are mobile data sinks or base stations, that can be embedded in buses, trains, balloons, drones, or aircraft [4], [5]. Even if data collection methods can be improved in WWA by using MULEs, they are quite slow and may suffer from inefficiency due to the increase of the data collection delay compared to fixed sinks [6].

However, with a good trade-off on the delay, Delay Tolerant Networks (DTN) may be deployed for applications in WWA [7], [8]. In this paper, we propose to develop a new opportunistic method based on the use of Low Power Wide Area Network (LPWAN) between aircraft and sensors on earth. In this scheme, the aircraft from commercial flights play the role of mobile sinks, to collect data while they fly over an area of interest sprinkled of sensors or sub-sinks.

In [9], we have proposed *BlobNET* to efficiently deploy a terrestrial multi-hop WSN to connect sources to a single sink connected to the server. It consists in selecting the most reliable positions in a set of potential relay nodes.

Here, we first extend *BlobNET* to deploy more than one sink, i.e. several sub-sinks which will be covered by aerial corridors, and act as relays between the ground and the aerial sink. This feature will allow to distribute collected traffic among several sub-sinks. Secondly, we develop a routing algorithm, called *BlobROUTE*, to transfer data from sources to their nearest sub-sink, while performing load balancing among sub-sinks. We then investigate the feasibility of such a scheme in terms of connectivity, number of hops by path, and forwarding delay including the transmission time from both sub-sinks-to-aircraft, and aircraft-to-sink.

To the best of our knowledge, our work is the first attempt to use commercial aircraft as opportunistic data MULES in WWA DTN.

The main contributions of this work are as follows:

ISBN 978-3-903176-42-3© 2021 IFIP

- To develop an opportunistic method to collect sensing data in WWA.
- To use commercial aircraft as opportunistic data MULES.
- To use a routing scheme to select efficient storage node or sub-sinks, in large WSN.
- To balance data traffic among sub-sinks according to their load and their throughput.

The remainder of the paper is organized as follows. Section II gives a literature survey. In the section III, we detail our modified version algorithms of *BlobNET*, the minimum relay nodes placement method, and our new algorithm, *BlobROUTE*, the routing algorithm to sub-sinks, plus load balancing. Performance evaluation is done in Section IV, where simulation scenarios and results are commented. Finally, conclusion and discussions are provided in Section V.

II. RELATED WORK

The extent of considered white areas such as deserts or forests, may be so large that a single connected WSN could not be deployed to cover the entire area, and white area WSNs are known to be prone to failures that often result in a disconnected network. Fortunately, such sensing applications are said to be delay-tolerant which means that they accommodate temporary disconnections, and can profit from any opportunistic access to send their data [2]. DTNs and data MULES are both solutions to cope with the wideness and the lack of connectivity of these types of WSNs. In DTN WSNs, data are collected from the sensors to some specific nodes called sub-sinks [10] located within their radio range. The data collector has then to visit these sub-sinks using an optimal trajectory to collect the data, to store and forward them to the final sink when coming back at home [11], [2]. These opportunistic data collectors can collect sensed information even if some parts of the WSNs are not connected. Thus, mobile collectors improve the network lifetime and increase the data collection efficiency by maximizing the amount of collected data [6], but they have some drawbacks such as the increase of data collection delay, the limited storage capacity, or the short contact time.

In SenCar [4], the authors propose a data gathering mechanism for large-scale multi-hop sensor networks. A robot or a vehicle, periodically collects data from the static sources as in [10]. By driving along a better path and balancing the traffic load from sensors to SenCar, the network lifetime can be prolonged significantly [4]. However, deploying terrestrial mobile sinks may not be possible because of the travel difficulties in WWAs, especially when the terrain is very hilly or mountainous.

Controlling the mobile sink path is a traditional issue considered in [12]. A path constrained mobility method is proposed to improve the energy and collection efficiency. In this protocol, the nodes in the network are organized in clusters composed of one cluster-head and simple members. Each cluster-head is a rendez-vous point where data are stored and forwarded to the mobile sink. The mobile sink only visits the cluster-heads to collect data. However, this method may suffer of node saturation problem, and unbalanced traffic because it does not take into account the contact time value between the cluster-heads and the mobile sinks. To overcome the saturation problem, authors in [10] have proposed a method to limit the number of members in each cluster. This method suffers from the increase of the forwarding delay which depends on the flight frequency [5]. To reduce time consumption, authors in [13] propose a basic framework for data collection, which includes the deployment of networks, nodes positioning, anchor points searching, fast path planning for UAVs. However, this solution focused mainly on the UAVs path calculation.

In our work, by considering mobile sink in high mobility scenario, we propose a new method which extends the network coverage for uncovered sources, and increases the collected data amount while decreasing the forwarding delay. By considering multiple sub-sinks, our work improves and overcomes the drawbacks of the study presented in [5].

III. DATA COLLECTION WITH PATH-CONSTRAINED AIRCRAFT

A. System Definition and Hypothesis

We consider a hybrid wireless sensor network (Fig. 1) consisting of a set S of sensor nodes, a set R of relay nodes, a set B of sub-sinks, a set M of mobile sinks which are the aircraft, and a data sink at the airport. Sub-sinks in B are intermediary nodes which are placed along the aircraft trajectory, and they act as gateways between the aircraft and the sources in S. They collect data from sensors, store and transmit them to the mobile sinks/aircraft when they fly over their range. The covered area is defined as the earth-projected aerial corridor. Several sub-sinks are uniformly placed in the covered area, and their position determines their contact time with the aircraft. Each sub-sink communicates with the aircraft by using a LoRa type radio communication [14] which is possible if the received signal power P_r is greater than the receiver sensitivity threshold ψ_r defined by:

$$\psi_r = -174 + 10log_{10}(BW) + NF + SNR_{target}. \quad (1)$$

ψ_r is in dBm, -174 the thermal noise in 1 Hz of bandwidth, NF the receiver noise figure (dBm), and SNR_{target} the minimum required signal to noise ratio in dB. P_r depends on the transmission power P_t and all gains and losses.

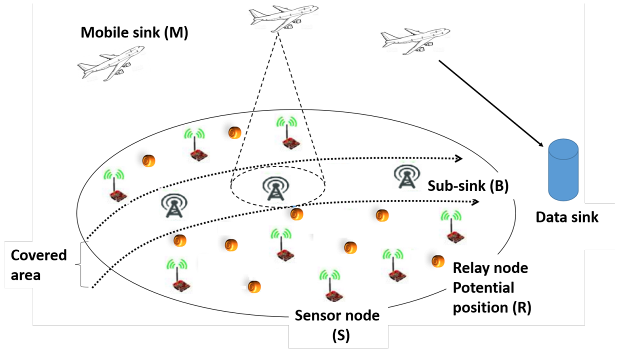


Figure 1: System model.

Two nodes i, j communicate with each other if the distance separating them $L_{ij} \leq r$ where *r* is the communication range of the nodes *i, j*. To determine *r*, we consider the free space path loss L_p model such that:

$$L_p = \bar{L}_p + 10\gamma log(\frac{r}{d_0}), \tag{2}$$

where $\bar{L}_p = 20log_{10}(\lambda/4\pi d_0)$ is the mean path loss at the reference distance $d_0 = [10 - 100]$ meters for outdoors [15], γ is the path loss exponent for suburban [15] and λ is the wavelength.

We assume that the instantaneous position called waypoint of the aircraft along its trajectory is given by $M(t) = v.(t - t_0) + M_0$ where $\vec{v}$ is a constant velocity vector, M_0 and t_0 are respectively the aircraft initial position and time. We also consider the sensor communication range as a half-sphere over the ground plane whose center is located in 3D coordinates (C_x, C_y, C_z). (X(t), Y(t), Z(t)) that are the 3D coordinates of an instantaneous way-point. A route crosses the sensor communication range if there exists at least one waypoint which is located in the spherical range and verifies the following equation:

$$(X(t) - C_x)^2 + (Y(t) - C_y)^2 + (Z(t) - C_z)^2 = r^2. \tag{3}$$

Among the two solutions, t_1 and t_2, a contact time $\tau = |t_1 - t_2|$ between the aircraft and the sensor is possible. Thus, the total amount of data that a sub-sink i can transmit to an aircraft is $C_i = DR_i.\tau_i$. *DR* is the LoRa data rate computed as in [14]. Then, over all the covered sensors, we have the amount of collected data Γ such as:

$$\Gamma = \sum_{i \in N} DR_i.\tau_i. \tag{4}$$

In order to tackle the energy consumption issues on the sensor nodes, we assume that the aircraft broadcast a wake-up signal on a common channel such that during the contact time, sensors are aware of its presence. This future is available with LoRa radio [16]. Furthermore, we define the frequency that the aircraft follows a given route by f_a, and it is computed by the ratio of the number of airplanes along the route N_a to the observation duration O_d. We note that the observation duration can be a day or a week. Following this definition, the collection capacity Υ_i of sub-sink i is determined by:

$$\Upsilon_i = C_i.f_a \tag{5}$$

In the next section, we introduce *BlobNET*, the bio-inspired method we have developed to interconnect, using WSN, the uncovered sources to the sub-sinks.

B. BlobNET: A Relay Nodes Placement Algorithm

BlobNET[1] is a heuristic that finds the sub-graph which connects data sources to sub-sinks by selecting a minimal number of relay nodes. We can tune the algorithm parameters such that it performs a complex network [9] with redundant links (Fig. 2c). Redundancy is an important feature as even if a link fails, an alternative route towards a destination may exist. In this study, we will further use this feature to select a target sub-sink among several candidates depending on the hop distance, the sub-sink load and the available storage capacity. All the sub-sinks are associated with a storage volume in which the collected data from sensor nodes are queued. With this hypothesis, the sub-sinks are able to receive all the generated data from the sensors. Therefore, we propose a routing protocol from sources to sub-sinks, called *BlobROUTE*. In the following, we first introduce *BlobNET* and then, *BlobROUTE*.

Let define a hybrid communication graph *G(V,E)* where $V = B \cup R \cup S$, the set of both sensors, relays and sub-sinks connected through communication links in the set of edges *E*. The relay-size of *G*, *s(G)*, is the number of relays in *G*, i.e $s(G) = |R|$. Let *T* be a subgraph of *G*. The weight of *T*, *w(T)* is defined as

$$w(T) = \sum_{e \in E(T)} w(e). \tag{6}$$

An edge weight $w(e)$ is the distance measure of the edge respectful to propagation constraints as in Eq. 1 and 2. As for *G*, the relay-size of *T* *s(T)* is defined as

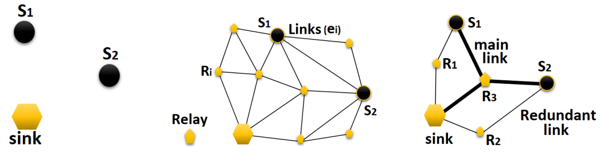
$$s(T) = |V(T) \cap R| \tag{7}$$

The relay nodes placement problem is defined as follows: Find the minimal $s(T)$ such that the subgraph T is connected and $s(T) \leq s(G)$. This is equivalent to the Steiner Tree problem known to be NP-hard [18]. We aim at finding a quite minimal set of relay nodes $R' \subset R$. To achieve the relay nodes selection, we have proposed *BlobNET*[9], an iterative algorithm to develop a WSN deployment heuristic for WWA which mimics

---

[1] It is called *BlobNET* by analogy with the name of ”blob” given to Physarum polycephalum in the work of [17].

the nutrient flow dynamics during the growth process of a biological organism called Physarum [19]. Physarum intelligence is able to find a minimal subset $s(T)$ from a potential relay positions R. This is illustrated in the Fig.2. For details on mathematical model of the algorithms used in *BlobNET*, see [9].



(a) Initial state (b) Connected state (c) Optimized state

Figure 2: Physarum network construction phases [20]

An initial physarum body (sub-sink) is placed in the space with 2 sources S_1, S_2 (Fig. 2a). It grows toward all sources (sensors), once found, it exploits them into nutrient which are conveyed through links. This leads to the creation of a fully connected network with a lot of tube cross-connect points, shown by the pentagons as in Fig. 2b. They act as relaying points. Finally, when there is no source at all, Physarum is able to optimize its body resources by deleting some unnecessary links and cross-connect points (Fig. 2c), which results in an optimized body structure. We are interested in the optimization phase. Two empirical rules describe changes in the Physarum structure: (1) open-ended links which are not connected to a source are removed; (2) when two or more links connect the same two nodes, the longest is deleted [21].

Fluid mechanics relationships are used in an iterative algorithm to describe these rules. To each node $i \in V$ (Fig. 2c) is associated an initial data flow I_i and a pressure P_i. For a source i, $I_i \neq 0$ allows to model generated data. For other nodes j, $I_j = 0$. The pressure P_i is a variable for which we solve equations at each iteration of the algorithm, and the difference of pressure $P_i - P_j$ between two nodes $i, j \in V$ drives the dynamics of the data flow in the network [21]. An edge connecting nodes i to j has a weight L_{ij}, and a capacity factor D_{ij} which is the width of the edge initialized with a value picked in $]0, 1]$. Finally, Q_{ij} is the data flow traveling through the edge (i, j), it is defined as:

$$Q_{ij} = \frac{D_{ij}}{L_{ij}}(P_i - P_j). \tag{8}$$

Moreover, at each iteration, two steps are performed: (1) Solving a set of equations (conservation law at each node) for the variables P_i such as

$$\begin{gathered} \sum_i Q_{ij} = 0 \quad j \neq s, t \\ \sum_i Q_{is} + I_0 = 0, \qquad \sum_i Q_{it} - I_0 = 0, \end{gathered} \tag{9}$$

with $j = s, t$ for source and sub-sink nodes respectively, and I_0 the absolute value of the initial flow for sources. (2) Updating the link (edges) parameters and new flow values in these links. To perform this step, we use a discretized flow adaptation equation such as:

$$\frac{D_{ij}^{n+1} - D_{ij}^{n}}{\delta_t} = f(|Q_{ij}^{n}|) - D_{ij}^{n+1}, \tag{10}$$

where f is an increasing function with $f(Q) = Q^{\mu}$, μ is the parameter that models the change of flow. D_{ij} tends to decrease if there is no flow in the edge but is augmented when the flow increases. n is the iteration index, and δ_t is the time step. The two steps are iterated until the difference $D_{ij}^{n+1} - D_{ij}^{n}$ is less than a predefined threshold. Then, the output of the algorithm only consists of D_{ij} for each link which survived from the suppression process, as explained in Fig. 2c. The initial data flow I_0 and the flow adaptation function exponent μ are two parameters to be tuned to modify the output of the network.

C. BlobROUTE in presence of several sub-sinks

After applying *BlobNET* (Fig. 2), the sources are connected using a mesh WSN to several sub-sinks covered by aerial flights. The next step, called *BlobROUTE*, is thus to select a target sub-sink among several candidates depending on the hop distance, the sub-sink load and the available storage capacity. All sub-sinks are associated with a storage volume in which the data collected by the sensor nodes is queued.

$$\phi = min(h_1, h_2, ...h_K), \tag{11}$$

where $k = \{1, 2, ..., K\}$ is the number of available paths and h_k the number of hops in the kth path. To find out ϕ, we have modified the heuristic in III-B by considering all the paths between each pair of (source, destination). The results are compared with another method which is based on the Dijkstra algorithm.

Before selecting a sub-sink, a source has to know all the sub-sinks characteristics that are connected to it. These characteristics are the contact time, the forwarding delay and the en-queued data amount or the available storage capacity. The source then selects its best sub-sink.

If a sub-sink $i \in B$ operates with a given spreading factor SF, a data rate of *DR* [16], and has τ as contact time, then the total amount of data C_i that it can transmit to the aircraft is $C_i = DR.\tau$. The sub-sink transmits a total of $C_i.F_a$ bits to all the aircraft (see Section III-A).

If X bits are queued in a sub-sink storage then we defined the forwarding delay Δ, as the time that it needs to drain all the stored data to the aircraft which is as follows:

$$\Delta = \frac{X}{C_i}. \tag{12}$$

Algorithm 1: *BlobROUTE*

```
Data: Initial graph G(V, E, L), source $s$,
      destination $t$
Result: Shortest path between $s$ and $t$.
/*Initialisation*/;
$D_{ij} \leftarrow ]0,1]$ $(\forall i, j = 1, 2, ..., N \wedge L_{ij} \neq 0)$;
$Q_{ij} \leftarrow 0$ $(\forall i, j = 1, 2, ..., N)$;
$Q^p_{ij} \leftarrow 0$ $(\forall i, j = 1, 2, ..., N)$;
$p_i \leftarrow 0$ $(\forall i = 1, 2, ..., N)$;
$n \leftarrow 1$;
$converge \leftarrow 0$;
while converge=0 do
    Choose a source ;
    Choose a destination ;
    $p_k \leftarrow 0$;
    Calculate all nodes pressure ;
    for $i = 0;\ i < N;\ i = i + 1$ do
        for $j = 0;\ j < N;\ j = j + 1$ do
            /*New flows*/ ;
            $Q_{ij} \leftarrow D_{ij}(p_i - p_j)/L_{ij}$ ;
        end
    end
    /*Links capacity adaptation*/ ;
    for $i = 0;\ i < N;\ i = i + 1$ do
        for $j = 0;\ j < N;\ j = j + 1$ do
            $D_{ij} \leftarrow f(Q_{ij}) + D_{ij}$ ;
        end
    end
    /*Comparing new and previous flows*/ ;
    if $Q^p == Q$ then
        converge $\leftarrow 1$;
    else
        $Q^p \leftarrow Q$;
        $n \leftarrow n + 1$;
    end
end
```

BlobROUTE aims at optimizing two criteria. The forwarding delay is the first criterion that the uncovered sources estimate while they choose a sub-sink as gateway to the aircraft. They must select the sub-sink which has the smallest forwarding delay such as $\{\Delta_{min} = min(\Delta_j), j = \{1; 2, ..., M\}\}$ where M is the number of the sub-sinks in the set B. Secondly, they perform a shortest path calculation to route their data to the selected sub-sink. We compare *BlobROUTE* with the method presented in [12] which is based on the distance calculation to choose the nearest sub-sink. For our convenience in this paper, We call it *SDBM* for Shortest Distance Based Method, as described in the section II.

In the next section, we evaluate our methods through simulation experiments.

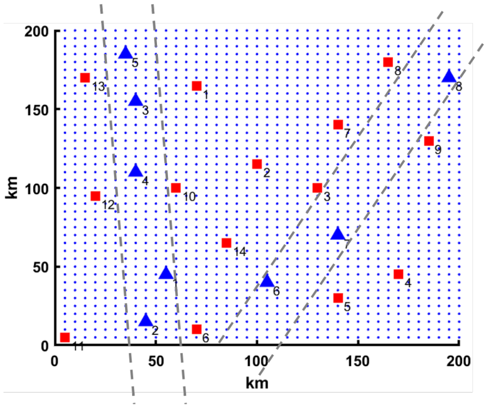


Figure 3: The Area of Interest. Circles are potential sensor positions, squares the data sources and triangles the covered sub-sinks.

IV. EVALUATION OF THE METHOD

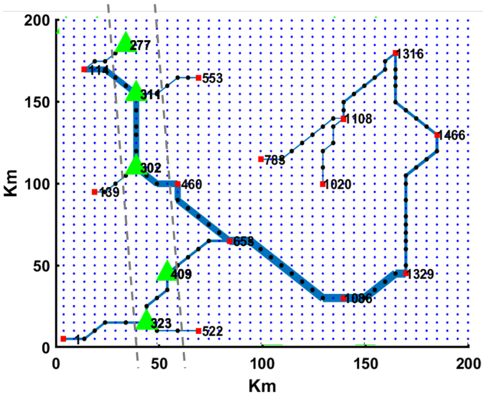
A. Scenario

An Area of Interest (AoI) is defined as illustrated in Fig. 3 which represents the AoI covering about $40000 km^2$. In Fig. 3, the dashed lines delimit the on-ground projections of the flight corridors, the triangles are the sub-sinks, the dots the relay node positions and the squares the sensor nodes. The sensor communication range is of $r \leq 15\ km$ to fit the LPWAN LoRa radio range [22]. As it has been shown in [5], with a LoRa spreading factor of 7, and for both 125 or 500 Khz of bandwidth, a range of 10 km may be reached. Then, this distance covers the aircraft altitude. The AoI is pixellized and each pixel is of 5 km side in the center of which is a virtual position of the relay nodes.

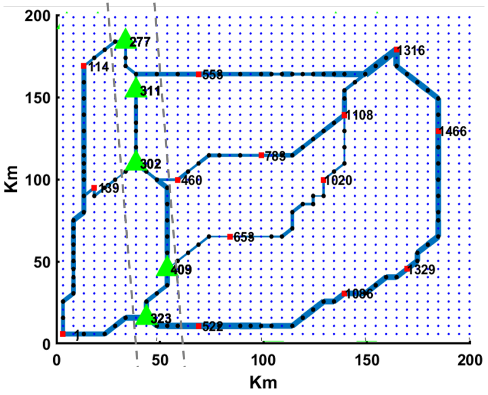
The numbers of corridors, sub-sinks and sensors are variable. In this scenario, we use the *BlobNET* algorithm, described in Section III-B, to select a minimal number of relays to interconnect all sensors to sub-sinks, then *BlobROUTE* computes the shortest path between a source and its selected sub-sinks. An observation period or *a tour* is done when all the sources have transmitted to the sub-sinks. All simulations are done with Matlab [23].

B. Interconnection of sources to sub-sinks and routing

1) Connectivity: To evaluate the proposed algorithms, we first analyze the resulting graph in terms of performance metrics such as the number of relays which are selected to achieve the interconnection network, and the number of hops in the selected paths to connect sources to sub-sinks. We compare our path calculation algorithm with the well-known Dijkstra algorithm [24] to determine the shortest path between two nodes where each edge is associated with a weight. For load balancing among the sub-sinks, we also compare our method with the method presented in [12] which is based on the distance calculation to choose the nearest sub-sink. Fig. 4 illustrates an interconnection topology of 14 sources and

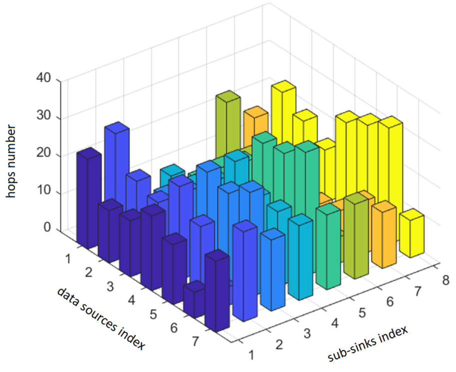


(a) I_0=1, μ=3, γ=20, 104 NR, 1744 ITER, 37.38s.

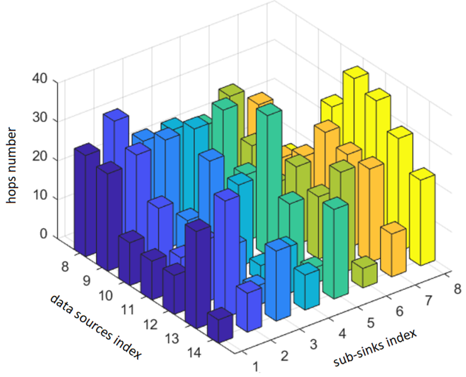


(b) I_0=2, μ=1.5, γ=20, 150 NR, 1835 ITER, 60.51s.

Figure 4: Relay nodes placement using *BlobNET*.



(a) Sources 1 to 7.



(b) Sources 8 to 14.

Figure 5: Shortest paths and their hops number.

5 sub-sinks uniformly placed in one aircraft corridor. As shown, the *BlobNET* algorithm is able to calculate a minimal structure of interconnection in a communication graph. For the results in Fig. 4a, we use as algorithm parameters I_0=1, μ=3, γ=20 and the output is of 104 relay nodes.

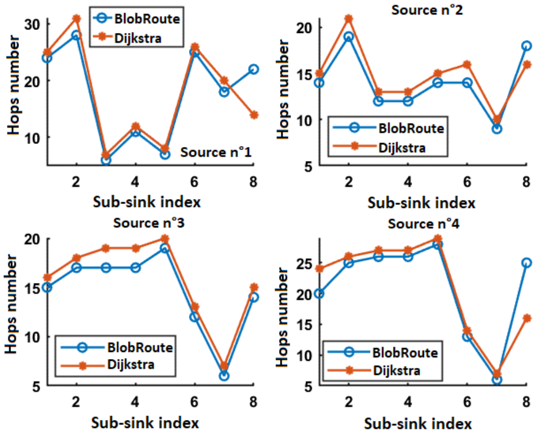
We also found that the sub-sinks throughput have an influence on the output topology such that the structure may change over simulations. For this scenario, the average throughputs are {21, 18, 16, 10, 12} kbps for sub-sinks index {277, 311, 302, 409, 232} respectively. As we can see on the Fig. 4a, thicker links reach the sub-sinks with the highest throughput while thinnest links reach sub-sinks with lower throughput.

Moreover, *BlobNET* can output a complex structure with redundant links and paths between sources if parameters are tuned into a good combination. The Fig. 4b shows an example of a complex structure as an output. We have obtained this result for I_0=2, μ=1.5, γ=20. The output graph is of 150 relays computed in 60.51 seconds. Each data source is connected to the network with two connections. In this network, we have tuned the redundancy property to define it as fault tolerance such as in [9]. With *BlobNET*, we have shown that we can connect all the uncovered data sources to the available sub-sinks, such that they may transmit their data traffic, thus the sub-sink forward to the aircraft.

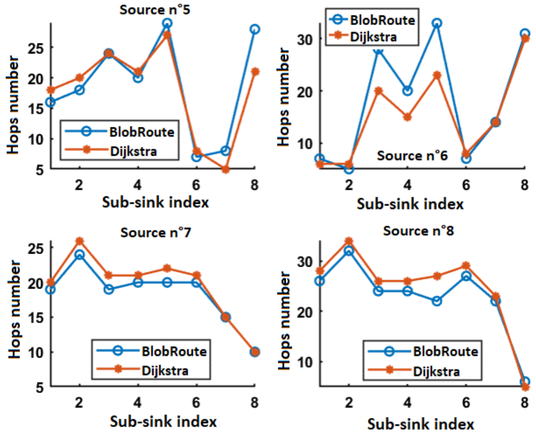
2) Path Calculation: We compute the number of hops in all the available paths in the network with the algorithm *BlobROUTE*, to finally select the shortest path. Fig. 5 shows the results for the scenario described in Fig. 3. In the left side corridor, 5 sub-sinks are placed, and in the right side only 3 sub-sinks are covered. Each one of the 14 data sources calculates all the shortest paths towards the sub-sinks, thus if a sub-sink is chosen as gateway, the forwarding route is known in advance. Fig. 6 gives the comparison of the number of nodes between *BlobROUTE* and Dijkstra algorithms for sources index 1 to 8. It shows that *BlobROUTE* is able to find shortest routes as Dijsktra does.

C. Performance

In Fig. 7, we have compared our traffic distribution method, *BlobROUTE*, to the *SDBM* method [12], respectively for the amount of queued data in the sub-sink storage (Fig. 7a), and the forwarding delay (Fig. 7b). In these figures, the abscissa are the sub-sink index. All the sub-sinks communicate with the aircraft with a data rate of 21 kbps which corresponds to a LoRa data rate using 500 kHz bandwidth, and a spreading factor of 7. However, their average throughput mainly depends on the contact time and the aircraft arrival rate[5]. Using our method, we obtained mean throughput values of 10,



(a) Sources 1 to 4.



(b) Sources 5 to 8.

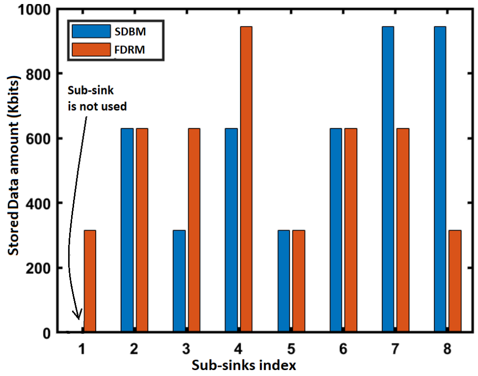
Figure 6: Comparison of the number of hops in paths computed by *BlobROUTE* and Dijkstra.

12, 18, 21, 6, 14, 16, 8 kbps respectively for sub-sink from index 1 to 8.

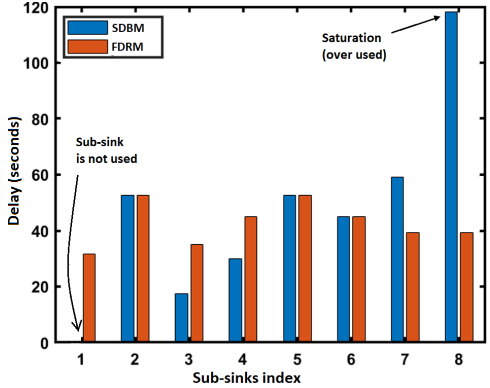
As illustrated in Fig. 7a, with the method *SDBM*, a source computes only the nearest sub-sink. We observe that no data is stored in sub-sink 1 while sub-sink 7 and 8 have almost 1000 kbps stored in their queue. *SDBM* does not perform an efficient load distribution as some nodes may be saturated while others are under-used. *BlobROUTE* overcomes this problem by considering the sub-sink capacity before forwarding process, and distributing data to every sub-sinks (Fig. 7a) more efficiently than *SDBM*. This performance is more remarkable if we consider the forwarding delay illustrated in the Fig. 7b. Indeed, despite a capacity of 10 kbps for the mobile sink, with the method *SDBM*, the sub-sink $n1$ has received nothing while sub-sink 8 needs 120 seconds to drain its 1000 kbps to the aircraft. In contrast, with our method, in less than 60 seconds, all the sub-sinks could drain their data to the aerial sink.

V. CONCLUSION

In this work, we have considered the problem of data collection in WWAs. We propose an opportunistic DTN solution in which commercial aircraft are used as MULEs. However, even if the aircraft covered areas are wide, some sources are still uncovered. Then we have proposed to efficiently deploy relays which interconnect sources to sub-sinks which are rendez-vous points covered by the aircraft. Finally, a new routing and data load balancing algorithm has been developed to make route between sources and sub-sinks, and efficiently distribute and balance the data among the sub-sinks. Our method takes into account the contact time between sub-sinks and aircraft, their mean throughput, their buffer capacity, and their position in the connected network. We have shown that our method outperforms other method used to solve the same problem. Furthermore, all the algorithms we have used have a polynomial complexity. Our future work treats the time synchronization problem between sub-sinks and aircraft.



(a) Queuing comparison.



(b) Forwarding delay comparison.

Figure 7: Improved collection method validation.

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[2] K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," Tech. Rep., 2003.

[3] S. J. Shah, S Roy and W. Brunette, "Data MULEs : Modeling a Three-tier Architecture for Sparse Sensor Networks," in *IEEE International Workshop on Sensor Network Protocols and Applications*, 2003, pp. 30–41.

[4] M. Ma and Y. Yang, "SenCar: An Energy-Efficient Data Gathering Mechanism for Large-Scale Multihop Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 10, pp. 1476–1488, 2007.
[5] I. Adamou, A. Mouradian, and V. Veque, "Towards the Use of Commercial Flights for Data Collection in Wide White Area Networks," in *IEEE WIMOB*, Barcelona, 2019, pp. 241–248.
[6] Y. N. M. Mozaffari, M. Bennis and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *IEEE Communications Surveys & Tutorials*, 2019.
[7] S. Nedenshi and R. Patra, "DTNLite: A Reliable Data Transfer Architecture for Sensor Networks," in *8th International Conference on Intelligent Engineering Systems*, 2004.
[8] M. Loubser, "Delay Tolerant Networking for Sensor Networks," Swedish Institute of Computer Science, Tech. Rep., 2006.
[9] I. Adamou, A. Mouradian, and V. Vèque, "A Bio-Inspired Deployment Method for Data Collection Networks in Wide White Areas," in *IEEE PIMRC*, 2018, pp. 1–6.
[10] S. Gao, H. Zhang, and S. Das, "Efficient Data Collection in Wireless Sensor Networks with Path-Constrained Mobile Sinks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 5, pp. 592–608, 2011.
[11] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile Unmanned Aerial Vehicles (UAVs) for Energy-Efficient Internet of Things Communications," *IEEE Transactions on Wireless Communications*, vol. 16, no. 11, pp. 7574–7589, 2017.
[12] A. Somasundara, A. Kansal, D. Jea, D. Estrin and M. Srivastava, "Controllably mobile infrastructure for low energy embedded networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 8, pp. 958 – 973, 2006.
[13] J. Y. Chengliang Wang and S. Das, "Efficient Aerial Data Collection with UAV in Large-Scale Wireless Sensor Networks," *International Journal of Distributed Sensor Networks*, vol. 11, no. 11, 2015.
[14] N. Sornin, M. Luis, T. Eirich, T. Kramp, and O.Hersent, "LoRaWAN Specification," LoRa Alliance, Tech. Rep., 2015.
[15] A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.
[16] S. Corporation, "LoRa Modulation Basics," Semtech Corporation, Tech. Rep., 2015.
[17] D. Vogel and A. Dussutour, "Direct Transfer of Learned Behaviour via Cell Fusion in Non-Neural Organisms," *Proceedings of the Royal Society B: Biological Sciences*, vol. 283, 2016.
[18] A. Tero, T. Nakagaki, K. Toyabe, K. Yumiki, and R. Kobayashi, "A Method Inspired by Physarum for Solving the Steiner Problem," *International Journal of Unconventional Computing*, 2010.
[19] T. Nakagaki, R. Kobayashi, Y. Nishiura, and T. Ueda, "Obtaining Multiple Separate Food Sources: Behavioural Intelligence in the Physarum Plasmodium," *The Royal Society*, 2004.
[20] I. Adamou, https://sites.google.com/view/inchaadamou/blob, Tech. Rep., 2021.
[21] A. Tero, R. Kobayashia, and T. Nakagaki, "A Mathematical Model for Adaptive Transport Network in Path Finding by True Slime Mold," *Journal of Theoretical Biology*, 2007.
[22] C. Goursaud and J.-M. Gorce, "Dedicated Networks for IoT : PHY / MAC State of the Art and Challenges," *EAI Endorsed Transactions on the Internet of Things*, 2015.
[23] "MATLAB Software," 2020.
[24] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. MIT Press et McGraw-Hill, 2001.