Official URL

DOI : https://doi.org/10.1109/IWCMC.2018.8450473

To cite this version: Mnasri, Sami and Nasri, Nejah and Val, Thierry *The 3D indoor deployment in DL-IoT with experimental validation using a particle swarm algorithm based on the dialects of songs.* (2018) In: 14th International Wireless Communications and Mobile Computing Conference (IWCMC 2018), 25 June 2018 - 29 June 2018 (Limassol, Cyprus).

The 3D indoor deployment in DL-IoT with experimental validation using a particle swarm algorithm based on the dialects of songs

Sami MNASRI*, Nejah NASRI †, Thierry VAL*
*University of Toulouse, UT2J, CNRS-IRIT-RMESS, Toulouse, France
†University of Sfax, ENIS, LETI, Sfax, Tunisia
Sami.Mnasri@irit.fr, nejah.nasri@isecs.rnu.tn, val@irit.fr

***Abstract*— The use of real prototyping systems allows implementing real-world deployments which permit evaluating new protocols, algorithms and network solutions. This study investigates the problem of 3D indoor redeployment of connected objects in IoT collection networks. The objective is to choose the right positions in which connected objects are added to an initial configuration, while optimizing a set of objectives. To solve this problem, a novel bird's dialect-based particle swarm optimization algorithm (named acMaPSO) is introduced. The new concept of bird's dialect is based on a set of birds which are separated into different dialect groups by their regional habitation and are classified into groups according to their common manner of singing. The obtained numerical results and the real experiments on our testbed prove the effectiveness of the two proposed variants compared with the standard PSO algorithm and a recent state of art of many-objective evolutionary algorithms: the NSGA-III.**

Keywords—3D indoor deployment, DL-IoT, experimental validation, dialect based PSO, many-objective optimization.

I. Introduction

Node deployment defines how to position a set of nodes and the topology of the network used to deploy these nodes. In this paper, we are interested in the three-dimensional deployment that reflects the topology of the RoI (region of interest) better than the two-dimensional deployment. Specifically, our aim is to resolve the issue of redeployment where the initial indoor 3D deployment is improved by adding new nodes to optimize various objectives such as network lifetime, coverage, energy consumption and localization.

The DL-IoT (Device Layer - Internet of Things) is the evolution of WSN (Wireless Sensor Networks) to IoT networks. DL-IoT are collection networks relying on nodes called connected objects used for the collection of information. In this study, we aim to deploy a 3D indoor DL-IoT network. It is a scenario in which autonomous entities (devices, robots, or people with sensors) with unique identifiers can interact together using network protocols such as Bluetooth or 802.15.4. WSN and IoT are complementary: the WSN is responsible for the hardware communication and the transmission of the physical values detected by the sensors. While the IoT provides the decision making after manipulating the collected data. Our approach is applicable in both contexts (IoT and WSN).

Recently, there is a growing trend to evaluate the performance of new network platforms and solutions via real prototyping testbeds. The advantage of real experiments compared to simulations is manifested in the simplicity of prototyping communication devices and in the realism of the equipment and the obtained results. To this is added the advantage of the human experience feedback. As examples of these real prototyping platforms having thousands of nodes, FIT/IoT-LAB [1] (formerly called SensLab) and SmartSantander platform [2] for smart cities. Other platforms like INDRIYA [3] and TWIST [4] allow deploying, on several levels, about 200 nodes. The mentioned testbeds share a physical layer relying on protocols standardized by the IEEE 802.15.4-2006 with a frequency of 868 MHz or 2.4 GHz. Contrary to classical works based on theoretical hypotheses, simulations and formal calculations, we aim to finely characterize the real world with physical nodes of our prototyping platform. This platform is presented in the Experiments section.

To identify the best positions of the connected objects, while optimizing a set of opposed objectives and constraints, a modified PSO (Particle Swing Optimization) algorithm based on a new concept of bird's dialects is used.

The major contributions of this study are as follows:

- The proposal of the acMaPSO which is a modified MaOPSO (many-objective PSO) algorithm that proposes a new concept of bird's dialects on the PSO. Indeed, it is a specific concept that reflects the particle experience to evaluate the experience of each particle in the swarm.
- We propose a real experimental validation of the indoor 3D deployment using a real testbed. The proposed algorithm (acMaPSO) is compared with MaOPSO and NSGA-III. Another comparison is also made between the results of the simulations and the real experiments. The interpretation of the obtained results is also provided.

Next, the following sections will be detailed: Section II discusses and interprets a set of related works. Section III presents the concept of bird dialects on the particle swarm algorithm. Section IV illustrates the numerical results. Section

V details a set of experiments on testbeds and compares them with simulations. Section VI shows a conclusion and different possible perspectives.

II. RELATED WORKS ON THE 2D-3D DEPLOYMENT PROBLEM

This section presents recent works proposing optimization algorithms for efficient node deployment. Banimelhem et al. [5] introduced a genetic algorithm (GA) to find the deterministic 2D deployment in WSN with consideration of coverage holes while minimizing the number of used mobile nodes. However, this study lacks a mathematical modeling that explains the details of the problem. Unaldi et al. [6] propose a GA based on a guided wavelet transform and a random mutation for the probabilistic deployment of WSN nodes in the context of 3D terrains. This study aims to minimize the number of sensors and maximize the quality of coverage. On the other hand, the proposed algorithm is evaluated only with stationary sensors, without empirical scenarios on a real-world problem. Danping et al. [7] propose a low-cost heuristic combined with an evolving multi-objective algorithm for solving the 3D deployment problem taking into account the propagation of the radio signal in indoor. The goal is to simultaneously improve the network life, the hardware cost, the coverage, and the link quality. Although, the authors have not demonstrated the scalability of the proposed approach with a high number of nodes. Ko et al. [8] resolve the deployment in irregular 3D terrains using an analysis crossover GA to simultaneously maximize the global coverage and probabilistic point coverage. Yet, no evidence is given regarding the effectiveness of the proposed crossover strategy compared to the original genetic approach. In [9], an algorithm based on a harmony search is proposed for the optimization of the coverage and the number of deployed sensors. The limits of this work lie in the proposed network model which is simplistic. Besides, in the consideration of only two objectives and in the validation of the approach based only on Matlab tests without real simulation or experimentation scenarios. The authors in [10] suggest a hybrid algorithm called AcNSGA-III that hybridizes the Ant Colony Optimization with the NSGA-III to solve the problem of 3D indoor deployment. They proved the effectiveness of the proposed algorithm compared to the standard ACO and NSGA-III algorithms. However, the applicability of this algorithm in dense networks is not proven and the used energy model is simplistic.

III. THE PROPOSED ACMAPSO ALGORITHM: INCLUDING THE CONCEPT OF BIRD'S DIALECT ON THE MAOPSO

The suggested modifications on the standard multi-objective PSO aim to avoid the difficulties encountered by this algorithm when solving real-world problems that are generally complex and have several local optima. These modifications rely on introducing changes in the topology of the swarm. Indeed, to avoid the premature character of convergence of the standard PSO [11], in addition to the two positions used in this algorithm (the best overall position (gbest) and the best personal position of the particle (pbest)), we proposed the best position of the local area around the particle, called: best cluster (cbest).

Recent research in biology [12] affirmed that songbirds have regional dialects such as humans. In fact, birds inherit from their parents the ability to sing and create a complete song. These biological studies have shown that if birds are bred in silence, they do not acquire this ability to sing and can only shout. Even more, birds from different regions develop distinct dialects. Following this biological finding, we propose an algorithm (called acMaPSO) which consists in a PSO relying on a topology of different categories of songbird dialects. Indeed, each dialect group has different convergence acceleration parameters, which contributes to the prevention of local optima. Moreover, the introduced concept of dialects helps to assess the particle search capabilities in their local areas where particles belong to different communities (groups or swarms). Fig. 1 shows a set of particles separated into groups according to their dialects during the process of searching for solutions. In order to keep the diversity of the population, particles in each dialect category can select their neighbors only from the least experienced particles of their own group or from other groups.

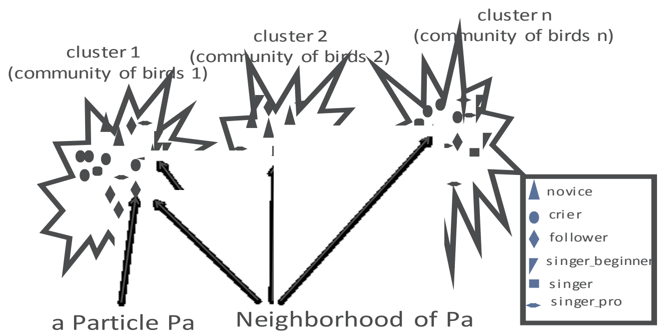

Fig. 1 The Neighborhood of a particle Pa

The acMaPSO algorithm is shown in algorithm1.

Algorithm 1 The acMaPSO algorithm

```
Initialize the parameters: n (number of the swarms), k (population size of
each swarm), w (inertia weight), c1 and c2 (acceleration coefficients)
Set the fitness evaluations counter Cf = 0
Set the initial accent ai of initial swarms to novice
/*Random generation of n initial swarms si∈S with size P, (i = 1, 2, . . , P) */
for i=1 to n
   Initialize parameters of the swarm i
   for j=1 to k
        Initialize position and velocity
        Calculate fitness
   end for
end for
while (stop condition is not fulfilled [sufficient good fitness | maximum
number of iterations]) do
      if highest_experience_category is reached then
          Delete particles having reached the highest experience category and
insert new random particles (accent operator)
        Construct accent-group topology structure (S,a)
      end if
      if fitness if not evolved (ameliorated) in recent µ iterations then
         Move on to the next category
      end if
     for i=1 to n
        for i = 1 : P do
           Adjust the inertial weight (S; w; a)
```

```
            Adjust the acceleration constant (S; c1; c2; a)
            Update position and velocity
            Fitness evaluation (S; Cf)
            If fp is better than f(pBest) then pBest = p
        end for
      cbest_i = best pbest_i in P  //updating cbest_i (the cbest of the swarm i)
    end for
    Update accent of particles
      gBest = best cbest_i in P  //updating gbest_i (the gbest of all the swarms)
end while
```

IV. NUMERICAL RESULTS

This section presents the used parameters of the algorithms and the obtained numerical results. The HV (Hypervolume) [13] is used as a metric to evaluate the quality of the results. Despite its high computing cost, the HV is ideal for real-world problems having generally a true unknown Pareto front. To have an idea about the influence of nomad node positions on network performance, acMaPSO is compared to the NSGA-III [14] which is another recent multi-objective optimization algorithm. The PlatEMO platform [15] is used for the implementation of NSGA-III, MaOPSO and acMaPSO. The details of the parameters of these algorithms are shown in Table I.

TABLE I. PARAMETERS SETTING OF THE USED ALGORITHMS

Parameter			Value
Number of objectives			Variable, see Table II
Number of independent runs			25, on different initial populations
Number of constraints			7
Population size			300
Maximum number of generations			Variable, see Table II
NSGA-III parameters	Recombination	Operator	SBX
		probability	0.8
		distribution index	45
	Mutation	Operator	Bit-flip
		probability	1/400
		Operator	SBX
PSO parameters	Inertial weight		0.95 to 0.4
	Cognitive components (C1)		2.8 to 2.2
	Social components(C2)		1.2 to 1.8
	Number of particles per swarm		10-50
	Initial minimum number of clusters		4
	Initial swarm particle velocity		distributed in [-4, 4] randomly

To obtain a statistically reliable comparison of results, the optimization algorithms must be run several times for each test because of the random behaviors of these algorithms. In our tests, an average of 25 executions are achieved for each value. Table II illustrates the average HV for different number of generations and objectives. Higher HV have better performance.

TABLE II. BEST, AVERAGE AND WORST HYPERVOLUME VALUES

Obj Nbr	Max nbr of generations	MaOPSO	NSGA-III	acMaPSO
3	1300	0.903458 0.902896 0.898023	0.902231 0.901658 0.898235	0.903631 0.903036 0.902563
4	1800	0.976985 0.976833 0.975612	0.974892 0.974743 0.973897	0.977331 0.977098 0.976892
5	2600	0.972892 0.972116 0.971084	0.972983 0.972563 0.972126	0.972985 0.972728 0.972436

The results in Table II affirm that, for different numbers of objectives, acMaPSO is often the most efficient algorithm. MaOPSO is more efficient than NSGA-III, but it has a higher relative degradation compared to other algorithms when increasing the number of objectives.

V. SIMULATION AND EXPERIMENTAL RESULTS

In what follows, a comparison is made between two scenarios: one is for simulations and the other is for experimental tests. The behavior and performance of the proposed acMaPSO are compared to those of NSGA-III and MaOPSO. Nowadays, there is a tendency to propose and test algorithms and protocols with real environments since simulators and theoretical analysis do not perfectly reproduce the physical and technical characteristics of the real environment. Hence, with our prototyping testbed, we aim to reduce discrepancies between practice and theory in IoT and WSN deployment. The use of a personal testbeds (such as ours: Ophelia) gives several advantages such as the ease of use, the reproducibility of results, the human feedback, the realism of conditions and the heterogeneity of nodes. This makes such testbeds ideal for IoT components.

A. Description of the testbed

1) TeensyWiNo deployed nodes: In our tests, we used TeensyWiNo based-on WiNoRF22 nodes. Since they are integrated into the Arduino system, these nodes allow researchers to easily incorporate software and hardware modules such as interaction devices, actuators, sensors, processing algorithms or prototyping solutions supporting users feedback. The technical characteristics of the deployed TeensyWiNo nodes are detailed in Table III.

TABLE III. TECHNICAL CHARACTERISTICS OF TEENSYWINO

Use	IoT, WSN
Transceiver (Arduino libraries)	HopeRF-RFM22b 200-900 MHz; 1-125kbps; GFSK/FSK/OOK; +20dBm Radio-Head
CPU/RAM/Flash	ARM-Cortex-M4 [32bit]-72MHz; 64kB-RAM, 256kB; Flash (PJRC-Teensy-3.1)

The components and an example of the deployed TeensyWiNo nodes are shown in Fig. 2.

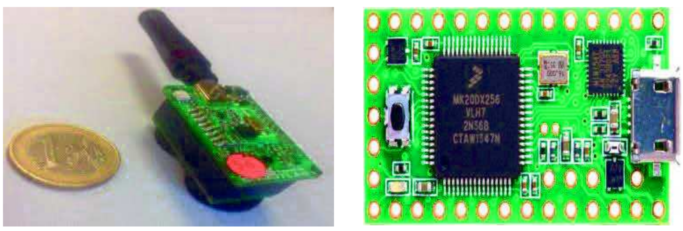

Fig. 2 The deployed Teensy WiNo nodes

*2) **OpenWino*** [16] *(Open Wireless Node):* is a free development environment for DL-IoT collection networks and WSN protocol engineering. It helps achieve rapid prototyping of MAC, NWK or other layer protocols. It also allows to evaluate the performances of these protocols. The simplicity of OpenWiNo is one of its advantages: indeed, the change of the physical layer of a WiNo node for example, is done by simply changing the transceiver (and its associated driver). An open hardware environment requires this ease of use. Among the transceivers that have been successfully tested on OpenWiNo: Proprietary 433MHz FSK/GFSK (HopeRF RFM22b), IEEE 802.15.4-2011 UWB (DecaWave DW1000), Classical IEEE 802.15.4 2.4GHz DSSS (Freescale), LoRa mode 868MHz (HopeRF RFM95).

*3) **Ophelia:*** it is our testbed based on a web interface, Openwino, Arduino and Teensywino installed nodes. The web interface allows to remote access the Ophelia testbed, the programming of experimental sketches (in python) and their execution on the nodes.

Fig. 3 illustrates the indoor deployment of nodes in one of the six used sites.

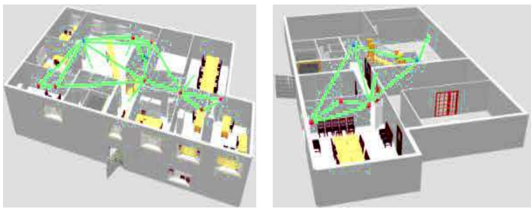

Fig. 3 The 3D deployment in one of the six sites

The simulations are performed using OMNeT ++, a platform for developing and simulating network protocols. Fig. 4 illustrates the interface of our OMNeT++ simulation scenario showing the distribution of the nodes.

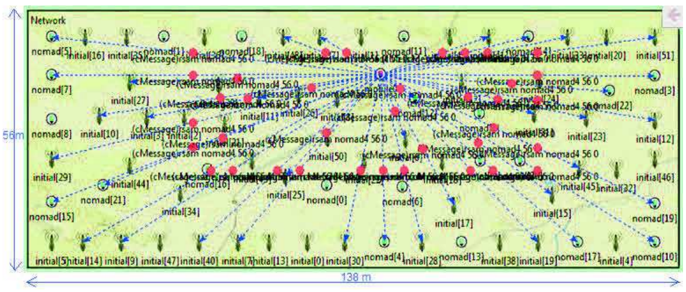

Fig. 4 The distribution of nodes in the simulation scenario

B. Simulation and experimental parameters

A 3.5 Ghz i5-6600K Core computer is used to test the algorithms. The implemented physical layer is 433 MHz, with an uncoordinated CSMA/CA (IEEE 802.15.4) access method and an AODV (ad hoc distance vector on demand) routing protocol. Table IV details the parameters used in our simulations and experiments.

TABLE IV. PARAMETERS USED ON SIMULATIONS AND EXPERIMENTS

Repartition of nodes	6 sites on 200 * 200 m²
Number of nodes	36 (1 mobile, 6 nomad, 29 fixed)
Average of runs	25 experiments
Simulation period	10800 seconds
Transmission power	100 mW
Bit rate	256 kbps
Indoor sensing range	8m
Modulation model	125 kbit/s GFSK
Message-length	16
Message-number	1000
Message-wait	5
Frequency	434.79 MHz
Operating temperature	25°c
Tx power	7 (the max of RFM22)
FER (Frame Error Rate)	0.01 (initially)
RSSI	100 (initially)
Reception gain	50 mA
Indoor transmission range	7m
Antenna model	transceiver RFM22
Modem configuration	12 # GFSK_Rb2Fd5

C. Comparing the experimental results to the simulations

*1) Simulation scenario**:*** The simulation scenario is as follows: A trigger node (the mobile node) sends an initial message to a random destination ***d***. when this node ***d*** is found by the AODV routing protocol, it becomes the source node and selects a new destination node. This process is repeated until the maximum simulation time is reached. To be able to compare experiments to simulations, we use the same scenario and architecture (type and number of nodes) in both cases. The initial distribution of the fixed nodes is chosen according to the distribution law of OMNeT++. This law evenly distributes the nodes from the center of the RoI. The connectivity matrix is based on empirical experiments by establishing the initial connectivity links between the nodes based on experiments. To ensure dynamism and new connectivity relationships between nodes during simulations, we introduce disruptions to the RSSI connectivity links. Indeed, a perturbation (+/-30 for each value) is performed on the RSSI matrix.

2) Experimental scenario: In the Ophelia testbed, 30 fixed nodes are used which are initially deployed and having known positions. These positions are determined according to the application needs of the users. We aim to add six nodes called nomad nodes. The positions of these last nodes are determined using the tested optimization algorithms. Only one mobile node is used. The execution of the experimental scenario will be as follows: initially, the nodes are flashed. Then, the initial configuration parameters (transmission power, etc.) are sent to the nodes. Afterads, we choose a node to send a first broadcast to all other nodes. The measures of the RSSI and FER are taken in two directions: the sending node records its FER and RSSI rates with each receiving node which also returns these same measurements. After a predetermined wait time, this sending node terminates the process. Subsequently, the sender is changed and other nodes become receivers. We repeat the same process for all nodes (36 experiments), to obtain two connectivity matrices of the FER and RSSI values between all the nodes. From these two matrices, we can deduce the neighbors of each node. We consider a node ***i*** to be the neighbor of another node ***j*** if the average of the RSSI emitted between these two nodes (from ***i*** to ***j*** and and from ***j*** to ***i***) exceeds a threshold set at 100; and the average FER is also below a threshold set at 0.1. Given the need for a statistical test to compare two algorithms and taking into account the stochastic nature of evolutionary algorithms, the average values taken in this experimental scenario are the result of 25 execution time for each value.

3) Compaison of the RSSI rates : To evaluate the localization and the cost of deployment, the RSSI metric is used because the localization is based on the Distance-VectorHop protocol, to which the RSSI information is added. Indeed, the localization is proportional to the RSSI rate. Fig. 5 shows the average rates of the RSSI of the nodes in connection with the mobile node, for different number of objectives. These objectives are to be satisfied by the tested algorithms. This average RSSI is a value (convertible in dBm) between 0 and 256.

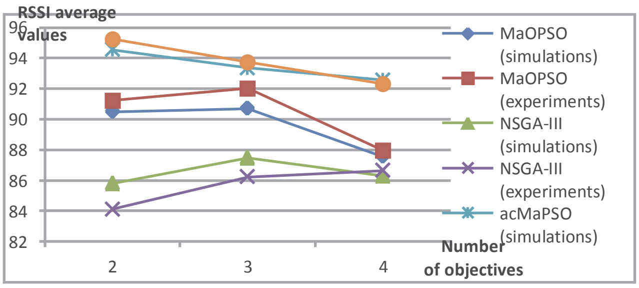

Fig. 5 Average RSSI rates of nodes connected to the mobile node

4) Comparison of the FER rates : FER is used as a metric to assess coverage and link quality between nodes. Indeed, the FER is inversely proportional to the coverage. Fig. 6 shows the average values of the FER of the nodes in connection with the mobile node, for a variable number of objectives.

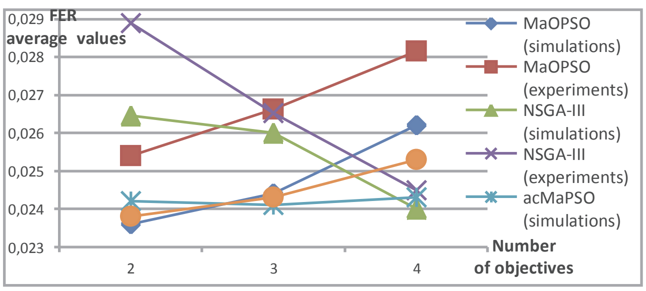

Fig. 6 FER average rates of nodes connected to the mobile node

5) Comparison of the number of neighbors : The average number of neighbors of nodes in connection with the mobile node is used as the metric to evaluate the network utilization rate and the network connectivity. We use the same notion of neighborhood that was previously explained in the experimental section. Fig. 7 shows the average number of neighbors of the nodes in connection with the mobile node, for a variable number of objectives.

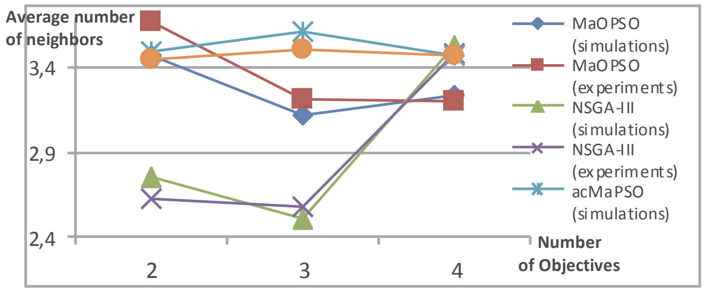

Fig. 7 Average of neighbors of the nodes connected to the mobile node

6) Comparison of the network lifetime and the energy consumption: Fig. 8 illustrates the changes in the energy consumption (as a function of time). Indeed, we measure the average of the indicator of energy of the nodes after the addition of the nomad nodes.

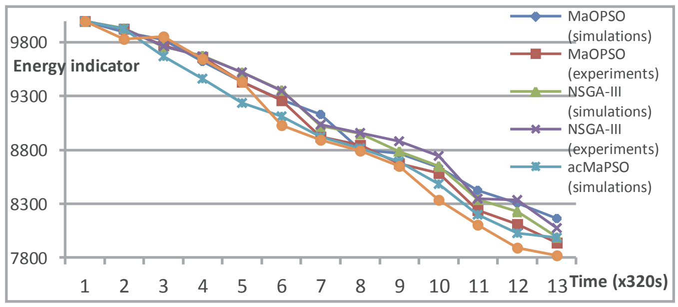

Fig. 8 Comparaison of the average levels of energy consumption

The averages of network lifetimes (calculated for different algorithms and objectives) are shown in Table V. The lifetime metric used is measured by the time (in seconds) after which the first node goes out of service.

TABLE V. COMPARING THE AVERAGE LIFETIME

Algorithm \ Nbr of objectives		2	5
MaOPSO	Simulations	3546	3478
	Experiments	3502	3469
NSGA-III	Simulations	3485	3528
	Experiments	3487	3546
acMaPSO	Simulations	3543	3540
	Experiments	3549	3553

D. Discussion and interpretations

After the evaluation of the experiments, several interpretations can be deduced, among others:

- Unlike RSSI, FER rates are lower in simulations than in experiments.
- Unlike other algorithms, the NSGA-III in the experiments has lower RSSI values than simulations.
- FER values are lower during night than day. This is due to human activities within the building during the day (opening and closing doors for example). These activities cause the signal disruption.

- Contrary to what is assumed, FER and RSSI values are not always inversely proportional. Indeed, a connection of two nodes may have, at the same time, a high FER and an excellent RSSI.
- The NSGA-III is evaluated by their authors only on instances of theoretical test problems. Our experiment is proof of the advantage of applying the NSGA-III in real-world contexts.
- Several studies such as [17], prove that MaOPSO is better than NSGA-III. In keeping with this, our numerical results (based on the HV metric) state that MaOPSO is not surpassed by the NSGA-III. Moreover, the carried out experimental results show that the proposed acMaPSO algorithm is generally better than the NSGA-III on the FER and RSSI rates (therefore, acMaPSO is more effective than NSGA-III in optimizing the localization, the quality of links and the coverage). While the NSGA-III is generally more efficient than the acMaPSO in satisfying the number of neighbors of nodes (consequently, it is more efficient in satisfying the network utilization). The experimental results are not in contradiction with the numerical results but this is explained by the fact that the 3D indoor deployment is a real problem which is different from the theoretical test problems used to evaluate the algorithms.

VI. Conclusion

In this paper, we proposed a real world deployment experiment based on prototyping on real nodes of an OpenWiNo-based testbed (Ophelia) to solve the problem of 3D indoor deployment of a DL-IoT. The proposed resolution approach is based on a new variant of the PSO algorithm: the acMaPSO which includes a new concept of dialect to avoid local optima. The proposed algorithm achieves (and surpasses for certain evaluation metrics such as the number of neighbors), the performance of the standard PSO and NSGA-III algorithms. Nevertheless, different improvements can be proposed for this study. Among others, supporting some other technologies and protocols of transmission by implementing them on OpenWiNo which has the shortcoming of lack of libraries implementing the standard protocols. Moreover, although Ophelia testbed is more realistic than a platform with a large number of uniform nodes such as IoTLab [1], SmartSantander [2] or INDRIYA [3], these latter platforms allow scaling up and testing our approach with a greater number of nodes (up to 1024 nodes). Since the IoTLab allows to test the same metrics of our experiments (RSSI, link quality...), tests on this latter platform are envisaged in future works to prove the scalability of our approach and compare its results with Ophelia ones.

References

[1] E. Fleury, N. Mitton, T. Noel and C. Adjith, "FIT IoT-LAB: The Largest IoT Open Experimental Testbed," ERCIM News, ERCIM, 2015, pp.4.

[2] L. Sanchez, L. Muñoz, J. A. Galache, S. Pablo, J. R. Santana, V. Gutierrez, et al., "SmartSantander: IoT experimentation over a smart city testbed," Computer Networks, vol. 61, 2014, pp. 217-238, doi:10.1016/j.bjp.2013.12.020

[3] M. Doddavenkatappa, M.C. Chan and A.L. Ananda, "Indriya: A Low-Cost, 3D Wireless Sensor Network Testbed," In: Korakis, T. Li, H. Tran-Gia, P. Park, HS., (eds) Testbeds and Research Infrastructure. Development of Networks and Communities, TridentCom, 2011. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 90, Springer, Berlin, Heidelberg, doi:10.1007/978-3-642-29273-6_23

[4] H. Vlado, K. Andreas, W. Andreas and W. Adam, "TWIST: a scalable and reconfigurable testbed for wireless indoor experiments with sensor networks," 2nd international workshop on Multi-hop ad hoc networks: from theory to reality (REALMAN '06), ACM, Florence, Italy, May 24-26, 2006, pp. 63-7, doi:10.1145/1132983.1132995

[5] O. Banimelhem, M. Mowafi and W. Aljoby, "Genetic Algorithm Based Node Deployment in Hybrid Wireless Sensor Networks," Communications and Network, vol. 5 no. 4, pp. 273-279, November 2013, doi: 10.4236/cn.2013.54034.

[6] N. Unaldi, S. Temel and V.K. Asari, "Method for optimal sensor deployment on 3D terrains utilizing a steady state genetic algorithm with a guided walk mutation operator based on the wavelet transform," Sensors, vol. 12, no.4. p. 5116-5133, 2012, doi: 10.3390/s120405116

[7] H. Danping, J. Portilla and T. Riesgo, "A 3D multi-objective optimization planning algorithm for wireless sensor networks," 39th Annual Conference of the IEEE Industrial Electronics Society, IECON 2013, pp.5428-5433, November 2013, doi: 10.1109/IECON.2013.6700019.

[8] A.H.R Ko and F. Gagnon, "Process of 3D wireless decentralized sensor deployment using parsing crossover scheme," Applied Computing and Informatics, vol. 11, issue 2, July 2015, pp. 89–101, doi: 10.1016/j.aci.2014.11.001.

[9] O.M. Alia and A. Al-Ajouri, "Maximizing Wireless Sensor Network Coverage With Minimum Cost Using Harmony Search Algorithm" IEEE Sensors Journal, vol. 17, no. 3, pp. 882-896 (2017), doi: 10.1109/JSEN.2016.2633409

[10] S. Mnasri, N. Nasri, A. Van Den Bossche and T. Val, "A Hybrid Ant-Genetic Algorithm to Solve a Real Deployment Problem: A Case Study with Experimental Validation," In: Puliafito A., Bruneo D., Distefano S., Longo F. (eds) Ad-hoc, Mobile and Wireless Networks. ADHOC-NOW 2017. Lecture Notes in Computer Science, vol 10517. Springer, Cham, doi: 10.1007/978-3-319-67910-5_30

[11] J. Kennedy and R. Eberhart, "Particle swarm optimization," IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948, 1995, doi: 10.1109/ICNN.1995.488968

[12] A. Oesel, A.C. Fries, L. Miller, H.L. Gibbs, J.A. Soha and D.A. Nelson, "High levels of gene flow among song dialect populations of the Puget Sound white-crowned sparrow," Ethology, 2017, vol. 123, pp. 581–592, doi: https://doi.org/10.1111/eth.12632

[13] L. While, P. Hingston, L. Barone and S. Huband, "A faster algorithm for calculating hypervolume," IEEE Transactions on Evolutionary Computation, vol. 10, no. 1, January 2006, pp. 29-38, doi: 10.1109/TEVC.2005.851275

[14] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference- point-based nondominated sorting approach, part I: Solving problems with box constraints," IEEE Transactions on Evolutionary Computation, vol. 18, no. 4, September 2013, pp. 577-601, doi: 10.1109/TEVC.2013.2281535

[15] Y. Tian, R. Cheng, X. Zhang and Yaochu Jin, "PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization," IEEE Computational Intelligence Magazine, 2017, vol. 12, no. 4, pp. 73-87, arXiv:1701.00879v1 [cs.NE]

[16] A. Van Den Bossche, R. Dalce and T. Val, "OpenWiNo: An Open Hardware and Software Framework for Fast-Prototyping in the IoT," 23rd International Conference on Telecommunications (ICT), Thessaloniki, Greece, May 2016, pp. 1-6, doi: 10.1109/ICT.2016.7500490

[17] E. M. N. Figueiredo, D. R. B. Araújo, C. J. A. B. Filho and T. B. Ludermir, "Physical Topology Design of Optical Networks Aided by Many-Objective Optimization Algorithms," 2016 5th Brazilian Conference on Intelligent Systems (BRACIS), Recife, 2016, pp. 409-414, doi: 10.1109/BRACIS.2016.080