ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/334385905

Nature-inspired metaheuristics for optimizing information dissemination in vehicular networks

Conference Paper · July 2019

DOI: 10.1145/3319619.3326847

CITATIONS
0

READS
93

5 authors, including:

Antonio David Masegosa
Universtiy of Deusto/ Ikerbasque - Basque Foundation for Science
60 PUBLICATIONS **358** CITATIONS
SEE PROFILE

Eneko Osaba
Tecnalia Corporación Tecnológica
109 PUBLICATIONS **719** CITATIONS
SEE PROFILE

Juan S. Angarita-Zapata
University of Deusto
8 PUBLICATIONS **6** CITATIONS
SEE PROFILE

Ibai Laña
Tecnalia
23 PUBLICATIONS **129** CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project
Causal structure built with system dynamics to model induced travel demand and facilitate policy-making to mitigate traffic congestion View project

Project
Cyberparks View project

All content following this page was uploaded by Eneko Osaba on 19 July 2019.

The user has requested enhancement of the downloaded file.

Nature-Inspired Metaheuristics for optimizing Information Dissemination in Vehicular Networks

Antonio D. Masegosa
University of Deusto, Bilbao, Spain
Ikerbasque, Basque Foundation for Science, Bilbao, Spain
ad.masegosa@deusto.es

Eneko Osaba
TECNALIA Research and Innovation, Derio, Spain
eneko.osaba@tecnalia.com

Juan S. Angarita-Zapata
University of Deusto, Bilbao, Spain
js.angarita@deusto.es

Ibai Laña
TECNALIA Research and Innovation, Derio, Spain
ibai.lana@tecnalia.com

Javier Del Ser
University of the Basque Country (UPV/EHU), Bilbao, Spain
javier.delser@tecnalia.com

ABSTRACT

Connected vehicles are revolutionizing the way in which transport and mobility are conceived. The main technology behind are the so-called Vehicular Ad-Hoc Networks (VANETs), which are communication networks that connect vehicles and facilitate various services. Usually, these services require a centralized architecture where the main server collects and disseminates information from/to vehicles. In this paper, we focus on improving the downlink information dissemination in VANETs with this centralized architecture. With this aim, we model the problem as a Vertex Covering optimization problem and we propose four new nature-inspired methods to solve it: Bat Algorithm (), Firefly Algorithm, Particle Swarm Optimization, and Cuckoo Search. The new methods are tested over data from a real scenario. Results show that these metaheuristics, especially Bat Algorithm, Firefly Algorithm and Particle Swarm Optimization, can be considered as powerful solvers for improving information dissemination in VANETs.

CCS CONCEPTS

• **Applied computing → Transportation**; • **Theory of computation → Random search heuristics**; **Evolutionary algorithms**; *Packing and covering problems*;

KEYWORDS

Nature-inspired Metaheuristics, Vertex Covering, Vehicular Communications, VANETs, Inteligent Transportation Systems

ACM Reference Format:

Antonio D. Masegosa, Eneko Osaba, Juan S. Angarita-Zapata, Ibai Laña, and Javier Del Ser. 2019. Nature-Inspired Metaheuristics for optimizing Information Dissemination in Vehicular Networks. In *Genetic and Evolutionary Computation Conference Companion (GECCO '19 Companion), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3319619.3326847

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
GECCO '19 Companion, July 13–17, 2019, Prague, Czech Republic
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6748-6/19/07…$15.00
https://doi.org/10.1145/3319619.3326847

1 INTRODUCTION

Transport and mobility are immersed in a major transformation nowadays thanks to the confluence of several innovations: autonomous driving, shared mobility, electric vehicle, and connected vehicle. Regarding this last aspect, the main technology behind is the so-called Vehicular Ad-Hoc Networks (VANETs), which are communication networks in which the nodes are vehicles [18]. Depending on which element is the transmitter and the receptor in each extreme of the communication, either Infrastructure or Vehicle, VANETs can be categorized into Infrastructure-to-Infrastructure (I2I), Infrastructure-to-Vehicle (I2V), Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V). These communication configurations can enable different services related to security, leisure and entertainment, traffic management and driving assistance [30].

Many of the aforementioned services require a centralized architecture in which a central server collects and disseminates data from/to vehicles [5] (e.g. to acquire information about the instant position, speed and heading of all vehicles on a region or to warn vehicles closed to a potentially dangerous situation). Within this scenario, in this paper we focus on the so-called down-link communication in which a central server disseminates information to vehicles. Different aspects make this task challenging in VANETs [18]. One of the most promising approaches proposed so far to address this challenge is the use of Virtual Infrastructure (VI) [10, 34], wherein vehicles are also used as infrastructure to increase the area of application and to reduce the deployment cost. The VI is composed of three phases: 1) some vehicles are selected as cluster-head (CH) to cover the broadcast area, 2) CH vehicles receive information from a central server through I2V, and 3) CH vehicles disseminate this information to nearby vehicles using V2V communications.

The process of deciding how many and what vehicles should be used as VI in each instant plays a pivotal role in this type of systems, since it can avoid the necessity of fixed infrastructure (as Road Side Units), reduce the network overload, or affect the communication quality [34]. Many of the services envisioned for VANETs require the dissemination of some messages to all or as much as possible vehicles in a specific geographical area. In this context, the priority of the VI selection phase must be to choose the minimum number of CH vehicles that can broadcast the information to the maximum number of vehicles in the target area, taking into account all the requirements imposed by the corresponding standard [5].

Based on the communication architecture for VANETs presented in [10] and called NAVI (Neighbor-Aware Virtual Infrastructure), in a previous work, Masegosa et. al [20] proposed a new approach to address the VI selection process, modelling it as the optimization of a Covering Location optimization Problem (CLP)[11]. Concretely, they modeled those vehicles that must receive the information as demand nodes, and those vehicles that play the role of VI as potential facility locations. Using a Genetic Algorithm (GA, [12]) as optimization method, they were able to outperform an ad-hoc state-of-the-art methodology designed for the same purpose.

In this paper, our objective is to continue deepening the research line proposed in [20] through the development of better optimization methods based on nature-inspired metaheuristics. Specifically, we propose and compare four nature-inspired metaheuristics to improve the performance of this approach for information dissemination in VANETs. The algorithms considered for such purpose are: Bat Algorithm (BA, [38]), Firefly Algorithm (FA, [37]), Particle Swarm Optimization (PSO, [15]), and Cuckoo Search (CS, [39]). The main rationale behind the use of these nature-inspired metaheuristics relies in their reputed efficiency for solving other combinatorial optimization problems along recent years. Some examples of successful application cases can be found in [23, 25, 27, 28]. The good acceptance of these previous studies have led us to the hypothesis that these methods may be promising also for the problem approached in this work. Furthermore, we also use a new modelling framework, shifting from a CLP to a Vertex Cover Problem (VCP), which is more appropriate to the features of the underlying optimization problem. To assess and compare the performance of the new methods and the new modeling approach, we use a real scenario consisting of 45 vehicles moving around in the downtown area of the city of Malaga in Spain. Apart from this, the proposed optimization algorithms are compared against two baseline algorithms, the GA proposed in [20] and the state-of-the-art approach for information dissemination presented in [10].

The rest of this paper is structured as follows. Section 2 gives background about information dissemination in VANETs and VCPs. The new modelling approach is presented and described in Section 3. Then, Section 5 is devoted to detail the experimental framework used to test our proposal. After that, in Section 6 we analyze the results of the comparison of the proposed metaheuristics and the two baseline methods. Finally, Section 7 presents main conclusion of the work as well as the future research lines.

2 RELATED WORK

2.1 Information Dissemination in VANETs

Due to the interest of deploying cooperative services and applications for vehicles, information dissemination in VANETs has been extensively researched [5, 29]. The aim of a VANET is to be an infrastructure-less self-organizing traffic information system. Therefore, the vast majority of proposed methods for information dissemination are based on a decentralized architecture wherein the organization of the network is managed by vehicles creating dynamic clusters of vehicles, which is enabled by short-range communication technologies. The creation and management of the cluster imply the periodical communication of status information [13], which in high density scenarios can lead to the exchange of a high number of messages to organize the network, an in turn to the consumption of many resources for network management. Other approaches try to reduce this overhead deploying infrastructure nodes at preferential locations, but they usually lack of flexibility and the dissemination covered region is fixed [29]. Few works in the literature propose to use vehicles as mobile infrastructure. Camara et al. [8] present the virtual RSU (vRSU) concept where nodes receive and cache messages from other vRSU, or access points that are located in areas with no coverage from conventional RSUs.

Recently, heterogeneous architectures have been proposed to exploit both the wide range low latency communications of cellular technologies - Long-Term Evolution (LTE) communication (commercially known as 4G network) - and the low cost of IEEE 802.11p - similar to common WiFi networks. Some works use the heterogeneous architecture to improve the efficiency of clustering [10, 32]. In [32], Remy et al. use a heterogeneous centralized architecture to reduce the clustering overhead. An interesting proposal can be found in [10] where D'Orey et al. presented the approach Neighbor-Aware Virtual Infrastructure (NAVI), for information dissemination using vehicles as mobile Virtual Infrastructure (VI) that were selected from a central entity called GeoServer. NAVI presented many advantages as an appropriate uplink performance, reduced use of fixed infrastructure and better use of individual technologies. More recently, in [20], A.D. Masegosa et al. using this VANET architecture, they modeled the selection of vehicles used as VI by means of a CLP whose objective consists on maximizing the covering (vehicles that receive the message), while minimizing the number of vehicles used as VI. They also proposed a Genetic Algorithm to solve the optimization problem, and they show that this new methodology outperformed the VI selection used in [10].

As mentioned in the introduction, in this paper, we aim at extending the research done in [20]. Therefore, we will use the same VANET architecture based on NAVI that will be further described in Section 3.1. However, we will model the selection of the vehicles used as VI, as a VCP instead of a CLP. In the next section we define CLPs formally and review some literature about them.

2.2 Vertex Cover Problems

Given an undirected graph $G = (V, E)$, the Vertex Cover Problem (VCP) [7] consists on finding a subset $C \in V$ such that each edge in E has at least one end point in C, and the cardinality of C is limited by a constant k ($|C| < k$). Some applications of the VCP can be found in dynamic detection of potential data races in multi-threaded programs[22], network monitoring [4], or telecommunications [21].

In the next part of this section, we describe and formulate two of the most important generalizations of the VCP: the Minimum Vertex Cover and the Maximum Partial Vertex Cover.

The Minimum Vertex Cover Problem (MVCP). The MVCP is an extension of the VCP and consists on finding the minimum subset C that covers all the edges in E [16]. It is formulated as:

- V - set of vertices in the undirected graph G

- E - set of edges in the undirected graph G ($E \in V \times V$). An edge between vertices i and j is denoted as the duple $(i, j) \in E$
- C - the cover set, $C \subseteq V$
- N_i - $\{j \in V | (i, j) \in E\}$, the set of vertices adjacent to i.
- x_j - boolean variable set to 1 if a vertex $j \in C$, 0 otherwise.

$$\textbf{\textit{Minimize:}}\ Z = \sum_{j \in V} x_j \tag{1}$$

Subject to:

$$\sum_{j \in N_i} x_j \geqslant 1 \qquad \forall i \in V \tag{2}$$

$$x_j = \{0, 1\} \qquad \forall j \in V \tag{3}$$

The objective function given in Equation 1 minimizes the number of vertices in the cover set C. Equation 2 ensures that each vertex i is covered by at least one vertex in C, and finally, Equation 3 enforces the binary restriction on the decision variable.

The Weighted Maximum Partial Vertex Cover (WMVC). Unlike the MVC, in the WMVCP the cardinality of the cover set C is bounded at a value equal or lower than p [1]. The objective of the MVC consists on finding a subset C, such that $|C| <= p$, which maximizes the number of edges covered, that is, the number of edges in G with at least one end point in C. The mathematical formulation of the MVCP is as follows:

- V - set of vertices in the undirected graph G
- E - set of edges in the undirected graph G ($E \in V \times V$). An edge between vertices i and j is denoted as the duple $(i, j) \in E$
- w_i - weight associated to vertex i
- C - the cover set, $C \subseteq V$
- N_i - $\{j \in V | (i, j) \in E\}$, the set of vertices adjacent to i.
- x_j - boolean variable set to 1 if a vertex $j \in C$, 0 otherwise.
- y_i - represent the coverage of vertex i. Its value is 1 if vertex i is covered ($\exists_j \in V | x_j = 1 \wedge j \in N_i$), and 0 otherwise.
- p - the maximum allowed cardinality for the cover set C.

$$\textit{Maximize} \qquad Z = \sum_{i \in V} w_i y_i \tag{4}$$

Subject to:

$$\sum_{j \in N_i} x_j \geqslant y_i \qquad \forall i \in V \tag{5}$$

$$\sum_{j \in V} x_j \leq p \tag{6}$$

$$x_j = \{0, 1\} \quad \forall j \in V, \qquad y_i = \{0, 1\} \quad \forall i \in V \tag{7}$$

Equation 4 refers to the objective function that maximizes the weighted sum of the vertex covered. Equation 5 is analogous to Equation 2. Equation 6 limits the cardinality of C to p. Finally, Equation 7 restricts variables x_i and y_i to binary values.

3 APPROACH FOR INFORMATION DISSEMINATION IN VANETS

This section is devoted to describe the approach for information dissemination used in this paper. As mentioned above, this approach was first proposed in [20] that in turn was based on NAVI's network architecture [10], but with a procedure for selecting vehicles as VI based on the optimization of a CLP through a Genetic Algorithm.

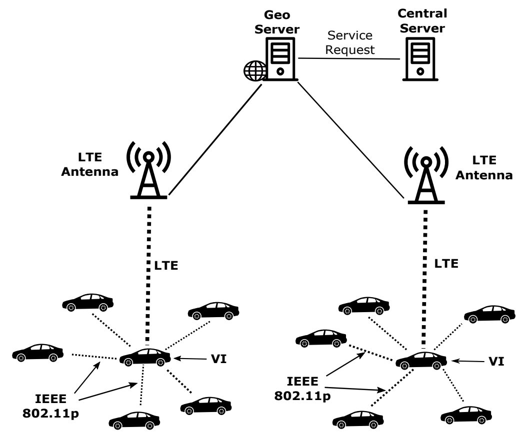

Figure 1: Scheme of the general architecture of NAVI

As we also pointed out in Section 1 and explained in Section 2, in this paper we reformulate the optimization model as a VCP instead of a CLP because it is more consistent with the real underlying optimization problem. The main reason behind this shift is the nature of the communications that do not depend only on a distant radius but in more complex factors. Given that VANETs can be seen as a graph where vertices correspond to vehicles and edges link vehicles that can communicate between each other, VCP fits better in this scenarios than CLP, although they are analogous problems.To facilitate the understanding of modelling process used, we first introduce NAVI's network architecture; then, we give the formal description of the VCP-based model for selecting the vehicles used as VI; and finally, the novel nature-inspired metaheuristics presented in this work to solve the optimization problem are presented.

3.1 NAVI's Network Architecture

For the sake of simplicity and due to limitation space, in this section we will only give a quick overview of the NAVI's network architecture. The interested reader may refer to [20] or [10] for further details. Having said this, NAVI's architecture was designed for an efficient collection and dissemination of information in VANETs through the selection of mobile infrastructure nodes (vehicles) in a scenario with multiple technologies [10]. Figure 1 shows a scheme of the general architecture of the system. The system comprises of a heterogeneous network architecture consisting on short-range communication networks, as IEEE 802.11p, and long-range communication networks, as LTE. Vehicles can be categorized into three main classes in terms of network equipment: i) short-range communication only, ii) long-range communication only and iii) short and long-range communication. The working of the system can be divided in three main stages: a) data collection, b) virtual infrastructure selection and c) data dissemination strategy execution.

In the *data collection phase*, the periodic broadcast single-hop CAMs are used to establish when two vehicles can communicate between them. Concretely, if a vehicle A receives a CAM message from another vehicle B, it means that they can transfer information among them. In this case, we establish that B is a neighbour of A. Each vehicle has it own Neighbourhood Table (NT) with an entry for each neighbour. The NTs are aggregated and transmitted periodically to the Geoserver by a subset of nodes in the Region of Interest (ROI). The *VI selection phase* updates the VI periodically from the information provided by the aggregated NTs, with a prefixed period length t. The selection of the VI is made centrally by Geoserver. The Geoserver launches this process each time it receives a dissemination request from the service provider. In the last phase, *data dissemination strategy execution*, the vehicles selected in the previous step perform the corresponding action: broadcast, relaying, and store-and-forward.

3.2 Virtual Infrastructure Selection Model based on Vertex Covering

In this section, we describe the reformulation of the model proposed for the VI selection process. Its objective consists on the efficient transfer of information from a central entity to the vehicles using the proper configuration of the VI. Concretely, it aims to balance the amount of data transferred through long-range communication (e.g. standard LTE) and short-range communication networks (e.g. standard IEEE 802.11p), in such a way that minimizes the use of long-range networks while maximizing the coverage area. As explained in [10], although data offloading to short-rage communication networks increases the overhead in short-range networks, it results in gains for the use of cellular networks that usually have a pay-per-use system.

The formulation used to model this optimization process is based on the VCP. The model can be divided into three main components: vehicles, scenario and network systems. Vehicles have access to network infrastructure resources and move on a given scenario S, which is partitioned in zones s such that $s \in S$. The properties of the scenario impact vehicle mobility and the communication reliability between vehicles, but we assume that during the period between the creation of the NTs and the dissemination of information, they remain unaltered. Taking into account that the length of this period is usually lower than 200ms, this assumption can be considered as realistic. The vehicles located in the scenario at the instant in which a dissemination request is received by the GeoServer it is represented by the set V. During the observation period, additional vehicles may join or leave the scenario depending on demand or routes, but as mentioned before, given the short length of the period, these changes are irrelevant. The objective of the underlying optimization process is to find the minimum subset $C \subseteq V$ that maximizes the number of zones covered in the scenario S. More formally, the model is defined as follows:

- S - set of zones to cover in the region. Each zone $s \in S$ has at least one vehicle located in it.
- V - set of vehicles that must be covered.
- V^{VI} - subset of vehicles from V ($V^{VI} \subseteq V$) that can be potentially used as VI, that is, those equipped with short and long range communication capabilities.
- CAM_{ij} - boolean variable whose value is 1 if vehicle $i \in V$ has received a CAM message from vehicle $j \in V$.
- NV_i - $\{j \in V^{VI} | CAM_{ij} = 1\}$, the set of neighbours of vehicle $i \in V$ (adjacent vertices).
- ZV_i - zone $s \in S$ in which vehicle $i \in V$ is located.
- V^s - $\{i \in V | ZV_i = s\}$, set of vehicles located in zone $s \in S$.
- N_s - $\{i \in V^{VI} | (i \in V_s) \vee (\exists j \in V_s \ s.t. \ j \in NV_j)\}$, the set of vehicles that can cover zone s (those vehicle from V^{VI} that can communicate with at least one vehicle in zone s).
- x_j - boolean variable set to 1 if vehicle $j \in V^{VI}$ is selected as VI, 0 otherwise.
- y_s - represent the coverage of zone $s \in S$. Its value is 1 if zone s is covered and 0 otherwise.
- p - maximum number of vehicles that can be used as VI.

$$Maximize \qquad Z = (1 - F_1) \cdot F_2 \tag{8}$$

where:

$$F_1 = \frac{1}{|V^{VI}|} \sum_{j \in V^{VI}} x_j \tag{9}$$

$$F_2 = \frac{1}{|S|} \sum_{s \in S} y_s + \frac{1}{|S||V|} (-1 + \sum_{s \in S} y_s |V^s|) \tag{10}$$

Subject to:

$$\sum_{j \in N_s} x_j \geqslant y_s \qquad \forall s \in S \tag{11}$$

$$\sum_{j \in V^{VI}} x_j \leq p \tag{12}$$

$$x_j = \{0,1\} \quad \forall j \in V, \qquad y_i = \{0,1\} \quad \forall i \in V \tag{13}$$

Equation 8 corresponds to the objective function which combines the two objectives, maximizing the zones covered while minimizing the number of vehicles used as VI. Equation 9 defines the proportion of vehicles considered as VI. In Equation 10, the first term of the expression corresponds to the ratio of zones covered, whereas the second one to the proportion of vehicles covered divided by $|S|$. Note that the values of this second term ranges in the interval $[0, \frac{1}{|S|}]$. In this way, a solution is considered better if it covers more zones or at the same number of zones, if it covers more vehicles. The constraint formulated in Equation 11 establishes that a zone s is covered only when at least one vehicle in the VI has, as neighbour, one of the vehicles located at s. Equation 12 ensures that the number of vehicles used as VI is at most p. Finally, Equation 13 restricts variables x_j and y_s to binary values.

To avoid dealing with feasible and unfeasible solutions, we used the penalization scheme show in Equation 14, where $p' = \sum_{j \in V^{VI}} x_j$ and c is a parameter weights the magnitude of the penalization.

$$Z(x) = \begin{cases} (1 - F_1(x)) \cdot F_2(x) & \text{if } p' \leq p \\ (1 - F_1(x)) \cdot F_2(x) - c(p' - p) & \text{otherwise} \end{cases} \tag{14}$$

4 PROPOSED BIO-INSPIRED SOLVERS

In this section, we describe the four nature-inspired metaheuristics that we have developed in order to solve the optimization problem detailed in 3.2, and therefore, to select the vehicles used as VI. Before the description of each deemed method, some important design issues are detailed in what follows, related to the codification strategy, the metric used for measuring the difference between two individuals, and the modeled movements operators.

Being one of the crucial aspects in the heuristic development, it is interesting to highlight that the *binary representation* has been adopted as encoding strategy. Thus, each potential solution is represented as a vector $\mathbf{x} = [x_1, x_2, \ldots, x_n]$ of n binary values $[0, 1]$, where n depicts the cardinality of the subset V^{VI}, that is, $n = |V^{VI}|$ and it refers to the number of vehicles that can be potentially used as VI. Additionally, x_j is analogous to the same symbol defined in 3.2, and represents the selection or not of the vehicle $j \in V^{VI}$ as VI or not (e.g. $x_j = 1$ if vehicle $j \in V^{VI}$ is selected as VI, 0 otherwise).

Regarding the distance metric employed for assessing the similarity between two solutions, the well-known Hamming Distance ($H(\cdot,\cdot)$) has been used. Several previous studies in the recent literature have utilized this same function for similar purposes [24], verifying its adequacy for solving combinatorial optimization problems, such as the one addressed in the present work. Specifically, $H(\cdot,\cdot)$ is computed as the number of non-corresponding elements between two solution. For instance, and assuming two individuals $\mathbf{x} = [1, 0, \mathbf{1}, 1, \mathbf{0}, 1, 1, 1, 0, \mathbf{1}]$ and $\mathbf{x}' = [1, 0, \mathbf{0}, 1, \mathbf{1}, 1, 1, 1, 0, \mathbf{0}]$, their Hamming Distance $D_H(\mathbf{x}, \mathbf{x}')$ would equal 3. It should be highlighted that this distance metric is the basis of the movement strategies inherent to each of the proposed techniques

Lastly, two movements operators have been considered for evolving the solutions along the search process. The first one is the commonly used Uniform Crossover ($UC_p(x, y)$ [35]), in which two different individuals share their genes probabilistically for producing a new hybrid solution. For this study, a probability p=50% has been used for each gene, which means that every gene c_i has the same probability of be inherited by x or y. The second operator used is the swapping $SW(x, c)$ operator, in which c genes are randomly chosen from solution x for reversing their value, from 1 to 0, or from 0 to 1. In this specific research, c=2 value has been fixed.

Below we defined the four nature-inspired metaheuristics proposed for the optimization problem described in Section 3.2.

Bat Algorithm (BA). The BA, introduced by Xin-She Yang in [38], is a bio-inspired method based on the echolocation behavior of microbats, which can find their prey and discriminate different kinds of insects even in complete darkness. Despite it was firstly proposed for solving continuous optimization problems, it has been adapted several times before [27, 33, 41] for solving discrete problems. Following the same philosophy, an adaptation has been made in the present research. In our proposal, each bat represents a solution to the problem, and concepts of loudness A_i and pulse emissions r_i have been considered identically to the canonical BA. On the other hand, frequency f_i parameter has not been deemed for the sake of simplification. Additionally, velocity v_i value has been adapted taking the $H(\cdot,\cdot)$ as reference similarity function. This way, the velocity is calculated following the formula $v_i^t = rand[1, D_H(\mathbf{x_i}, \mathbf{x}_*)]$. In other words, the v_i of a bat i at time step t is a random number following a discrete uniform distribution between 1 and the different between i and the leading bat. Regarding the movement criterion, each bat x_i moves towards the best individual x_* at each generation $t \in \{1, \ldots, T\}$ following the formula:

$$\mathbf{x_i}(t+1) = \Psi\Big(\mathbf{x_i}(t), \min\big\{V, v_i^t\big\}\Big), \tag{15}$$

where $\Psi(\mathbf{x}, Z) \in \{UC_{0.5}(x_i, x_*), SW(x, c)\}$, each one parametrized by the amount of times Z this operator is executed to $\mathbf{x}$. Then, the best movement carried out on $\mathbf{x}$ is chosen as output. To properly select the movement operator, and with the aim of enhancing the exploration capacity of the method, the *inclination* mechanism recently proposed in works, such as [23, 25, 27], is also used considering $UC_p(x, y)$ as wide movement and $SW(x, c)$ as short movement.

Firefly Algorithm (FA). FA is another method introduced by Xin-She Yang in 2010 [37] that tries to emulate flashing behaviour of fireflies, which act as a signal system to attract other colleagues. Some modification have been performed also in this case for adapting the FA to the discrete problem tackled in this research. As in the previous case, each firefly of the swarm represents a possible solution to the problem. Besides that, the concept of light absorption γ is considered, being essential for adjusting fireflies' movement. The distance between two individuals is calculated employing the Hamming Distance $D_H(\cdot,\cdot)$ functions. Additionally, the movement criterion adopted by a individual x is also defined by the Expression (15). This way, each time x is about to move towards a counterpart x', it analyzes the related $D_H(\mathbf{x}, \mathbf{x}')$. Following also the *inclination* mechanism above mentioned, if $D_H(\mathbf{x}, \mathbf{x}')>V/2r$, a wide movements is carried out through $UC_{0.5}(x, x')$. Otherwise, a short move is conducted using $SW(x, 2)$.

Cuckoo Search (CS). The CS was introduced by Yang and Deb in 2009 [39] for solving continuous optimization problems. One of its main advantages is its easy parameterization, which along with its efficiency has lead the CS to have a great success recently [3, 17, 19]. In this work, the well-known adaptation proposed in [28] for the Traveling Salesman Problem has been embraced as base, using similar mechanisms and parameters. Specifically, for the cuckoos movement, the same $UC_p(x, y)$ and $SW(x, c)$ operators have been employed. Additionally, the movement of each cuckoo is conducted using the same logic depicted in Expression (15), considering $D_H(\cdot,\cdot)$ as similarity function, and best individual as reference.

Particle Swarm Optimization (PSO). PSO has been adapted to discrete problems multiple times before [6, 40]. Taking as base some of these interesting works, each particle in our developed PSO also represents a possible solution for the problem at hand. Additionally, velocity parameter v_i has been considered in the same way as done in BA case. Furthermore, both inclination mechanism and $UC_p(x, y)$, $SW(x, c)$ movement strategies have also been deemed for the PSO, as well as the movement criterion shown in Expression (15). Finally, $D_H(\cdot,\cdot)$ has been use as similarity measurement.

5 EXPERIMENTAL FRAMEWORK

This part of the chapter is devoted to describe the experimental framework used in this study. Concretely, in Section 5.1 we give the details of the realistic simulation scenario employed in the experimentation. Then, in Section 5.2 we describe the two methods taken as baseline to compare it versus the four nature-inspired metaheuristics proposed in this paper. After that, we list the performance measures used in the experimentation and finally, Section 5.4 points out the details of the implementations carried out.

5.1 VANET simulation scenario

The simulation scenario used is the same employed in [10]. The VANET simulation was done using GPS traces publicly available

Table 1: Main Simulation Parameters

Type	Parameter	Value
Neighbor Information	CAM Frequency	1 Hz
	Neighbor Table Timeout	5 s
	Server update Frequency	1 Hz
Dissemination Request	Frequency	1 Hz
	Dissemination area	0.44 km^2
Scenario	Type	Urban
	Number of Vehicles	45
	Simulation Duration	180 s
	Vehicle Speed	10-50 km/h
	Vehicle Density	113 veh/km^2
	Maximum VI size	[2,4,6,8,10] nodes
802.11p Network	Bit Rate	6 Mbps
	Bandwidth	10 Mhz
	Frequency band	5.9 GHz
	Maximum Tx Power	[16, 21, 23] dBm
LTE Network	eNodeB Tx Power	30 dBm
	UE Tx Power	10 dBm
	Propagation Model	Friis Tx Eq

in NS-2 format[1], which were taken as input in the network simulation platform. The tool SUMO (Simulation of Urban Mobility) was then used to generate the mobility traces using realistic input data, including road network, vehicles routes or traffic lights among others. The location of the simulation scenario is a rectangular area of $600m \times 700m$ in the downtown of the city of Malaga, Spain. The simulation period is 180s and the maximum vehicle speed is $50km\backslash h$. Table 1 contains the details of the simulation parameters. For further details, the interested reader is referred to [36]. Finally, we have considered three different maximum transmission powers for the short-range network (IEEE 802.11p), 16dBm, 21dBm and 23dBm, and five different maximum VI sizes (2, 4, 6, 8 and 10 nodes). The transmission power controls the communication range of the vehicles and therefore the neighbourhood awareness levels: a higher transmission power implies a higher communication range that, in turn, entails a higher number of neighbours per vehicle, but also a higher consumption of energy. In the VCP models, the transmission power can be seen as a factor that modifies the number of vertices adjacent to each vertex. Regarding the maximum VI size, represented by the parameter p in the model described in Section 3.2, it limits the resource consumption in terms of LTE connections.

5.2 Baseline algorithms

The methods used to compare the performance of the four nature-inspired methods proposed in this paper are, on the one hand, the algorithm proposed in the original paper of NAVI architecture [10] and on the other hand, the Genetic Algorithm proposed by Masegosa et. al in [20]. We will refer to this methods as to which we will refer as NAVI_Alg and NAVI_GA, respectively.

Regarding *NAVI_Alg*, the authors applied a Min-Max formulation to model the optimization problem and an ad-hoc greedy algorithm as solver. The objective consisted on maximizing the number of zones covered while minimizing the number of vehicles used as VI, using at most p vehicles. The general idea of the greedy algorithm is that, at each step, it selects as VI that vehicle that covers the maximum number of zones that has not been covered yet. The method stops when the VI size is equal to p or when all zones are covered. The interested reader is referred to [10] for further details. As for *NAVI_GA*, it is a generational GA with binary codification and elitism. Regarding the genetic operators, the method used tournament selection with size q, uniform crossover and uniform mutation. Further details can be found in [20].

5.3 Performance measures

The performance metrics used to compare the algorithms considered are given below:

- *Coverered Area (Maximize):* percentage of zones from covered, which corresponds with those regions that has at least one vehicle that would receive the dissemination message.
- *VI size (Minimize):* number of vehicles selected as VI. In this way, we measure the resource consumption of the solution.

5.4 Implementation details

To finish with the description of the experimental framework, we give here the details of the implementation done and the parameter settings used for all the four proposed nature-inspired metaheuristics. First of all, the population size has been fixed in 100 for each approach. For the *FA*, γ=0.95. Furthermore, for *BA* α=β=0.98, A_i^0=1.0 and r_i^0=0.1. Besides that, for *CS* p_a = 0.2. Finally, *PSO* have been configured as described in Section 4. For the development and parameterization of these methods, the guidelines given in [24, 26–28] have been followed. Regarding GA_NAVI, the population size was also 100, the tournament size was set to 5, the crossover rate to 0.5 and the mutation rate to 0.015. The stopping criteria for FA, BA, CS, PSO and GA_NAVI was 100 iterations/generations. In the objective function, the penalization coefficient c was empirically set to 0.2.

As for the simulation scenario, the region of interest was divided into 100 rectangular zones, all of them with the same width and height. The number of vehicles to be covered was 45, and we assume that all of them are equipped with both short and long range communication capabilities, that is, all of them can be used as VI. The positions of the vehicles were sampled every second. Given that the simulation time for the data was 180s, in this way, we have a total of 180 instances of the problem. Taking into account that we considered three different transmission powers and five different maximum VI size, this experimentation counts with a total of $180 \times 3 \times 5 = 2700$ instance configurations. For each of these instance configurations, our proposal was run 10 times and the mean covered area and VI size were registered.

The implementation was done in Java 8, and the experiments were run on a computer with Ubuntu 16.08, 42GB RAM and 2 CPUs Intel Xeon Silver 4114 2,2GHz 13.75MB Cache 10 Cores. The results of the NAVI method were provided by its authors.

6 RESULT ANALYSIS

The objective of the experimentation done in this paper is two-fold:

[1]http://neo.lcc.uma.es/staff/jamal/vanet/?q=node/11

- *Assess the performance of the proposed nature-inspired metaheuristics in a real scenario.* Concretely, our aim is to test on a real scenario the four methods presented in this paper to get insights about their performance and check under what circumstances one method is preferred over the others.
- *Compare the performance of the proposed nature-inspired metaheuristics versus two baseline algorithms.* To assess the competitiveness of our proposal, we will compare it against the two base line algorithms described in Section 5.2.

To make this analysis, we show the results obtained by the four considered nature-inspired metaheuristics (BA,FA,CS and PSO) and the two baseline algorithms (NAVI_GA and NAVI_Alg) in Figures 2 and 3. These figures show boxplots with information about the distribution of the results obtained over the 180 instances by each method. The Y axis represents the covered area and VI size, respectively, the X axis includes the five Maximum VI Sizes (MVISs) and the three panels depict the Maximum Transmission Powers (MTxPs) considered. Each series correspond to a different method: orange, green, turquoise and purple for BA, CS, FA and PSO, respectively; whereas red and blue for NAVI_Alg and NAVI_GA, in that order. In the boxes, the central horizontal line indicates the median, the hinges of the boxes the first and third quartiles, and the whiskers the value $1.5 \cdot IQR$, where IQR is the interquartile range. The dots refer to outlier values. The graphics were generated using R programming language and the ggplot2 package[2].

Beginning with the covered area, we can observe in Figure 2 that, as expected, for a specific MTxP, the higher the MVIS the higher the percentage of zones covered, and viceversa, for a specific MVIS, the higher the MTxP the higher the covering. A similar rule can be established for the complexity of the problem which is negatively correlated with both MTxP and MVIS. The lower the MTxP or the MVIS, the worse the performance of the methods. Comparing the four proposed methods among them, we can see that the difference in performances varies with the MVIS, but not with the MTxP. When MVIS is equal to two, for the three MTxP values considered, CS is the best performing method, whereas BA, FA and PSO obtain pretty similar results. However, when MVIS is equal to four and MTxP to 16 dBm, BA works slightly worse than the other three ones that show very similar performance, again. In the rest of cases, no differences in percentage of covered zones can be appreciated. If the comparison is made w.r.t the baseline algorithms, the four proposed methods outperform NAVI_Alg and obtain better or equal results than NAVI_GA, which proves the good performance of the nature-inspired metaheuristics proposed in this paper.

Regarding the VI size, displayed in Figure 3, it is interesting to see that the relative performance of BA, CS, FA and PSO varies significantly w.r.t when the covered area was considered. In this case, we can observe that CS is actually the worse method, event for "easy" instances, as those with MVIS set to two. The second worse performing method was the PSO algorithm, specially if we look at the results with ten as MVIS, but very close to the best performing methods, which are BA and FA. The comparison w.r.t the base line algorithms is quite similar to what we see above when BA, FA and PSO are considered, that is, they outperfrom NAVI_Alg and they obtain better result than NAVI_GA in almost all cases. The

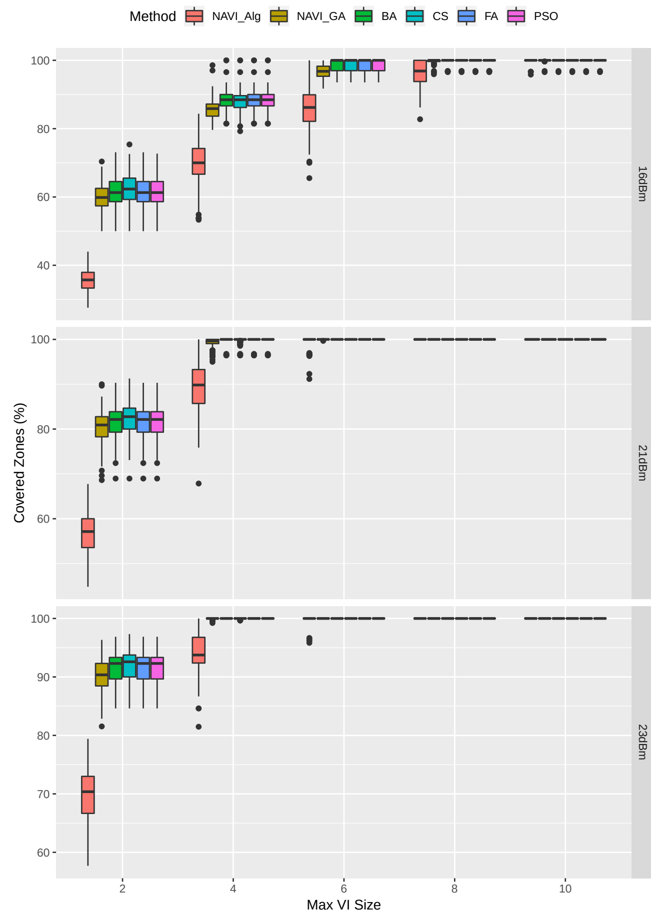

Figure 2: Boxplot for the covered area results. The Y axis represents the covered area, the X axis the MVIS, the panels the MTxP and the series the methods compared.

exception here is the CS algorithm, which perform even worse than NAVI when the a MVIS of two, and similar or worse than NAVI_GA for 21 and 23 dBm as MTxP. The only cases when we can observe a better performance of the CS w.r.t NAVI_GA are in those scenarios with 16 dBm as MTxP and 8 and 10 as MVIS values, respectively.

To assess whether the differences in performance observed among the different analyzed methods are significant or not, we made use of non-parametric statistical tests. Two statistical tests have been applied, following the guidelines proposed in [9], for the two measures considered: the covered area and the VI size. First, the Friedman's test for multiple comparisons has been applied to check whether there are significant differences among the studied methods. The samples correspond with the mean performance of the algorithm for a pair MTxP-MVIS, that is, for each method there are 15 samples. Given that the p-value returned by this test for both measures was 0.0, the null hypothesis can be rejected in all cases. The mean ranking returned by this test for the covered area and the VI size is displayed in Table 2. According to this ranking, BA and FA are the best performing methods for the two considered measures, respectively. Holm post-hoc test has also been applied

[2] http://ggplot2.org/

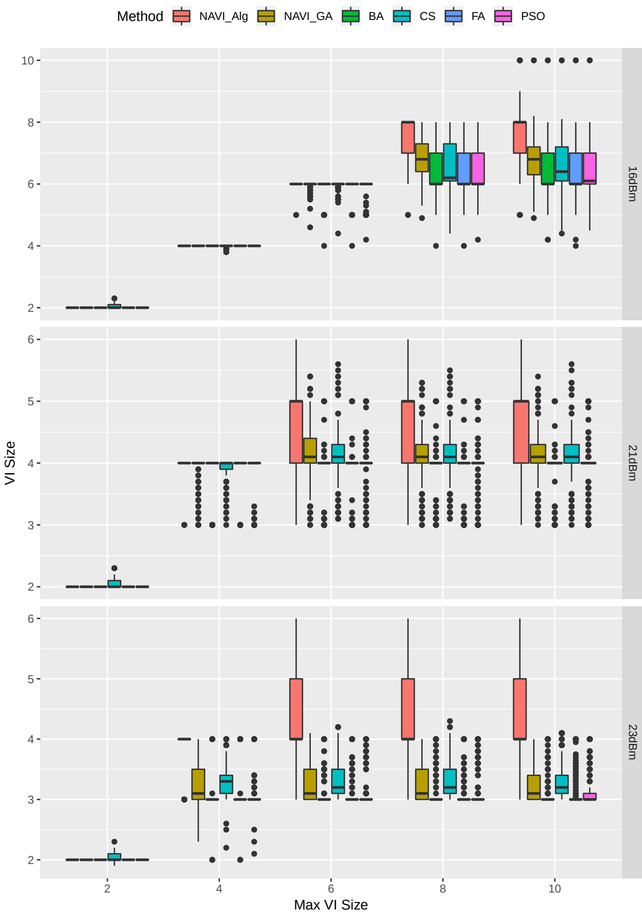

Figure 3: Boxplot for the VI size results. The Y axis represents the covered area, the X axis the MVIS, the panels the MTxP and the series the methods compared.

using the best rank method in each case (highlighted in bold). In order to highlight significant differences, those p-values lower than 0.05 are are marked with an asterisk (*). For the covered zones, BA is significantly better than the two baseline algorithms. Regarding the VI size, in addition to the two baseline algorithms, FA improving significantly CS. To sum up, this analysis confirm the better performance of the proposed nature-inspired metaheuristics w.r.t. the baseline methods, particularly BA and FA.

7 CONCLUSIONS

In this work, we have aimed at improving information dissemination in VANETs deepening into the approach presented by Masegosa et al. in [20], which is based on the optimization of a Covering Location Problem by means of a Genetic Algorithm. Concretely, we have presented and compare four new nature-inspired metaheuristics to address this optimization problem: BA, CS, FA y PSO. Furthermore, we have shift the model to Vertex Cover because its abstraction is more close to the real optimization problem addressed.

The methods presented here were tested on a real scenario consisting of 45 vehicles moving on a rectangular area of $600m \times 700m$

Table 2: Ranking provided by Friedman's non-parametric test for both Covered Area and VI Size

Method	Ranking Covered Area	Ranking VI Size
NAVI_Alg	5.33*	5.20*
NAVI_GA	4.47*	4.13*
BA	**2.53**	2.17
CS	2.83	4.47*
FA	2.67	**2.03**
PSO	3.17	3.00

in the downtown of the city of Malaga, Spain. Beside, we considered three different maximum transmission powers for the 802.11p network and five different maximum VI sizes. The performance of the solutions was measured in terms of covered area and number of vehicles used as VI (VI size). The objective of the experimentation done over this real scenario was two-fold: on the one hand, to analyze and compare the performance of the four proposed metaheuristics, and on the other hand, to assess the competitiveness of the four methods comparing it with the NAVI's original algorithm for selecting the VI and the Genetic Algorithm proposed in [20].

The analysis of the obtained results showed that in terms of covered area, BA was the best of the four algorithms, followed by FA, CS and PSO, in that order although with no significant diferences among them. However, when the VI size was considered as performance measure, FA was the best algorithm but with no significant differences w.r.t BA and PSO. CS was significantly worse than FA. Regarding the comparison w.r.t the baseline algorithms, BA, FA and PSO improve the performance of NAVI_Alg and NAVI_GA both in terms of covered area and VI size being significantly improved by BA and FA in terms of covered area VI size, respectively.

In conclusion, we have shown that nature-inspired metaheuristics, specially BA, FA and PSO can be considered as powerful solvers that can contribute to improve the information dissemination approach for VANETs discussed in this paper.

Several research lines will be tackled in the near future. In the short-term, additional nature-inspired and evolutionary methods [2, 14, 31] are planned to be included in the benchmark to assess whether they are able to obtain better results. In addition, we intend to solve the same problem using bigger scenarios, with a wider area and composed by a higher amount of vehicles. Finally, given the characteristics of the problem addressed here, we would also plan to model it as a dynamic optimization problem and as robust over time optimization problem.

ACKNOWLEDGMENTS

E. Osaba, I. Laña and J. Del Ser would like to thank the Basque Government for its support through the EMAITEK program. E. Osaba, I. Laña and J. Del Ser acknowledge the financial support from the project SCOTT: Secure Connected Trustable Things (ECSEL Joint Undertaking, ref. 737422).

J. S. Angarita-Zapata thanks the funding support received from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 636220 and the Marie Sklodoska-Curie grant agreement No. 665959.

A. D. Masegosa thanks the LOGISTAR project, which received funding from European Union's Horizon 2020 research and innovation programme under grant agreement No. 769142.

REFERENCES

[1] Nicola Apollonio and Bruno Simeone. 2014. The maximum vertex coverage problem on bipartite graphs. *Discrete Applied Mathematics* 165 (2014), 37 – 48. https://doi.org/10.1016/j.dam.2013.05.015 10th Cologne/Twente Workshop on Graphs and Combinatorial Optimization (CTW 2011).

[2] Esmaeil Atashpaz-Gargari and Caro Lucas. 2007. Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. In *2007 IEEE congress on evolutionary computation*. IEEE, 4661–4667.

[3] Kirils Bibiks, Yim Fun Hu, Jian-Ping Li, Prashant Pillai, and Aleister Smith. 2018. Improved discrete cuckoo search for the resource-constrained project scheduling problem. *Applied Soft Computing* (2018).

[4] E. Bonfoh, S. Medjiah, and C. Chassot. 2018. A Parsimonious Monitoring Approach for Link Bandwidth Estimation within SDN-based Networks. In *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*. 512–516. https://doi.org/10.1109/NETSOFT.2018.8459972

[5] Moumena Chaqfeh, Abderrahmane Lakas, and Imad Jawhar. 2014. A survey on data dissemination in vehicular ad hoc networks. *Vehicular Communications* 1, 4 (2014), 214–225.

[6] Ai-ling Chen, Gen-ke Yang, and Zhi-ming Wu. 2006. Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem. *Journal of Zhejiang University-Science A* 7, 4 (2006), 607–614.

[7] Jianer Chen, Iyad A Kanj, and Weijia Jia. 2001. Vertex cover: further observations and further improvements. *Journal of Algorithms* 41, 2 (2001), 280–301.

[8] Daniel D. Camara, Christian Bonnet, Navid Nikaein, and Michelle Wetterwald. 2011. Multicast and virtual road side units for multi technology alert messages dissemination. In *2011 IEEE 8th International Conference on Mobile Adhoc and Sensor Systems (MASS)*. 947–952.

[9] Joaquín Derrac, Salvador García, Daniel Molina, and Francisco Herrera. 2011. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation* 1, 1 (mar 2011), 3–18. https://doi.org/10.1016/J.SWEVO.2011.02.002

[10] Pedro M. D'Orey, Nitin Maslekar, Idoia de la Iglesia, and Nikola K. Zahariev. 2015. NAVI: Neighbor-Aware Virtual Infrastructure for Information Collection and Dissemination in Vehicular Networks. In *2015 IEEE Vehicular Technology Conference (VTC Spring)*. 1–6. https://doi.org/10.1109/VTCSpring.2015.7145945

[11] Marshall L. Fisher and Pradeep Kedia. 1990. Optimal Solution of Set Covering/Partitioning Problems Using Dual Heuristics. *Management Science* 36, 6 (1990), 674–688. https://doi.org/10.1287/mnsc.36.6.674

[12] David E Goldberg. 1989. Genetic algorithms in search. *Optimization, and MachineLearning* (1989).

[13] Unai Hernandez-Jayo, Aboobeker Sidhik Koyamparambil Mammu, and Idoia De-la Iglesia. 2014. *Contemporary Issues in Wireless Communications*. InTech, Chapter Reliable Communication in Cooperative Ad hoc Networks. https://doi.org/10.5772/58482

[14] Dervis Karaboga and Bahriye Basturk. 2008. On the performance of artificial bee colony (ABC) algorithm. *Applied soft computing* 8, 1 (2008), 687–697.

[15] James Kennedy. 2010. Particle swarm optimization. *Encyclopedia of machine learning* (2010), 760–766.

[16] Sami Khuri and Thomas Bäck. 1994. An evolutionary heuristic for the minimum vertex cover problem. In *Genetic Algorithms within the Framework of Evolutionary Computation–Proc. of the KI-94 Workshop*. Saarbrücken, Germany, 86–90.

[17] Zixiang Li, Nilanjan Dey, Amira S Ashour, and Qiuhua Tang. 2018. Discrete cuckoo search algorithms for two-sided robotic assembly line balancing problem. *Neural Computing and Applications* 30, 9 (2018), 2685–2696.

[18] Wenshuang Liang, Zhuorong Li, Hongyang Zhang, Shenling Wang, and Rongfang Bie. 2015. Vehicular ad hoc networks: architectures, research issues, methodologies, challenges, and trends. *International Journal of Distributed Sensor Networks* 11, 8 (2015), 745303.

[19] Arindam Majumder and Dipak Laha. 2016. A new cuckoo search algorithm for 2-machine robotic cell scheduling problem with sequence-dependent setup times. *Swarm and Evolutionary Computation* 28 (2016), 131–143.

[20] Antonio D. Masegosa, Idoia de la Iglesia, Unai Hernandez-Jayo, Luis Enrique Diez, Alfonso Bahillo, and Enrique Onieva. 2018. *A New Approach for Information Dissemination in VANETs Based on Covering Location and Metaheuristics*. Springer International Publishing, Cham, 179–202. https://doi.org/10.1007/978-3-319-62359-7_9

[21] Parisutham Nirmala, Ramasubramony Sulochana Lekshmi, and Rethnasamy Nadarajan. 2016. Vertex cover-based binary tree algorithm to detect all maximum common induced subgraphs in large communication networks. *Knowledge and Information Systems* 48, 1 (01 Jul 2016), 229–252. https://doi.org/10.1007/s10115-015-0874-z

[22] Robert O'Callahan and Jong-Deok Choi. 2003. Hybrid Dynamic Data Race Detection. In *Proceedings of the Ninth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP '03)*. ACM, New York, NY, USA, 167–178. https://doi.org/10.1145/781498.781528

[23] Eneko Osaba, Javier Del Ser, David Camacho, Akemi Galvez, Andres Iglesias, and Iztok Fister. 2018. Community Detection in Weighted Directed Networks Using Nature-Inspired Heuristics. In *International Conference on Intelligent Data Engineering and Automated Learning*. Springer, 325–335.

[24] Eneko Osaba, Javier Del Ser, Ali Sadollah, Miren Nekane Bilbao, and David Camacho. 2018. A discrete water cycle algorithm for solving the symmetric and asymmetric traveling salesman problem. *Applied Soft Computing* 71 (2018), 277–290.

[25] Eneko Osaba, Xin-She Yang, Fernando Diaz, Pedro Lopez-Garcia, and Roberto Carballedo. 2016. An improved discrete bat algorithm for symmetric and asymmetric traveling salesman problems. *Engineering Applications of Artificial Intelligence* 48 (2016), 59–71.

[26] Eneko Osaba, Xin-She Yang, Fernando Diaz, Enrique Onieva, Antonio D Masegosa, and Asier Perallos. 2017. A discrete firefly algorithm to solve a rich vehicle routing problem modelling a newspaper distribution system with recycling policy. *Soft Computing* 21, 18 (2017), 5295–5308.

[27] Eneko Osaba, Xin-She Yang, Iztok Fister Jr, Javier Del Ser, Pedro Lopez-Garcia, and Alejo J Vazquez-Pardavila. 2019. A Discrete and Improved Bat Algorithm for solving a medical goods distribution problem with pharmacological waste collection. *Swarm and Evolutionary Computation* 44 (2019), 273–286.

[28] Aziz Ouaarab, Belaïd Ahiod, and Xin-She Yang. 2014. Discrete cuckoo search algorithm for the travelling salesman problem. *Neural Computing and Applications* 24, 7-8 (2014), 1659–1669.

[29] Sooksan Panichpapiboon and Wasan Pattara-Atikom. 2012. A review of information dissemination protocols for vehicular ad hoc networks. *IEEE Communications Surveys & Tutorials* 14, 3 (2012), 784–798.

[30] P. Papadimitratos, A. La Fortelle, K. Evenssen, R. Brignolo, and S. Cosenza. 2009. Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation. *IEEE Communications Magazine* 47, 11 (2009), 84–95. https://doi.org/10.1109/MCOM.2009.5307471

[31] Radu-Emil Precup, Emil-Ioan Voisan, Emil M Petriu, Mircea-Bogdan Radac, and Lucian-Ovidiu Fedorovici. 2016. Gravitational search algorithm-based evolving fuzzy models of a nonlinear process. In *Informatics in Control, Automation and Robotics 12th International Conference, ICINCO 2015 Colmar, France, July 21-23, 2015 Revised Selected Papers*. Springer, 51–62.

[32] Guillaume Rémy, Sidi-Mohammed Senouci, François Jan, and Yvon Gourhant. 2011. LTE4V2X: LTE for a centralized vanet organization. In *2011 IEEE Global Telecommunications Conference (GLOBECOM)*. 1–6.

[33] Yassine Saji and Mohammed Essaid Riffi. 2016. A novel discrete bat algorithm for solving the travelling salesman problem. *Neural Computing and Applications* 27, 7 (2016), 1853–1866.

[34] Cristiano M Silva, Barbara M Masini, Gianluigi Ferrari, and Ilaria Thibault. 2017. A survey on infrastructure-based vehicular networks. *Mobile Information Systems* 2017 (2017).

[35] Gilbert Syswerda. 1989. Uniform crossover in genetic algorithms. In *Proceedings of the third international conference on Genetic algorithms*. Morgan Kaufmann Publishers, 2–9.

[36] J. Toutouh, J. Garcia-Nieto, and E. Alba. 2012. Intelligent OLSR Routing Protocol Optimization for VANETs. *IEEE Transactions on Vehicular Technology* 61, 4 (2012), 1884–1894.

[37] Xin-She Yang. 2010. Firefly algorithm, stochastic test functions and design optimisation. *International Journal of Bio-Inspired Computation* 2, 2 (2010), 78–84.

[38] Xin-She Yang. 2010. A new metaheuristic bat-inspired algorithm. In *Nature inspired cooperative strategies for optimization (NICSO 2010)*. Springer, 65–74.

[39] Xin-She Yang and Suash Deb. 2009. Cuckoo search via Lévy flights. In *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*. IEEE, 210–214.

[40] Yiwen Zhong, Juan Lin, Lijin Wang, and Hui Zhang. 2018. Discrete comprehensive learning particle swarm optimization algorithm with Metropolis acceptance criterion for traveling salesman problem. *Swarm and Evolutionary Computation* (2018).

[41] Xu Zhou, Xiaohui Zhao, and Yanheng Liu. 2018. A multiobjective discrete bat algorithm for community detection in dynamic networks. *Applied Intelligence* 48, 9 (2018), 3081–3093.

View publication stats