VPGA: an SDN-based Location Privacy Zones Placement Scheme for Vehicular Networks

Abdelwahab Boualouache, Ridha Soua, and Thomas Engel
SnT, University of Luxembourg, Luxembourg
Email: {abdelwahab.boualouache, ridha.soua, thomas.engel}@uni.lu

***Abstract*—Making personal data anonymous is crucial to ensure the adoption of connected vehicles. One of the privacy-sensitive information is location, which once revealed can be used by adversaries to track drivers during their journey. Vehicular Location Privacy Zones (VLPZs) is a promising approach to ensure unlinkability. These logical zones can be easily deployed over roadside infrastructures (RIs) such as gas station or electric charging stations. However, the placement optimization problem of VLPZs is NP-hard and thus an efficient allocation of VLPZs to these RIs is needed to avoid their overload and the degradation of the QoS provided within theses RIs. This work considers the optimal placement of the VLPZs and proposes a genetic-based algorithm in a software defined vehicular network to ensure minimized trajectory cost of involved vehicles and hence less consumption of their pseudonyms. The analytical evaluation shows that the proposed approach is cost-efficient and ensures a shorter response time.**

***Index Terms*—Vehicular Networks; Security; Location privacy Zones; Software Defined Networks, Genetic Algorithm.**

I. Introduction

Vehicular networks have obtained considerable interest from both academia and industry given their positive impact on traffic efficiency road safety. However, the successful deployment of vehicular networks strongly depends on providing secure vehicular communications [1]. Location privacy is one of the main security requirements since it threatens the privacy of drivers and passengers. For this reason, location privacy must early be taken into account in the deployment of connected vehicles. Pseudonym-changing approach is the solution adopted by the current security standards to solve this issue [2] [3]. This approach proposes that vehicles frequently change their temporal identifiers (pseudonyms) to provide the unlinkablity between their identifiers and their currently positions, which are included in the safety-related messages broadcast in clear text. However, several studies have demonstrated that pseudonyms could be linked [4]. This limitation is significant and span a range of technical issues for both acceptance and deployment of connected vehicles. Hence, several strategies were proposed to ensure the unlinkablity between the pseudonyms [5]. However, none of the proposed pseudonym-changing strategies have been adopted by the standardization bodies until now.

Recently, the Location Privacy Zone (VLPZ) based strategy has been considered as one of the promising strategies that can ensure strong protection against pseudonym-linking attacks and an effective trade-off between location privacy and road safety [6][7]. Indeed, this strategy is based on virtual zones (VLPZs) that can easily be deployed in Roadside Infrastructures (RIs) such as gas and electrical charging stations. The location privacy protection offered by VLPZ can thus be considered as a secondary service provided by these RIs. Given that the deployment of VLPZs may generate additional costs to these RIs, VLPZs should intelligently be allocated to reduce the costs of location privacy service. On the other hand, the trajectory cost of vehicles to reach the VLPZ should also be considered. VLPZs should be placed at as near as possible to vehicles to optimize the service time and prevent losing more pseudonyms during their trajectory to the deployed VLPZs.

Nowadays, Software Defined Networking (SDN) paradigm is being considered in the future vehicular networks and promises to bring programmability and flexibility which are needed in this kind of dynamic networks. SDN provides a logically-centralized architecture and decouples the control plan from the data plane to efficiently manage the network. In this vein, Huang et al. [8, 9] proposed a new three-plane SDN architecture to provide efficient pseudonym resources management. The SDN control plane is responsible for deciding the rules of how the pseudonyms are distributed. Our solution is also built on a software-defined networking architecture to hold the mobility change and exploit the centralized intelligence of the SDN-Controller to efficiently select the VLPZs placement.

Different from [10], we mainly focus on studying how to choose the locations of the VLPZ in the vehicular network in order to minimize the use of pseudonyms during the move to the allocated VLPZ. To this end, we propose a genetic algorithm (GA) based VLPZs placement that can scale and provide good solutions in a fast way. In addition, given the inherent characteristics of vehicular networks such as high mobility and high density variation, we incorporate our GA based VLPZs placement in an SDN-enabled architecture to ensure its flexibility and reprogrammability. The main contributions of our work are as follows:

- To the best of our knowledge, this is the first study to consider the VLPZ placement problem in an SDN-enabled vehicular network from the perspective of trajectory cost and hence the number of used pseudonyms while vehicles are heading to their assigned VLPZ.
- To find the optimal solution, we formulate the cost-efficient VLPZ placement problem as an optimization

problem with the objective function that minimizes the trajectory cost for all involved vehicles.
- Considering the high complexity of the problem in large as vehicular networks, a VLPZ Placement Genetic Algorithm (VPGA) is introduced to find an effective sub-optimal solution.
- We conduct numerical analysis to compare our genetic algorithm based VLPZ placement strategy with our previous work PRIVANET [10].

The remainder of this paper is organized as follows. The background and some related work are reviewed in Section 2. The system model and the problem formalization are presented in Section 3. Section 4 describes the proposed VLPZ Placement Genetic Algorithm (VPGA). The Numerical results are presented in Section 5. Finally, the conclusion is given in Section 6.

II. Background and related work

A. VLPZ model

Vehicular Location Privacy Zone (VLPZ) is a logical zone that aims at protecting the location privacy of vehicular users [6]. The internal design of VLPZ is seemingly similar to RIs such as gas stations and vehicle charging stations. Indeed, a basic VLPZ consists of one entry point called ***the router***, one exit point called ***the aggregator*** and a limited number of lanes l where $l > 1$. For this reason, VLPZs can easily be placed on RIs. In addition, VLPZs can be created as independent RIs in the future vehicular networks given the urgent need of protecting the location privacy of road users. Figure 1 illustrates a two-way street where two VLPZs are installed: (i) $VLPZ_1$: for vehicles coming from West to East, and (2) $VLPZ_2$: for vehicles coming from East to West.

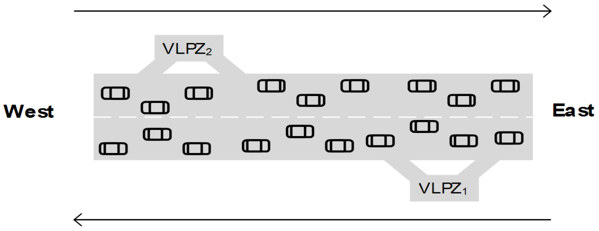


Fig. 1: Multiple VLPZs model.

Inside the VLPZ, vehicles can change their pseudonyms in a secure way as follows: vehicles arrive to a VLPZ, one after another, on a one-lane. When a vehicle reaches ***the router***, it stops broadcasting safety messages and heads for an assigned VLPZ's lane. The assigned lane is randomly and privately selected by ***the router***. The vehicle can then reside inside a VLPZ for a random period of time depending on the service time. A vehicle must change its pseudonym before leaving the VLPZ and all vehicles exit a VLPZ through ***the aggregator***. As discussed in [6], this strategy provides the protection not only against both of the syntactic and the semantic linking of pseudonyms but also against the FIFO attacks. In addition, differently, from the strategies that rely on the radio silence technique, road safety is preserved in this strategy. The reader can refer to [6] [7] and [10] for further information.

B. Related work

Many Pseudonym-Changing Strategies (PCSs) have been proposed to protect location privacy in vehicular networks from linking attacks. These strategies can be classified into three categories [5]: (i) Synchronized pseudonym-changing process based strategies [11–15]. These strategies are weak since the contents of safety messages can be used to link the pseudonyms; (ii) Encrypted safety messages based strategies [16–18]. These strategies can also be broken since internal passive adversaries can decrypt safety messages and provide a clue to the external global passive adversary to link the pseudonyms, and finally, (iii) Radio silence based strategies [19][20, 21][22, 23], which are more effective than the previous ones since they provide protection against both external and internal passive adversaries. However, the use of radio silence is challenging in vehicular networks due to its impact on road safety [24]. The authors of [6] proposed VLPZ-based PCS, which uses also the radio silence. This strategy provides strong protection against the pseudonym linking attacks and preserves also road safety. The authors of [7, 10] proposed PRIVANET, which is a framework which uses VLPZs. In this framework, VLPZs are thus equipped with RSU_{VLPZs} to notify their presence and to distribute the pseudonyms sets. Additionally, PRIVANET proposed a reputation-based mechanism to motive selfish vehicles to enter VLPZs. It is worth to mention that the optimal placement problem of VLPZs was preliminary formulated. However, only a simple illustration solution was provided. The study of [25] suggested providing 43,800 pseudonyms per year for a vehicle to avoid linking attacks. This number mainly depends on the pseudonyms changing frequency. A huge number of pseudonyms should thus be stored in each vehicle, which can exceed vehicle storage capabilities. [26] pointed out that pseudonyms are scarce resources and costly acquired and managed and hence should be efficiently used.

III. System Model and Problem formulation

In this section, we present the proposed software defined vehicular network architecture and give the formalization of the optimal placement of VLPZs problem.

A. Vehicular system model

We consider a software-defined vehicular network architecture. As illustrated in Figure 2, this architecture has one level of SDN control consisting of the global SDN controller that has full knowledge about the vehicular network. The data forwarding plane consists of vehicles and Road Side Units (RSUs). Each vehicle is equipped with 802.11p interface to communicate with other vehicles and with RSUs. Each RSU is also equipped with two interfaces. A wired link to communicate with the neighboring RSUs and an LTE interface to communicate with the global SDN controller. An SDN agent is also run on each vehicle and RSU. The communication links between the global SDN controller and the data plane are secured. We also consider that road area contains a set of RIs managed by trusted authorities. RIs periodically send updates including their current capacity to the global SDN controller. The internal architecture of the global SDN controller mainly consists of three modules:

1) **Roadside Infrastructure Module (RIM)**: it collects information and updates about the RIs.
2) **Mobility and Topology Module (MTM)**: it collects the mobility information of vehicles.
3) **VLPZ Placement Genetic Algorithm (VPGA)**: it selects periodically the best RIs to host the VLPZs based on the information provided by RIM and MTM. When a vehicle decides to enter a VLPZ, it sends a request to the global SDN controller. This latter uses the solution provided by the VPGA to assign each vehicle to the adequate VLPZ.

Each vehicle periodically broadcasts a safety message every t millisecond, where each message includes a location, a time, a velocity and a content. Before joining the vehicular network, each vehicle registers with the CA (certification authority). During registration, each vehicle V_i is pre-loaded with a set of m pseudonyms $K_{i,k}$ where k ∈ {1,..., m }, that are, public keys certified by the CA. For each pseudonym $K_{i,k}$ of a vehicle V_i, the CA provides a certificate $Cert_{i,k}(K_{i,k})$. The safety messages are properly signed by private key $K_{i,k}^{-1}$ corresponding to the pseudonym $K_{i,k}$ to ensure the authentication. A certificate is attached to each message to enable other vehicles to verify the sender's authenticity.

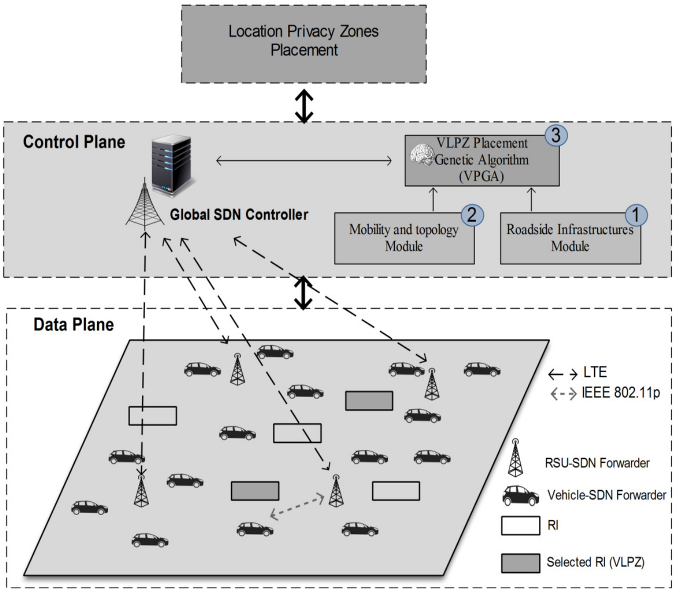

Location Privacy Zones Placement
Control Plane
Global SDN Controller
VLPZ Placement Genetic Algorithm (VPGA)
3
Mobility and topology Module
2
Roadside Infrastructures Module
1
Data Plane
LTE
IEEE 802.11p
RSU-SDN Forwarder
Vehicle-SDN Forwarder
RI
Selected RI (VLPZ)

Fig. 2: Software defined vehicular network architecture

B. Problem formalization

Here we answer the following question: Given m RIs that exist in a road area, with m>= N_{max}, what are the best RIs that should deploy VLPZs in order to reduce the trajectory cost of vehicles ?

To answer to this question, we formulate the problem as follows: Let *i*= { 1,...,n} the set of existing vehicles at time t. Let *j*={1,...,m} be the set of the candidate RIs to deploy the required VLPZs. Let c_{ij} the trajectory cost of a vehicle v_i to move to a RI_j. Let y_j a binary decision variable, which indicates that the RI is selected to host a VLPZ at time t. x_{ij} is a binary variable, which indicates that the vehicle v_i is assigned to RI_j or not.

To select the best RIs that host the required number VLPZs, we should minimize the following objective function **F**, which aims to minimize the trajectory cost of vehicles when moving to the assigned VLPZ [10].

$$F = min \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} x_{ij} \quad \text{... (1)}$$

The transportation cost c_{ij} can be expressed as the time spent by a vehicle v_i to reach a candidate RI_j and quantified by the loss of pseudonyms during this time, which can be calculated using the following formula:

$$cij = \frac{d_{ij}}{v} * \eta \quad \text{... (2)}$$

- d_{ij}: the distance between a vehicle *i* and a candidate RI_j
- *v*: the average speed of vehicles (meter/second).
- η: the frequency of changing of pseudonym (pseudo/second).

We assume that *v* and η are fixed values. Thus, the objective function F can be rewritten as function of d_{ij} as follows:

$$F = min \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} x_{ij} \quad \text{... (3)}$$

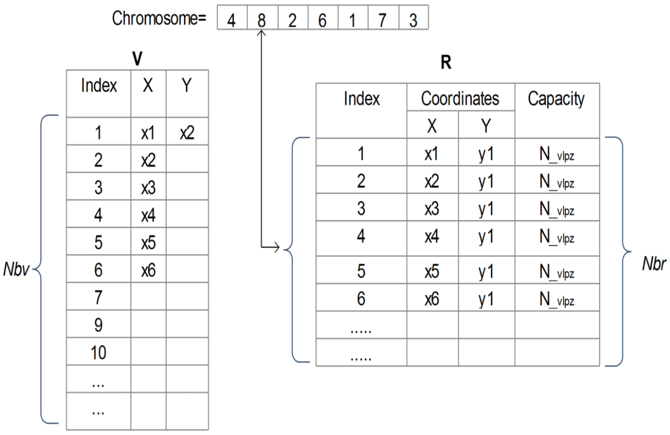
The feasibility of the solution depends on different constraints, which are represented by the following equations:

$$\begin{cases} \sum_{j=1}^{m} x_{ij} = 1 & \text{... (4)} \\ \sum_{j=1}^{m} y_j = N_{vlpz}(t) & \text{... (5)} \\ \sum_{j=1}^{n} x_{ij} <= K_{opt} & \text{... (6)} \\ x_{ij} \in \{0,1\} & \text{... (7)} \\ y_j \in \{0,1\} & \text{... (8)} \end{cases}$$

(4) ensures that each vehicle v_i is only assigned to one RI; (5) ensures that the number of selected RIs is equal to the number of VLPZ that are needed at time t (N_{vlpz}(t)). (6) guarantees that the number of vehicles that are assigned to each infrastructure does not exceed the capacity of the RI (K_{opt}); and finally, (7) and (8) are the integrity constraints.

TABLE I: The description of variables

Variable	Description
c_{ij}	the trajectory cost of a vehicle v_i to move to RI_j
y_j	A binary decision variable which indicates that RI is selected to deploy a VLPZ at time t
x_{ij}	A binary variable, which indicates that v_i is assigned to RI_j.
N_{max}	the maximum number of required VLPZs at the road area.
N_{vlpz}(t)	the number of required VLPZs at a given time t.
K_{opt}	the number of vehicles that can be hosted by the RI
d_{ij}	the distance between a v_i and RI_j
v	the average speed of vehicles (m/s)
η	the frequency of changing of pseudonym (pseudo/s)
R	Information table of RIs
V	Information table of vehicles
A	Assignment table



Chromosome=	4	8	2	6	1	7	3

V

Index	X	Y
1	x1	x2
2	x2	
3	x3	
4	x4	
5	x5	
6	x6	
7		
9		
10		
...		
...		

R

Index	Coordinates		Capacity
	X	Y	
1	x1	y1	N_vlpz
2	x2	y1	N_vlpz
3	x3	y1	N_vlpz
4	x4	y1	N_vlpz
5	x5	y1	N_vlpz
6	x6	y1	N_vlpz
.....			
.....			

Fig. 3: Chromosome representation

IV. VPGA: A Genetic Algorithm for an Optimal Placement of VLPZs

As shown in [27], finding an optimal solution for the VLPZs placement is a NP-hard problem. Hence, we explore approximation techniques, and model the problem as finding the best fitted solution according to a genetic algorithmic model of the problem, after a fixed amount of generations have been explored. In this vein, we propose a VLPZ Placement Genetic Algorithm (VPGA) for an optimized placement of VLPZs within RIs. The pseudo-code of VPGA is illustrated in Algorithm 1. VPGA takes as input the current mobility information of vehicles and the positions of RIs and returns the VLPZs placement decision. In the following, the phases of VPGA are detailed.

A. Chromosome representation

In VPGA, each candidate solution is presented as a chromosome that is a chain of integers where each value is the index number of a potential RI. The length of the chromosome is equal to the optimal number of required VLPZs. As illustrated in Figure 3, the coordinates (x,y) and the capacities of the potential RIs are stored in **R**, which is a 2D Array $(3 * m)$. The vehicle coordinates are also stored in **V**, which is 2D Array(2*n). **V** is updated periodically according to the mobility of vehicles.

B. Initialisation Phase

In this phase, the initial population of chromosomes is generated and vehicles are also assigned to each generated chromosome in order to compute the fittest chromosome. To cover the whole search space, the initial population is randomly generated. In addition, the size of the generated population is maintained in each iteration, which equals to the size of the initial population. However, a simple generation of the population could generate invalid chromosomes, which do not satisfy the constraint (4). For this reason, as described is Subroutine 1, for each generated gene, VPGA checks if it has already been added to the given chromosome or no. The random population procedure runs until the generation of all chromosomes. The assignment and the fitness calculation procedures are described in subsections IV-F and IV-G respectively.

C. Selection Phase

Selection is the first procedure to build a new population. A set of chromosomes from the old population should be selected to be parents for the rest of the procedures (crossover and mutation). VPGA uses two selection methods: elitism and tournament. Elitism selects the best fittest chromosomes from the old population and adds them to the new population. As described in Subroutine 2, VPGA only selects the best fittest chromosome and copy it to the new created population. VPGA also uses the tournament method to select the parents that are used by the crossover to generate new chromosomes. The tournament selection method randomly chooses a set of chromosomes from the old population. The size of this set should be equal to the tournament size. After that, the fitness of each tournament chromosome is evaluated and the fittest chromosome is selected as a parent for the crossover.

D. Crossover Phase

The crossover is a convergence operation that are used to generate new offsprings for the new population. It is intended to pull the population towards a local min or max. Crossover selects genes from the selected parents to create the new chromosome. As described in Subroutine 3, the crossover runs until the generation of the new population. In each iteration, a new chromosome is created based on the two parent chromosomes which were selected using the tournament selection method. The genes of the new chromosome are selected using the uniform crossover i.e. the genes are randomly copied from the first or the second parent. The crossover computes the probability that determines from which parents the gene comes. Then the new chromosome is added to the new population.

E. Mutation Phase

Contrarily to crossover, the mutation is a divergence operation which is intended to occasionally break one or more members of a population out of a local min/max space and potentially discover a better space. The mutation operator works on a single chromosome. It aims to randomly introduce a new gene instead of inheriting from the old chromosomes. The purpose is to avoid the local optimal covering the whole search space. As described in subroutine 4, the mutation runs until the generation of the new population. In each iteration, the genes of each chromosome are changed according to the mutation probability. This latter is used to determine whether the gene should be changed or not. In case a change is needed, a gene is randomly generated from the whole search space.

F. Assignment Phase

The next procedure after the generation or the update of the population is the assignment of vehicles to genes (RIs) of each chromosome in order to be able to evaluate the fitness. VPGA uses two algorithms of assignment: (i) The classical assignment: that calculates the Euclidean distance between each vehicle v_i and each candidate RI, and (ii) The clustering-based assignment: that uses the $K\text{-}means$ same size algorithm to create clusters of vehicles that have a same size, which equals to the capacity of the RI. The clustering-based assignment calculates the Euclidean distance between the cluster centroids and each candidate RI.

1) Classical assignment: consists of three steps: (i) Compute the Euclidean distances between each vehicle v_i and each candidate RI. These distances are saved in the distance table (D); (ii) Sort D from the lowest to the highest distance value; and (iii) Assign each vehicle to the nearest RI and save this assignment in **A**.

2) Clustering-based assignment: consists of four steps: (i) Create same-size clusters of vehicles. VPGA uses a variation of k-means clustering algorithm, proposed by ELKI Framework to create these clusters [28]; (ii) Calculate the distances between each centroid of a cluster and each candidate RI and save them in (D); (iii) Sort D from the lowest distance value to the highest one; and (iv) Assign each centroid to the nearest RI and save these assignments on **A**.

G. Fitness evaluation

Each generation of the genetic programming approach goes through mutations and crossovers. The newly generated solutions are evaluated according to a fitness function. We derive the fitness function according to the objective functions defined in the ILP formulation, namely equation (1). The variables in the paper are described in Table I.

H. Stop conditions

A genetic algorithm requires certain stop conditions to terminate. In VPGA, we consider two stop conditions related to two different aspects. The first condition is related to the convergence of our solution: if the fitness value keeps unchanged during three iterations, we assume that the optimal value of the fitness is reached and the algorithm should be terminated. The second condition is linked to the number of iterations. We have simply limited the maximum number of iterations. VPGA returns the fittest chromosome .i.e. chromosome with the minimal fitness value.

Algorithm 1: VLPZ Placement Genetic Algorithm

```
Data: Mobility information of Vehicles and RIs
Result: VLPZs placement decision
Initialize
    Build local variables: V, R, etc. ;
    Generate initial population;
    Assignment;
    Fitness evaluation;
Main process
    while termination conditions not satisfied do
        New population (Selection, Crossover, Mutation);
        Assignment;
        Fitness evaluation;
    end
    return the fittest chromosome
Subroutine 1 — Random Population Generation
    Data: Local variables
    Result: Population (P)
    while i < population_size do
        while j < chromosome_size do
            P[i,j] ← Randomly generate a new gene ;
        end
    end
Subroutine 2 — Selection
    Data: Population
    Result: Updated population
    Fittest Chromosome ← P[0];
    for  i= 1 ... population size do
        if fitness (P[i]) > fitness (Fittest Chromosome) then
            Fittest Chromosome ← P[i];
        end
    end
    P[0] ← Fittest Chromosome;
Subroutine 3 — Crossover
    Data: Population
    Result: Updated population
    for i= 1 ... (population_size-1) do
        do
            Chromosome1 ← TournamentSelection();
            Chromosome2 ← TournamentSelection();
            for  (j= 0... (chromosome_size-1)) do
                if random()<= crossover_probability then
                    New_chromosome[j] ← Chromosome1[j] ;
                else
                    New_chromosome[j] ← Chromosome2[j] ;
                end
            end
        while !checking(New_chromosome);
        P[i] ← New_chromosome ;
    end
Subroutine 4 — Mutation
    Data: Population
    Result: Updated population
    for i= 1 ... (population_size-1) do
        for j= 0... (chromosome_size-1) do
            if random() <= mutation probability then
                P[i,j] ← generate new gene;
            end
        end
    end
```

V. Numerical results

In this section, we evaluate the performance of VPGA considering the classical and clustering-based assignment. VPGA is one of the main functions of the vehicular SDN controller: optimal VLPZs placement. To show the merit of our approach, we compare it to the solution already proposed in PRIVANET [10]. VPGA is programming and implemented using Java programming language and run on Intel i5 2.6 GHz.

Table II shows the parameters used by VPGA. We have considered three levels of Vehicular Density (VD): Low (LVD), Medium (MVD), and High (HVD) for 100, 150, and 200 vehicles/km^2 respectively. We varied also the number of RI from 15 to 35. The capacity of each RI is fixed to 15. We set the size of the generated population in each iteration to 50. The size of the chromosome of is calculated according to the following formula:

$$Chromosome\_size = \left[\frac{Number\_Vehicles}{Capacity\_RI} \right]$$

The performance of VPGA depends on the crossover and the mutation operators. For this reason, we fixed the tournament size and the elitism parameters to 5 and 1 respectively and varied the crossover probability and the mutation probability from 5% to 95% respectively. Each test is repeated 10 times and the results are calculated with 95% of the confidence interval.

TABLE II: Simulation Parameters

Parameter	Value
Number of tests	10,100
Population size	50
Crossover probability	[0.05-0.95]
Mutation probability	[0.05-0.95]
Tournament size	5
Elitism set size	1
Number of vehicles	100, 150, and 200
Number of RIs	15,20, 25, 30, and 35
Capacity of RI	15

A. Fitness Comparison

Figure 4 compares the obtained fitness values using VPGA with its variations (classical and clustering-based assignments), and PRIVANET [10]. In this evaluation, the position of vehicles and RIs are generated before the beginning of each iteration. We have considered the case of MVD and varied the number of RI from the lowest (15) to the highest value (35). As we can see, the best value of fitness is obtained when using VPGA with the classical assignment. The fitness decreases gradually when the number of RIs increases. This is due to the fact that with a large number of RIs, a high number of RIs will be in the vicinity of vehicles, hence the distances between vehicles and RIs are minimized.

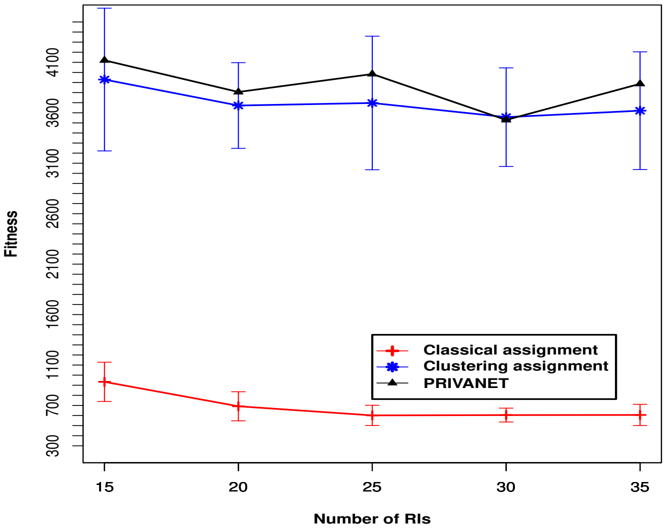


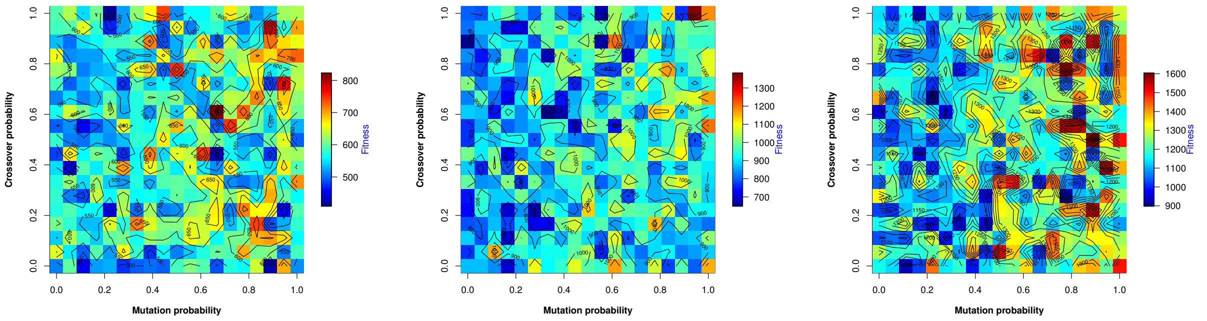
Fig. 4: Fitness comparison under different approaches.

B. Impact of vehicular density

We evaluate in Figure 7, the fitness and the convergence speed obtained under different vehicles density (LVD, MVD, and HVD). As we can see in Figure 7a the fitness decreases with the increase in the number of RIs for all VDs. For LVD and MVD, the fitness values approximately keep stable values between 25 and 35 RIs. However, for high densities, the value of fitness is enhanced in this interval. The reason for that with a high density of vehicles and with a large number of RIs the distances between the vehicles and RIs will be short. As a result, the fitness value decreased. Figure 7b illustrates the speed convergence under different vehicle densities. We notice that the number of iterations increases with the number of RIs for all vehicle densities. Additionally, the convergence speeds of vehicle densities are close when the number of RIs is equal to 35. These results can be explained that with a large number of RIs, the search space of VPGA will be larger. Consequently, VPGA takes more iterations to reach the fittest chromosome whatever the vehicle densities are.

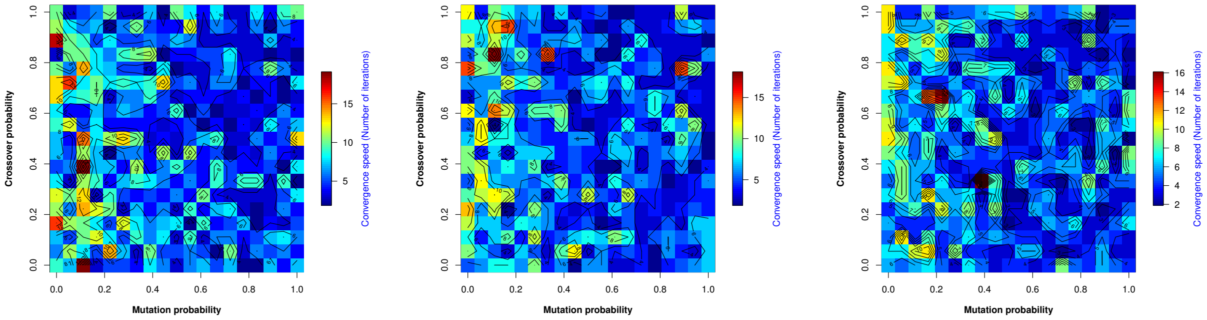
C. Parameters tuning

We evaluate in Figure 5 and 6 the impact of the crossover probability and the mutation probability on the obtained fitness and convergence speed values respectively under different VDs. The blue zones in the contour plots are the minimum values of fitness and convergence speed respectively. We can see in Figure 5 that the density of the blue color is higher when the mutation probability between 5% and 20% and the crossover probability between 50% and 90%. Figure 5 shows that the density of the blue color is higher when the mutation probability is greater than 20%. To this end, the mutation and the crossover probabilities should carefully be tuned to establish the equilibrium between the fitness and convergence speed. In VPGA, the best results are obtained when the mutation probability equals 20% and the crossover probability $\in [50, 90]\%$.



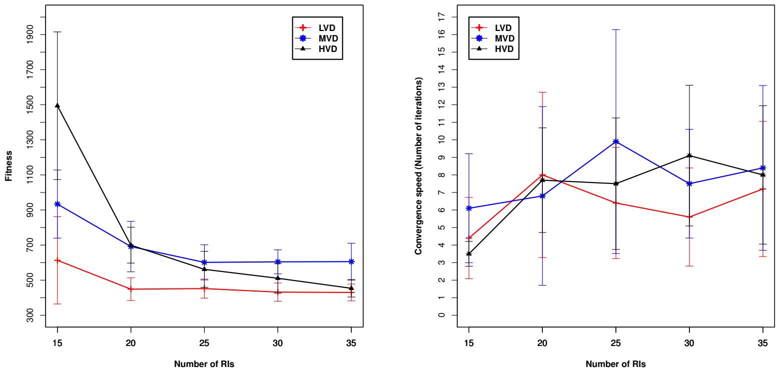
(a) LVD (b) MVD (c) HVD

Fig. 5: Fitness evaluation with different crossover and mutation probabilities under different VDs.



(a) LVD (b) MVD (c) HVD

Fig. 6: Convergence speed evaluation with different crossover and mutation probabilities under different VDs.



(a) Fitness. (b) Convergence speed.

Fig. 7: Fitness and convergence speed comparison under different VDs.

D. *Response time of the SDN controller*

To run adequately, VPGA needs an accurate input such as number of RIs, densities of the traffic, coordinates of vehicles, etc. This input is provided by the SDN controller which supervises the behavior of the moving vehicles via transmitted beacons and get information about the RIs from authorities. In our SDN-enabled architecture, centralized control operations require less signaling traffic and shorter delays. When a change occurs in the network, the SDN knowledge is updated. Going further, we have compared the performances of our SDN-enabled architecture in terms of response time of SDN controller under different vehicles densities. The response time is the time taken by the SDN controller to select the placement of the VLPZs. Recall that VLPZs placement are periodically calculated with SDN-controller As shown in Figure 8, the response time increases with vehicular density. The maximal value is 7 seconds which is observed under HVD.

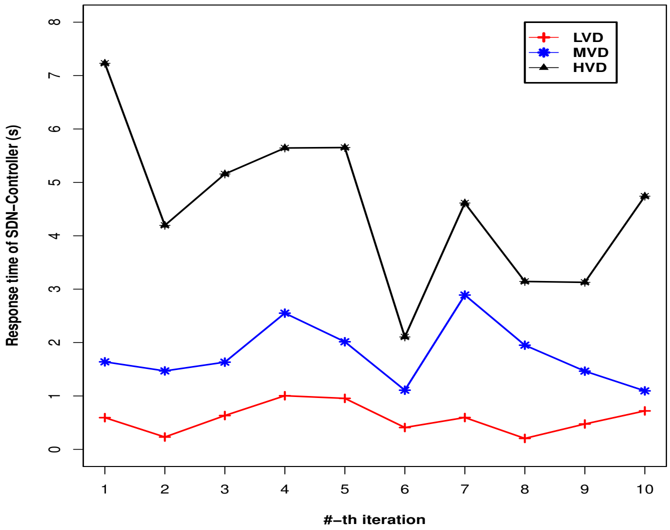


Fig. 8: Response time of the SDN controller under different VDs.

VI. Conclusion

In this paper, we proposed a cost-efficient vehicular location privacy zones placement strategy that relies on the emerging concept of SDN in vehicular networks and a meta-heuristic based on a genetic algorithm to provide an optimal and a dynamic placement solution. To this end, we provided a modeling and mathematical formulation of the problem that served as a basis to derive constraints and criteria for the the proposed genetic algorithm. Comparative analytical results with previous studies illustrate that our strategy has better performances especially in the reducing the number of used pseudonyms and the response time.

Acknowledgment

This work was supported by the H2020 5G-DRIVE project (ID: 814956).

References

[1] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung, and J. P. Hubaux", "Secure vehicular communication systems: Design and architecture," *IEEE COMMUNICATIONS*, vol. 46, no. 11, pp. 100–109, 2008.

[2] IEEE, "IEEE standard for wireless access in vehicular environments security services for applications and management messages," *IEEE Std 1609.2-2013 (Revision of IEEE Std 1609.2-2006)*, pp. 1–289, April 2013.

[3] ETSI, "ETSI ts 102 941 v1.1.1- intelligent transport systems (its); security; trust and privacy management," *Standard, TC ITS*, 2012.

[4] B. Wiedersheim, Z. Ma, F. Kargl, and P. Papadimitratos, "Privacy in inter-vehicular networks: why simple pseudonym change is not enough," in *Proceedings of the 7th international conference on Wireless on-demand network systems and services*, ser. WONS'10. IEEE Press, 2010, pp. 176–183.

[5] A. Boualouache, S.-M. Senouci, and S. Moussaoui, "A survey on pseudonym changing strategies for vehicular ad-hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 770–790, 2017.

[6] ——, "VLPZ: The vehicular location privacy zone," in *The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016)*. Procedia Computer Science Elsevier, 2016.

[7] ——, "Towards an efficient pseudonym management and changing scheme for vehicular ad-hoc networks," in *2016 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2016, pp. 1–7.

[8] X. Huang, R. Yu, J. Kang, N. Wang, S. Maharjan, and Y. Zhang, "Software defined networking with pseudonym systems for secure vehicular clouds," *IEEE Access*, vol. 4, pp. 3522–3534, 2016.

[9] X. Huang, J. Kang, R. Yu, M. Wu, Y. Zhang, and S. Gjessing, "A hierarchical pseudonyms management approach for software-defined vehicular networks," in *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*. IEEE, 2016, pp. 1–5.

[10] A. Boualouache, S. Senouci, and S. Moussaoui, "PRIVANET: An efficient pseudonym changing and management framework for vehicular ad-hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2019.

[11] J.-H. Song, V. Wong, and V. Leung, "Wireless location privacy protection in vehicular ad-hoc networks," *Mobile Networks and Applications*, vol. 15, no. 1, pp. 160–171, 2010.

[12] M. Gerlach and F. Gttler, "Privacy in vanets using changing pseudonyms - ideal and real," in *VTC Spring*. IEEE, 2007, pp. 2521–2525.

[13] J. Liao and J. Li, "Effectively changing pseudonyms for privacy protection in vanets," in *Pervasive Systems, Algorithms, and Networks (ISPAN), 2009 10th International Symposium on*. IEEE, 2009, pp. 648–652.

[14] R. Lu, X. Lin, T. H. Luan, X. Liang, and X. Shen, "Pseudonym changing at social spots: An effective strategy for location privacy in vanets," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 1, pp. 86–96, Jan 2012.

[15] D. Eckhoff, R. German, C. Sommer, F. Dressler, and T. Gansen, "Slotswap: Strong and affordable location privacy in intelligent transportation systems," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 126–133, Nov 2011.

[16] J. Freudiger, M. Raya, M. Felegyhazi, P. Papadimitratos, and J. Hubaux, "Mix-zones for location privacy in vehicular networks," in *Proceedings of the First International Workshop on Wireless Networking for Intelligent Transportation Systems (Win-ITS)*, 2007.

[17] A. Wasef and X. Shen, "Rep: Location privacy for vanets using random encryption periods," *Mobile Networks and Applications*, vol. 15, no. 1, pp. 172–185, 2010.

[18] B. Ying, D. Makrakis, and H. Mouftah, "Dynamic mix-zone for location privacy in vehicular networks," *Communications Letters, IEEE*, vol. 17, no. 8, pp. 1524–1527, August 2013.

[19] L. Buttyan, T. Holczer, A. Weimerskirch, and W. Whyte, "Slow: A practical pseudonym changing scheme for location privacy in vanets," in *In: Proceedings of the IEEE Vehicular Networking Conference (VNC)*, Tokyo, Japan. IEEE, Los Alamitos, 2009.

[20] K.Sampigethaya, M. Li, L. Huang, and R. Poovendran, "Amoeba: Robust location privacy scheme for vanet," *IEEE Journal on Selected Areas in Communications*, p. 1589, 2007.

[21] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki, "Caravan: Providing location privacy for vanet," in *in Embedded Security in Cars (ESCAR)*, 2005.

[22] A. Boualouache and S. Moussaoui, "S2si: A practical pseudonym changing strategy for location privacy in vanets," in *Advanced Networking Distributed Systems and Applications (INDS), 2014 International Conference on*, June 2014, pp. 70–75.

[23] ——, "Urban pseudonym changing strategy for location privacy in vanets," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 24, no. 1-2, pp. 49–64, 2017.

[24] S. Lefevre, J. Petit, R. Bajcsy, C. Laugier, and F. Kargl, "Impact of V2X privacy strategies on intersection collision avoidance systems," in *Vehicular Networking Conference (VNC), 2013 IEEE*, Dec 2013, pp. 71–78.

[25] M. Raya and J. Hubaux, "Securing vehicular ad hoc networks," *Journal of Computer Security*, vol. 15, no. 1, pp. 39–68, jan 2007.

[26] J. Freudiger, M. H. Manshaei, J.-P. Hubaux, and D. C. Parkes, "Non-cooperative location privacy," *Dependable and Secure Computing, IEEE Transactions on*, vol. 10, no. 2, pp. 84–98, 2013.

[27] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1990.

[28] "Same-size k-means variation," http://elki.dbs.ifi.lmu.de/wiki/Tutorial/SameSizeKMeans, last accessed date: 2019-07-05.