



HAL
open science

A Multi-Objectif Genetic Algorithm-Based Adaptive Weighted Clustering Protocol in VANET

Mohamed Hadded, Rachid Zagrouba, Anis Laouiti, Paul Mühlethaler, Leila Azouz Saidane

► **To cite this version:**

Mohamed Hadded, Rachid Zagrouba, Anis Laouiti, Paul Mühlethaler, Leila Azouz Saidane. A Multi-Objectif Genetic Algorithm-Based Adaptive Weighted Clustering Protocol in VANET. CEC'2015: IEEE Congress on Evolutionary Computation, May 2015, Sendai, Japan. pp.994 - 1002, 10.1109/CEC.2015.7256998 . hal-01211442v1

HAL Id: hal-01211442

<https://hal.science/hal-01211442v1>

Submitted on 6 Oct 2015 (v1), last revised 25 Nov 2015 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Multi-Objective Genetic Algorithm-Based Adaptive Weighted Clustering Protocol in VANET

Mohamed Hadded ^{*‡}, Rachid Zagrouba ^{*}, Anis Laouiti [‡], Paul Muhlethaler [†] and Leila Azouz Saidane ^{*}

^{*}RAMSIS Team, CRISTAL Laboratory, 2010 Campus University, Manouba, Tunisia

[†]INRIA, BP 105. 78153 Le Chesnay Cedex, Paris-Rocquencourt, France

[‡]TELECOM SudParis, CNRS Samovar, UMR 5157, France

{mohamed.hadded, leila.saidane}@ensi.rnu.tn, rachid.zagrouba@cristal.rnu.tn, anis.laouiti@it-sudparis.eu, paul.muhlethaler@inria.fr

Abstract—Vehicular Ad hoc NETWORK (VANET) is the main component that is used recently for the development of Intelligent Transportation Systems (ITSs). VANET has a highly dynamic and partitioned network topology due to the constant and rapid movement of vehicles. Recently, the clustering algorithms are widely used as the control schemes to make VANET topology less dynamic for MAC, routing and security protocols. An efficient clustering algorithm must take into consideration all the necessary information related to node mobility. In this paper, we propose an Adaptive Weighted Clustering Protocol (AWCP), specially designed for vehicular networks, which takes the highway ID, direction of vehicles, position, speed and the number of neighbors vehicles into account in order to enhance the network topology stability. However, the multiple control parameters of our AWCP, make parameter tuning a non-trivial problem. In order to optimize AWCP protocol, we define a multi-objective problem whose inputs are the AWCPs parameters and whose objectives are: providing stable cluster structure as possible, maximizing data delivery rate, and reducing the clustering overhead. We then face this multi-objective problem with the the Multi-Objective Genetic Algorithm (MOGA). We evaluate and compare its performance with other multi-objective optimization techniques: Multi-objective Particle Swarm Optimization (MOPSO) and Multi-objective Differential Evolution (MODE). The experiments analysis reveal that NSGA-II improves the results of MOPSO and MODE in terms of the spacing, spread, and ratio of non-dominated solutions and generational distance metrics used for comparison.

I. INTRODUCTION

A Vehicular Ad hoc NETWORK (VANET) were deployed to make the communication between a set of vehicles possible through ad hoc wireless devices. Nowadays, these networks are used for wide range of applications which can be divided into the following three services namely, safety services, traffic management and user-oriented services. The inter-vehicle communication V2V (Vehicle To Vehicle) enables each vehicle to provide a warning to the driver in real time when a crash is predicted. The warning message can be either through a seat vibration, tone or visual display or combinations of these indicators. Moreover, in order to make V2V communication work, the FCC (Federal Communication Commission) [1] has established a wireless protocol similar to WiFi, called Dedicated Short Range Communications or DSRC for short [2]. The DSRC radio technology is defined in the frequency band of 5.9 GHz with a total bandwidth of 75 MHz. This band is divided into 7 channels of 10 MHz for each one. These channels comprise one Control CHannel (CCH) reserved for

the network management and high priority messages and six Service CHannels (SCHs) dedicated to data transmission.

Due to high vehicle mobility, supporting network connection introduces high communication overhead for exchanging and updating the topology information [3]. For instance, in flat-topology network, each vehicle is required to periodically maintain its own connectivity to other one hop neighboring vehicles. Without using expensive components such as central points (RSUs), the establishment of a hierarchical clustering structure within the network can reduce the relative mobility between neighboring vehicles, and communication overhead [4]. The clustering allows the formation of organized groups used to coordinate the channel access [6], to simplify routing [5], and security [7]. However, the main issue for clustering protocols in VANETs networks is ensuring topology stability. Thus, efficient clustering protocol should take into account many mobility metrics to form stable clusters, and also maintain the current cluster structure with less overhead. In this paper we propose a multi-metrics based adaptive weighted clustering protocol in VANET that takes advantage of the geographic information of vehicles. The main contributions of the paper are listed below

- We propose a clustering protocol based on WCA algorithm [12] for VANETs networks in which a vehicle only considers neighbors moving in the same Highway and in the same direction, and ignores others broadcasts messages otherwise.
- We define a solution vector of real and integer variables that can be fine tuned to obtain an efficient QoS AWCP configurations.
- We formulate the parameter tuning problem of AWCP protocol as a Multi-Objective Linear Programming MOLP and we propose an optimization strategy in which the *Non-dominated Sorting Genetic Algorithm, version 2* NSGA-II [13] is combined with a ns2 simulator to solve the MOLP problem.
- We use a realistic VANET mobility scenarios taken from the metropolitan area of Tunis (Tunisia), in order to find and validate the best optimal configuration.

The rest of the paper is organized as follows. In Section 2, we present related work. Section 3 presents our clustering protocol called AWCP. Section 4 describes the AWCA QoS problem and identifies the AWCA parameters and performance

criterion. Section 5 describes the optimization methodology which consists in combining an NSGA-II and a network simulator to determine the optimal parameters of AWCP. Section 6 shows the simulation results and the performance evaluation. Finally, conclusions and future work are reported in Section 7.

II. RELATED WORK

Several researches focus on developing clustering protocols for VANET, while most of them are based on the MANET clustering techniques. However, none of the protocols proposed takes into consideration the highways ID for clusters formation in VANET. As results, these protocols do not create a stable clustering architecture. Some of these proposed protocols are described below.

In [8], the authors proposed a lane-based clustering algorithm designed to extend the cluster lifetime and reduce the communication overhead. The cluster head is selected based on the lane where most of the vehicles will flow. The authors have supposed that each vehicle knows its exact lane on the road via a lane detection system and an in-depth digital street map that includes lane information. A lane weight (LW) metric is applied for each traffic flow in order to select the most stable clusterhead. The clustering algorithm involves only the cluster formation phase where all vehicles are assumed to follow a steady roadway and does not involve cluster maintenance phase where the vehicles change their mobility directions or lanes.

A Multi-Head Clustering Algorithm was proposed in [9]. This technique intends to create stable clusters and reducing re-clustering overhead by supporting single and multiple cluster head. In the cluster head election phase, all vehicles that are in communication range to each other are organized into clusters and one vehicle for each cluster is elected to act as master cluster head (MCH). Then, some cluster members from a cluster are selected to be slave cluster heads (SCHs). In order to form stable clusters, the authors have restricted that all vehicles in a cluster have the same moving direction.

The authors proposed in [4] a multi-metric algorithm for cluster-head elections suitable for highway area with the aim of achieving better results for network stability as well as decreasing the dynamic nature of VANET. In addition to the position and direction, this algorithm uses speed difference metric as a new parameter to increase the cluster lifetime. The vehicles that are moving with high mobility speed are grouped in one cluster while the vehicles moving with low speed are grouped in another cluster.

Several others clustering algorithms designed for Mobile Ad hoc NETwork (MANET) that operates also in VANET and used frequently for comparison with other VANET clustering protocols. For instance, the Lowest-ID clustering algorithm [14] is based on electing a node with the smallest ID as a cluster head, where each node has a fixed ID. The Highest-Degree algorithm [15] which selects a node as a cluster head based on the node connectivity. The node with the maximum number of neighbors is elected as the clusterhead. The Weighted Clustering Algorithm (WCA) [12] elects a node to act as a clusterhead based on a combined weight which includes the number of neighbors, average distance, average speed, and battery-life. This algorithm also restricts the number

of nodes in a cluster so that the performance of the MAC protocol is not degraded because it is always desirable for a cluster head to handle up to a certain number of nodes in its cluster. However, the performance of these approaches is significantly influenced by the value of their control parameters (such as Hello_Interval, Election_Interval, Timeout_Interval, ...). Finding the optimal values of the control parameters is the major issue. Moreover, the authors do not provide guidelines to tune and optimize them for various mobility scenarios. In this paper we propose a multi-objectif genetic algorithm-based adaptive weighted clustering protocol in VANET that takes advantage of the geographic information of vehicles.

III. ADAPTIVE WEIGHTED CLUSTERING ALGORITHM DESCRIPTION

AWCP is an improvement of WCA [12] protocol which classifies vehicles into groups based on their highway ID and mobility direction. Since clusters with vehicles having diverse moving directions and different highway ID are unstable, we restrict that all vehicles in a cluster to have the same highway ID and the same moving direction.

A. System Model

Our proposed algorithm is based on the assumption that each vehicle in VANET can know its highway ID via a digital road map and a positioning system, e.g. GPS (Global Positioning System) or GALLILEO receiver that allows it also to obtain an accurate real-time three-dimensional geographic position (latitude, longitude and altitude), direction, speed and exact time. The VANET network consists of a set of Roadside Units RSU and a set of vehicles moving in opposite directions on two-way vehicle traffic roads [16]. A vehicle is said to be moving in a left (right) direction if it is currently heading to any direction from north (west) to south (East), as shown in Figure 3.

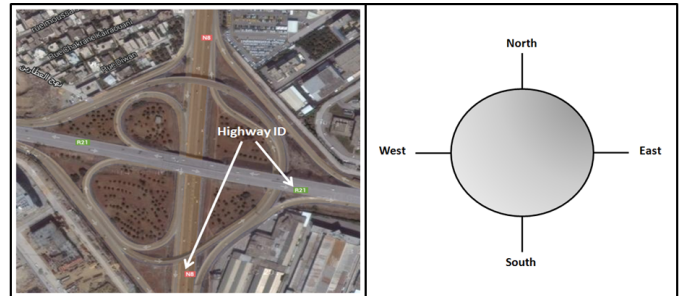


Fig. 1. Right and left direction in VANET network

In the next sections, we will describe how AWCP elects clusterhead and maintains stable cluster structure able to mitigate the VANET issues related to the network connection and reduce the relative mobility between neighboring vehicles with less communication overhead.

B. Cluster Head Election

Initially, all vehicles are in the Undecided State (US). To divide the network into clusters, each vehicle changes its state to Cluster-Head Candidate (CHC) and it starts to periodically

broadcast a HELLO message containing all of the necessary information "highway ID, direction, position, speed" to its One-Hop neighbors OH. In order to form stable clusters, each vehicle only considers neighbors moving in the same highway and in the same direction, and ignores broadcasts from vehicles in the opposite direction. The HELLO broadcast algorithm is outlined in Algorithm 1 while S_i is the current state of vehicle i and *Hello_Interval* is the time interval for a vehicle to broadcast a HELLO message. Upon reception of a HELLO

Algorithm 1 Broadcast of HELLO Messages

- 1: **for** each vehicle i **do**
 - 2: $S_i \leftarrow CHC$
 - 3: **end for**
 - 4: **for** every *Hello_Interval* seconds and each vehicle j **do**
 - 5: Update the geographic information: highway ID, direction, position, and speed.
 - 6: broadcast a HELLO message to all one-hop neighboring vehicles.
 - 7: **if** i receives a hello messages from vehicle j **then**
 - 8: **if** j is moving in the same direction **then**
 - 9: add j to the one-hop neighboring list OH_i
 - 10: **else**
 - 11: do nothing
 - 12: **end if**
 - 13: **end if**
 - 14: **end for**
-

messages from all one-hop neighbors, each vehicle i will calculate its current weight W_i using (1). We have defined the election function W_i based on the one defined in [12]. The most stable vehicle that can act as a Cluster Head (CH) is a vehicle which has the minimum average distance to the other vehicles in the cluster, the closest speed to the average speed and the maximum number of neighboring vehicles

$$W_i = w_2 * D_i + w_1 |\nu_i - \Delta_i| - w_3 * N_i \quad (1)$$

Where $D_i = (\sum_{j \in OH_i} dist(j, i)) / N_i$ is the average distance between vehicle i and its one-hop neighbors vehicles OH_i , ν_i is the speed of vehicle i , $\Delta_i = (\sum_{j \in OH_i} \nu_j) / N_i$ is the average speed of the vehicles, and N_i is the number of one-hop neighboring of vehicle i . The corresponding weighing factors are such that $\sum_{i=1}^3 w_i = 3$. After that each node i will periodically broadcast an election beacon containing all of the necessary information for the CH election algorithm. The election beacon for vehicle i contains its: ID, CH-ID which indicates the ID of the CH to which the node is attached, Highway ID, direction, and current Weight. Node i then announces itself as CH by assigning its own ID to the ID field of the election beacon. When a vehicle i receives beacons, from its one-hop neighbors, it sorts its neighbor list OH_i according to the weights received in the beacons, and then it executes the cluster head election algorithm to change its status from CH to Cluster Member (CM), Cluster Gateway (CG) or remain CH.

The vehicle i that has the minimum value of W_i is elected as the CH. Then, all the vehicles that are within transmission range of the CH become CMs or CGs and are not allowed to participate in another cluster head election procedure. The CH election algorithm terminates once all the vehicles either

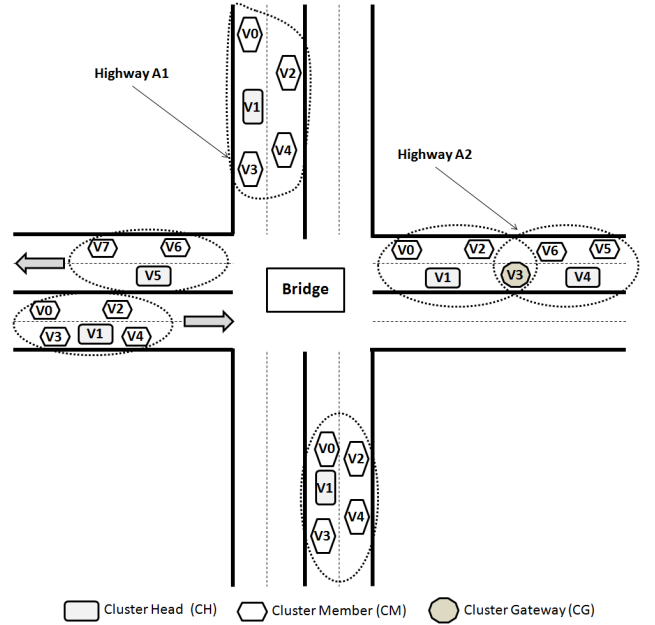


Fig. 2. example of 1-hop cluster formation on two highways

become a CH, CM or a CG. Algorithm 2 outlines the details of the election of CH nodes. It is executed by each vehicle i having at least one neighbor vehicle. In Algorithm 2, i , j , and x represent three vehicles which are moving in the same highway and in the same direction and participating in the CH election process. In addition *Cluster_Size* is the size of the cluster, *ITJ_Interval* is the time interval for a CH vehicle to broadcast the invite-to-join (ITJ) advertisement message, *PRE_Interval* is the time interval for CM to signal its presence to its CH, while *CH_Timeout_Interval* is the time interval for a vehicle to elect itself as a cluster head, if it did not receive nor broadcast messages nor an ITJ advertisement messages during this period.

C. Cluster maintenance

In VANETs, a vehicle can join or leave a cluster at any time. These two operations will have only local effects on the topology of the cluster if the vehicle is a CM. However, if the vehicle is the CH before leaving the cluster, it must hand over the responsibility to one of the very close cluster members. The first reason for that is to keep the cluster structure even if the current CH leaves. The second reason is to avoid using the re-clustering algorithm and thus no re-clustering overhead is generated when the CH leaves the cluster. Then, the current CH will order the CM to switch to CH and switch its own state to CM.

1) *Join a Cluster*: The cluster head periodically broadcasts invite-to-join ITJ message to their one-hop neighboring. Once an US or CHC vehicle receives the ITJ message and if it wishes to join the cluster, it will check the received signal strength Pr . The US or CHC vehicle will consider the ITJ message to be valid if its signal strength is bigger than the predefined threshold denoted by *Pr_Threshold*. When receiving a valid ITJ message, the vehicle sends a request-to-join (RTJ) message including the vehicles highway ID, position, speed and

Algorithm 2 Cluster head election

```
1: for every Election_Interval seconds each vehicle i do
2:   Calculate and update the election weight  $W_i$ 
3:   broadcast a beacon message to all one-hop neighboring
   vehicles.
4: end for
5: Upon reception of election beacons, each vehicle i will:
6: Sort its  $OH_i$  list
7:  $j \leftarrow$  head of  $OH_i$ 
8: if ( $i == j$ ) then
9:    $S_i \leftarrow$  CH
10:  for every ITJ_Interval seconds do
11:    Vehicle i broadcast ITJ message
12:    if i receives Request-to-Join (RTJ) from another
    vehicle x then
13:      if The number of CM vehicles < Cluster_Size
    then
14:        i will send an ACK message to x
15:      end if
16:    end if
17:  end for
18: else
19:   i sends RTJ message to j
20:  if i receives an ACK from j then
21:     $S_i \leftarrow$  CM
22:     $CH - ID \leftarrow j$ 
23:    for every PRE_Interval seconds do
24:      i Sends presence message to its CH j
25:    end for
26:  end if
27: end if
28: if i remains in US or CHC state more than Time-
    out_Interval seconds then
29:    $S_i \leftarrow$  CH
30: end if
```

direction. When the cluster head receives the RTJ message, it checks the direction of the requesting vehicle and if in the same highway and in the same direction, the CH sends an acknowledgment (ACK) including the ID number. After the reception of the ACK, the corresponding vehicle becomes CM of this cluster. Once US vehicle becomes CM, it is not allowed to participate in the other cluster head election procedure. If a CM receives an (ITJ) message from another neighboring CH moving in the same highway and in the same direction, the corresponding vehicle will switches from CM state to cluster gateway state (CG).

2) *Leaving a Cluster*: A vehicle remains in CM state as long as it receives an ITJ message from its CH every *ITJ_Interval*. When the CM vehicle cannot receive ITJ message from its CH during *CH_Timeout_Interval*, it considers that it loses contact with the CH and thus switches its state to CHC. Each CH updates a timestamp field for each CM based on the received PRE-MSG messages. The CH removes a CM from its cluster members list if the difference between the current time and the last time stamp of the PRE-MSG message received from it is greater than *CM_Timeout_Interval*. The CH will change its state to CHC, if its list of cluster members is empty.

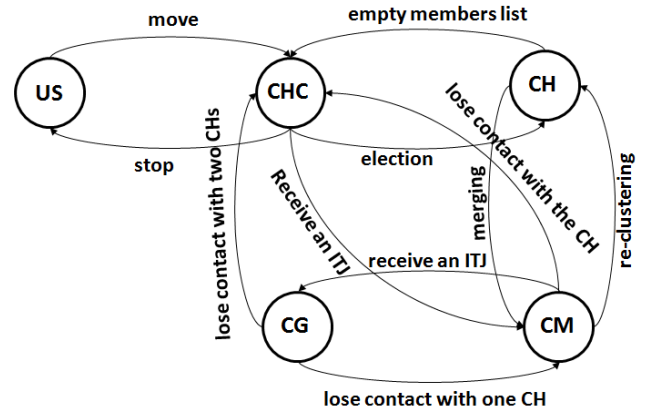


Fig. 3. State transition diagram

3) *Merging two or three clusters*: When two or three CHs moving in the same highway and in the same direction receive ITJ messages from each other with signal strength bigger than the predefined threshold $Pr_Threshold$, only one of them will keep its CH responsibility while the other will switch to CM. The CG between clusters becomes CM of the new cluster, and each CM which its CH become CM will switch to CM if it receives ITJ message from the new CH and will switch to CHC otherwise. The selection of cluster head between merging clusters is done based on the weight W_i . Figure 3 shows the state transition diagram of our clustering protocol AWCP, where each state represents the vehicles role in a given cluster, and the arrows represent the different events occurred on the road in which a vehicle switches from one state to another.

IV. AWCP PARAMETERS AND PERFORMANCE CRITERION

The performance of AWCP depends on the selection of the parameter settings that determine its behavior. For instance, the detection of topological changes can be adjusted by changing the *Hello_Interval* parameter. We have defined a solution vector of real variables that can be fine tuned by using an optimization technique with the aim of obtaining QoS efficient AWCA configurations. Table IV shows the parameters of AWCP and their variation ranges. These parameters are four timers, four counters and three weighing factors. The variation ranges of the four timers and the two first counters are set based on the clustering protocols proposed in the literature. The *Cluster_Size* is the maximum number of vehicles in the cluster which should be less than $(R * l) * 2 / (w + d)$, where R , l , w and d are respectively the transmission range, the number of lanes in the road, the standard length of vehicle which is about $3m$ and the safety distance. P_{min} is the received signal strength where the distance between two vehicles is equal to the safety distance, where P_{max} is the received signal strength where the distance between two vehicles is equals to $3 * R/4$.

A given AWCP configuration is evaluated based on the three of the most QoS metrics used in this area [4]: The *Average Cluster Lifetime (ACL)*, that is the average time period from the moment when a vehicle becomes a CH, CM or CG to the time when it is changed its state. The *Control*

Parameter	Type	Lower bound	Upper bound
<i>Hello_Interval</i>	\mathbb{R}	0.5	15
<i>Election_Interval</i>	\mathbb{R}	0.5	15
<i>ITJ_Interval</i>	\mathbb{R}	1	15
<i>PRE_Interval</i>	\mathbb{R}	1	15
<i>CH_Timeout_Interval</i>	\mathbb{R}	2	45
<i>CM_Timeout_Interval</i>	\mathbb{R}	3	45
<i>Cluster_Size</i>	\mathbb{Z}	1	$(R * l) * 2 / (w + d)$
<i>Pr_Threshold</i>	\mathbb{R}	P_{min}	P_{max}
<i>Distance_Weight_factor</i> (w_1)	\mathbb{R}	0	1
<i>Speed_Weight_factor</i> (w_2)	\mathbb{R}	0	$1 - w_2$
<i>Neig_Weight_factor</i> (w_3)	\mathbb{R}	0	$1 - (w_1 + w_2)$

TABLE I. AWCP PARAMETERS

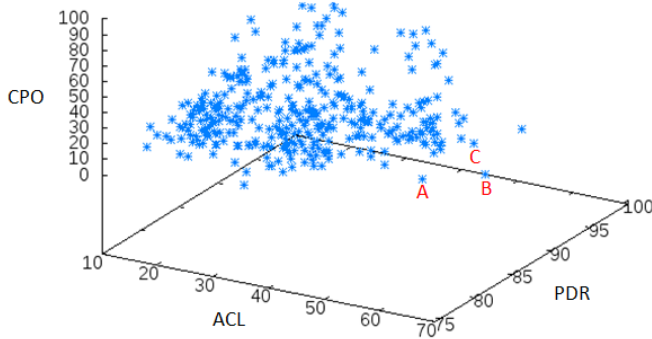


Fig. 4. Distribution of solutions on the objective space.

Packet Overhead (CPO), that is the rate of AWCP control packets used to form and maintain the cluster structures. And finally, the *Packet Delivery Ratio (PDR)*, that is the ratio of the number of the data packets that are correctly delivered to their destinations. Figure 4 shows the values of the three optimized objectives for different AWCP configurations. From the figure, it is clear that the performance of AWCP depends on the choice of the tuning parameters. Due to the conflicting nature of the objective functions and the large size of the search space, the AWCP parameters tuning is an NP-complete problem. Hence, we formulated the AWCP parameters tuning as a multi-objective problem and we have proposed an optimization tool which consists in combining a non-dominated sorting genetic algorithm, version II (NSGA-II) [13] and a network simulator NS2 to determine the optimal parameters of AWCP.

V. NSGA-II BASED APPROACH FOR AWCP OPTIMIZATION

A. Overview of NSGA-II

The optimization of a group of objective functions that are in conflict with each other is not a simple task. For simplicity, we assume that all objective functions are to be minimized. In fact, the multiplication of some objective functions by -1 allows to transform a maximization to minimization. Thus, the multi-objective optimization problem (MOP) can be formulated as follows:

$$(MOP) \begin{cases} \min & f_m(x), \quad m = 1, \dots, M \\ & s.t \\ & x_i^l \leq x_i \leq x_i^u, \quad i = 1, \dots, n \end{cases}$$

A solution $x^{(i)} \in S$ dominates the solution $x^{(j)} \in S$ if $Pf^* = \{x \in S \mid \nexists x' \in X, x \prec x'\}$

The vector $x = (x_1, \dots, x_n) \in S$ is the vector of n decision variables. The x_i^l and x_i^u are respectively the lower and upper bounds of the variable x_i . These bounds define the decision space D . Let a minimization MOP, a solution $x^{(i)} \in S$ dominates the solution $x^{(j)} \in S$ (it is denoted $x^{(i)} \prec x^{(j)}$) if the following conditions are satisfied:

$$i) f_m(x^{(i)}) \leq f_m(x^{(j)}) \quad \forall m \in \{1, \dots, M\}$$

$$ii) \exists m \in \{1, \dots, M\} \text{ such that } f_m(x^{(i)}) < f_m(x^{(j)})$$

The set of optimal solutions is composed by the non-dominated vectors, named Pareto optimal set. The set of Pareto optimal is often called the Pareto front, denoted $Pf^* = \{x \in S \mid \nexists x' \in X, x' \prec x\}$. In other words, the Pareto front is the set of compromise solutions. In Figure 4, the points A and B are two points on the Pareto front: A does not dominate B, B does not dominate A, but both dominate the point C. The goal of the multi-objective optimization is to find the Pareto front for a given problem.

Non dominated Sorting Genetic Algorithm, version 2 (NSGA-II) [13] is often used to solve the multi-objective optimization problem. This method is a multi-objective version of the genetic algorithm in which the explored solutions are classified into Pareto-optimal fronts. Several NSGA-II based approach have been proposed in the literature to solving MANET, WSN and VANET optimization problems. For instance, in [17], the authors have proposed a solution based on Multi-objective genetic algorithm for energy efficient QoS routing in cluster based WSN. Abdou et al. [18] applied different multi-objective optimization algorithms for finding an optimal parameters for the broadcasting methods in mobile ad hoc networks.

B. Proposed approach

The proposed approach is based on NSGA-optimization tool, a network simulator and ns2-trace analyzer (see figure 6). These three modules cooperate to determine the optimal AWCP configuration in different mobility scenarios. Firstly, the optimization tool generates a set of possible parameters that is transmitted to the network simulator. Thereafter, the simulations are run and trace file is built, this file is passed on to the third module (trace analyzer) that computes the values of the fitness functions. The calculated objective values are then transmitted to the optimization tool that evaluates and ranks the solutions according to these values. Then, the optimization tool runs its operations to regenerate another set of possible solutions. This process starts again, until the stop criteria is reached. In the next, we will describe in detail the NSGA-II based optimization tool. NSGA-II begins from an initial population (P) formed by solutions vectors "individuals". In each iteration, an auxiliary population Q is formed by applying the genetic operators (crossover and mutation). Then, both the current (P) and the new population (Q) are merged together to form one set of solutions R, which will be sorted according to the non-domination and crowded comparison (line 4.5). For more details, please see [13]. Finally, only the best individuals in R can access to the next generation and will participate in the production step while the others individuals are deleted (line 4.6 to 4.10). These steps are repeated until the maximum number of iteration is reached.

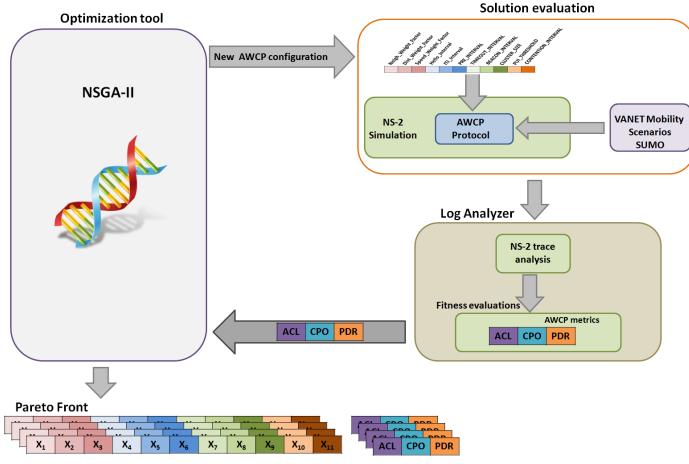


Fig. 5. NSGA-II based approach for AWCP optimization

Algorithm 3 NSGA-II algorithm for AWCP optimization

```

1:  $I_{tr} \leftarrow 0$ 
2:  $P_{I_{tr}} \leftarrow \{\emptyset\}$ 
3: Initialization of the initial population  $P_{I_{tr}}$  of size  $N$ 
4: while ( $I_{tr} < Nbr\_iteration\_max$ ) do
5:    $Q_{I_{tr}} \leftarrow \{\emptyset\}$ 
6:   while ( $t \leq popSize/2$ ) do
7:      $parents \leftarrow selection(P_{I_{tr}})$ 
8:      $Child \leftarrow crossover(P_c, parents)$ 
9:      $E \leftarrow mutation(P_m, Child)$ 
10:    compute\_objective\_values( $Child$ )
11:     $Q_{I_{tr}} \leftarrow Q_{I_{tr}} \cup \{Child\}$ 
12:  end while
13:   $R_{I_{tr}} \leftarrow P_{I_{tr}} \cup \{Q_{I_{tr}}\}$ 
14:   $R_{I_{tr}} = \bigcup_{i=1}^r F_i$  and  $F_1 < F_2 < \dots < F_r$ 
15:   $P_{I_{tr+1}} \leftarrow \{\emptyset\}; i \leftarrow 0$ 
16:  while ( $|P_{I_{tr+1}}| + |F_i| < N$ ) do
17:     $P_{I_{tr+1}} \leftarrow P_{I_{tr+1}} \cup F_i$ 
18:     $i \leftarrow i + 1$ 
19:  end while
20:  ranking( $F_i, crowding\_distance$ )
21:   $I_{tr} \leftarrow I_{tr} + 1$ 
22:   $P_{I_{tr}} \leftarrow P_{I_{tr}} \cup \{N - |P_{I_{tr}}| \text{ first solutions in } F_i\}$ 
23: end while

```

Each individual is encoded as a vector with nine genes. Each gene that encodes one AWCP parameter is defined by its type (real, integer), bounds and its precision p . The real-valued parameters are converted into integers by multiplying their real values by 10^p . The initial population is generated by randomly choosing the values of each gene in its variation range. Thereafter, it is used for the circulated genetic operators to create a new population.

The crossover operator is one of the main parts of NSGA. The input of the operator consists of two solution vectors (known as parents). The output is two children vectors, which has certain features from both parents. Because all genes in each solution vector in the population are in their given intervals, the resulting vector should satisfies the constraints formulated in the Section V. In order to enforce this constraint, we suggest the use of uniform crossover method. First, a crossover mask

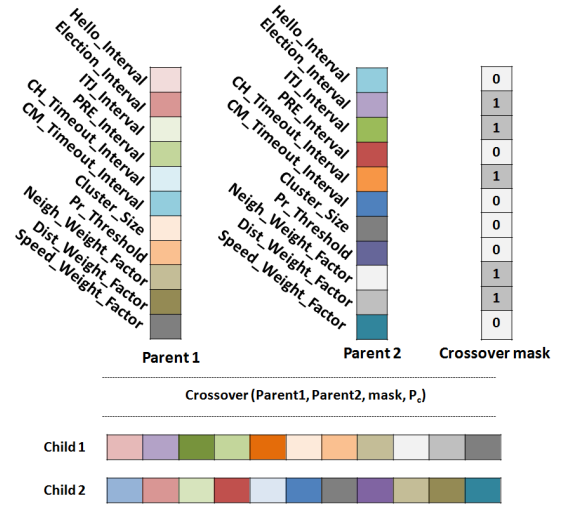


Fig. 6. Uniform crossover operator example

$X = (x_i) \in \{0, 1\}^{11}$ is randomly computed, which determines for each genes from which parent vector it inherits. In the following description, assigning gene i to the first parent means setting $x_i = 1$ and assigning gene i to the second parent means setting $x_i = 0$. After recombination, the mutation operator is applied to randomly change some genes in individual. During the mutation step, a gene is randomly chosen and then it generates a new value for this parameter with respect of its variation range. This operator serves as a strategy to prevent solutions from being trapped in local optima. The crossover and mutation operator permit to generate a list of solution vectors, objective value is again computed using ns simulations.

VI. SIMULATION RESULTS AND PERFORMANCE EVALUATION

We carried out a set of experiments to prove the ability of NSGA-II coupled with ns2 simulator to search quickly for optimal performances, as well as its capability for fine tuning the optimal values for the AWCP parameters. The optimization tool was implemented in Java language while the simulation phase is carried out by running ns-2.34. Moreover, all our experiments were conducted using a personal computer Intel Core Duo 3.2GHz PC with 2 Gb of memory and O.S. Linux Ubuntu 12.04. In order to achieve the best optimal behavior of AWCP protocol, several experiments on various VANET scenarios are necessary. In this section, we present the set of VANET scenarios used to obtain efficient QoS AWCP parameters and the experimental validation.

A. VANET scenarios

we generated realistic VANET environment by selecting real highway area which taking into account road directions, roads intersection, highway bridge, and traffic rules) from digital maps. Figure 10 shows a metropolitan area from the Map of Tunis of size $3 \text{ km} \times 3 \text{ km}$ exported from OpenStreetMap (OSM) and edited using Java OpenStreetMap Editor (JOSM). Then SUMO [19] and MOVE [20] are used respectively to generate vehicles traffic scenario and to simulated the area

with vehicular traffic. The MOVE generates realistic NS2



Fig. 7. VANET mobility scenario taken from the metropolitan area of Tunis

traffic traces, which will be then used in the NS2 simulations. All tests have been performed on different VANET scenarios taking into account different vehicles densities and data loads. Different scenarios are defined named Low (25 vehicles and 15 data sources), Medium (50 vehicles and 20 data sources), High (100 vehicles and 30 data sources) and Very High (200 vehicles and 50 data sources). The features of the VANET scenarios and the simulation parameters used in our experimentation are summarized respectively in Tables VI-A and VI-B.

Scenario	Number of vehicles	Number of CBR sources
<i>Low</i>	25	15
<i>Medium</i>	50	20
<i>High</i>	100	30
<i>Very High</i>	200	50

TABLE II. VANETs scenarios

B. NSGA-II Results Analysis

This section presents and analysis the results of applying NSGA-II for the AWCP tuning problem. For these results, the size of the initial population is 40 individuals, the number of generation is fixed to 40, the crossover probability is 0.9, whilst the mutation probability is fixed to 0.1. We perform 10 independent runs of the NSGA-II algorithm in which the candidate individuals are evaluated by involving the simulation over the medium scenario (100 vehicles with 30 data sources). The computational time for each run is 54419 seconds (about 15 hours) with a deviation of 0.8%. After the experimentation, we identify a set of Pareto optimal solutions of size τ by gathering all the non-dominated solutions found in the 10 independent runs. These solutions give different degrees of trade-offs between three QoS metrics. These solutions are bounded by a so-called ideal objective vector z^{ideal} which contain the optimal value for each separately objective.

$$(z_j^{ideal})_{1 \leq j \leq k} = Opt_{x \in X} f_j(x); j = 1, \dots, \tau$$

Table VI-B shows the solutions that give the best values for each AWCP QoS metric that are the maximum ACL (*max-ACL*), minimum NRL (*max-PDR*), and minimum CPO (*min-CPO*), and the average values of the obtained τ non-dominated solutions on the Pareto front. As shown in the table, in our case the ideal vector has three values that are the *max-ACL*, *max-PDR*, and *min-CPO* (99.02, 91.39, 3.82). Moreover, the Euclidean distance of each solution in the non-dominated set

Configuration	ACL	PDR	CPO	EUDT
<i>max-ACL</i>	94.06 s	89.05%	12.68%	9.16
<i>max-PDR</i>	79.71 s	91.39%	7.15%	14.73
<i>min-CPO</i>	45.81 s	87.46%	3.82%	48.41
<i>NSGA-II_avg</i>	72.75 s	86.92%	6.69%	21.97
<i>min-EUDT</i>	90.02 s	88.54%	6.72%	5.73

TABLE III. NSGA-II simulation results and three optimized configurations, and average values of the obtained non-dominated solutions on the Pareto front

to the ideal objective vector is calculated and the solution with smallest Euclidean distance is selected (*min-EUDT*).

We can note that the closet configuration to the ideal objective vector (*min-EUDT*) presents the best trade-offs between the three QoS metrics, since the *min-EUDT* configuration gives the best objective values for each QoS metric close to the best ones. The *max-ACL* configurations achieves a high cluster lifetime, it has a high packet delivery performance and the clusters are formed and maintained with an excessive overhead (12.68%). The configuration that optimizes the PDR metric, *max-PDR*, delivers an important amount of data packets. However, it decreases the performance of AWCP protocol in terms of ACL (79.71s). The configuration that crates clusters with less overhead *min-CPO*, produces a significant reduction in the performance of AWCP in terms of ACL (45.81%) and it delivers a low packet delivery ratio although the advantage of less control messages exchange.

Parameter	Value/Protocol
<i>Simulation area</i>	4000 × 4000 m ²
<i>Simulation time</i>	100 s
<i>Vehicle speed</i>	120 – 150 km/h
<i>Propagation model</i>	<i>Two Ray Ground</i>
<i>Medium Capacity</i>	6 Mbps
<i>PHY/MAC Layer</i>	<i>IEEE 802.11p</i>
<i>Transmission range</i>	1000 m
<i>Routing Layer</i>	<i>AODV</i>
<i>Transport Layer</i>	<i>UDP</i>
<i>CBR Packet Size</i>	512 bytes
<i>CBR Time</i>	60 s

TABLE IV. Simulation parameters in ns-2

C. NSGA-II Results Validation

In this section we present the results obtained by other multi-objective optimization approaches: Multi-Objective Differential Evolution (MODE) MODE [22], Multiple Objective Ant Colony Optimization (MOACO) MOACO [21], and Multi-Objective Particle Swarm Optimization MOPSO[23] which are the most recently used to optimize communication in ad hoc networks. Moreover, in order to verify the efficiency of our proposed approach, we carried out an experimental comparison with these techniques. For this, a set of simulations involving the 4 Highway scenarios presented in Section A are carried out. we have taken into account the *min-EUDT* AWCP parametrization in the validation experiments because it is the most balanced configuration of the Pareto front, and therefore, it presents the most competitive trade-off among the three QoS metrics. The *min-EUDT* configuration is *Hello_Interval*=0.78, *Election_Interval*=0.16, *ITJ_Interval*=7.23, *PRE_Interval*=9.16, *Pr_Threshold*= 7.23E-16, *CH_Timeout_Interval*=12.75, *CM_Timeout_Interval*=12.7,

Cluster_Size=50, $W_1 = 0.716$, $W_2 = 0.204$, and $W_3 = 0.07$.

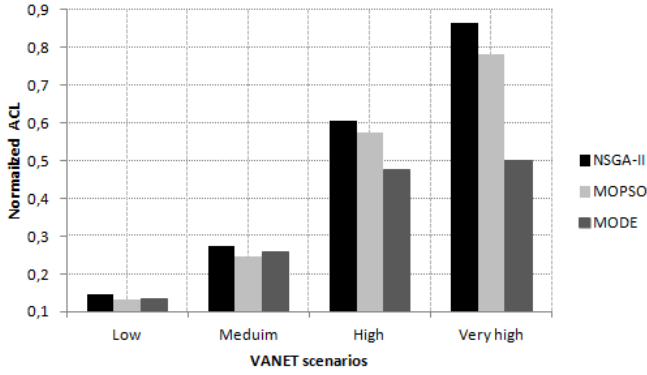


Fig. 8. ACL results under different VANET scenarios

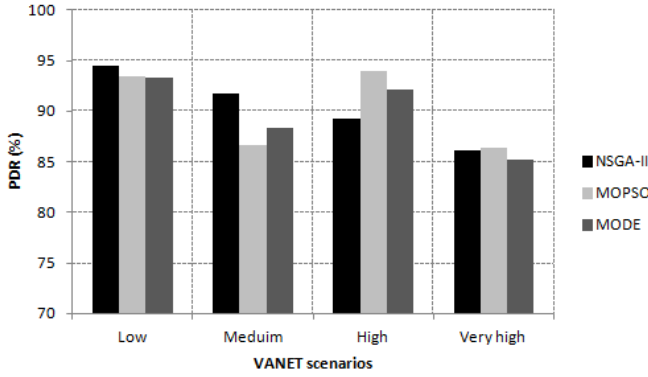


Fig. 9. PDR results under different VANET scenarios

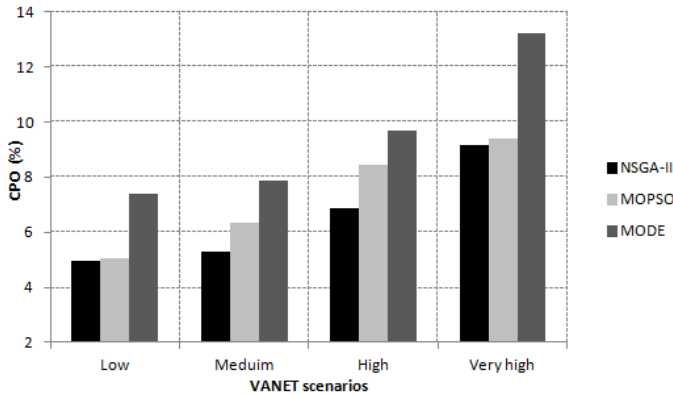


Fig. 10. CPO results under different VANET scenarios

In order to compare better the performance of different MOEAs, we evaluate the *Pareto fronts* (PF) obtained by the three approaches in terms of spacing, spread, generation distance, Ratio of non dominated solutions, and computational time.

- *The spacing metric* (S). It measures the distribution of solutions in the obtained PF set. It is proposed by Schott in [24] and defined as:

$$S = \sqrt{\frac{1}{\tau} \sum_{i=1}^{\tau} (d_i - \bar{d})^2}$$

$$d_i = \min_{x^{(j)} \in PF \wedge j \neq i} \sum_{k=1}^m |f_k(x^{(i)}) - f_k(x^{(j)})|$$

Where τ is the size of the Pareto front obtained, m is the number of objectives, and $\bar{d} = \sum_{k=1}^{\tau} d_i / \tau$ is the mean value of all d_j . A small value for this metric means that all non-dominated solutions in PF set are nearly spaced. Thus, the best multi-objective algorithm is the one that provides PF set with minimum spacing value.

- *The spread metric* (D). It determines the maximum range achieved among the obtained non-dominated solutions. A high value of the spread metrics means that the non-dominated solutions are widely distributed over the objective space. Thus, a higher value of D indicates a better algorithm performance. This metric is proposed by Ranjithan in [?] and defined as:

$$D = \sqrt{\sum_{k=1}^m (\max_{i=1}^{\tau} f_k(x^{(i)}) - \min_{i=1}^{\tau} f_k(x^{(i)}))^2}$$

$$x^{(i)} \in PF, j = 1, 2, \dots, \tau$$

- *Ratio of Non-dominated Individuals* (RNI). The performance measure determines the ratio of the number of the known solutions whose are chosen in Pareto front for a given population P . This metric is mathematically formulated as:

$$RNI = \frac{n}{|P|}$$

Where n is the number of non-dominated solutions in population P , and $|P|$ is the size of population. In the situation where $RNI = 1$, all individuals in the population are non-dominated. While $RNI = 0$ means that none of the known solutions in the population are non-dominated.

VII. CONCLUSION

Because of the rapid topology changing and the infrastructure absence, it is very difficult to efficiently cope with these challenges while deploying clustering methods in vehicular networks. In this paper, we present our work, that focusing on designing an adaptive clustering algorithm for VANET. We have proposed an adaptive weighted clustering protocol AWCP that takes into consideration the highway ID, direction, position, and speed information, in order to choose among existing vehicles, the most stable vehicles to act as cluster heads. With the availability of GPS and an in-depth digital street map, we are able to use Highway ID information to maximize the cluster structure stability. In this study, we also identified the parameters that are used to regulate AWCP. The size of the search space and the conflicting nature of objectives motivated the use of a multi-objective evolutionary algorithm coupled with NS2 simulator to solve the problem. We tested and evaluated our protocol on realistic VANET scenarios taken from the metropolitan area of Tunis (Tunisia). The simulation

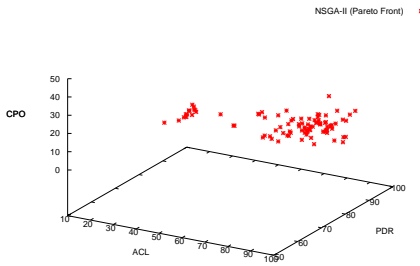


Fig. 11. Pareto front obtained by using NSGA-II approach

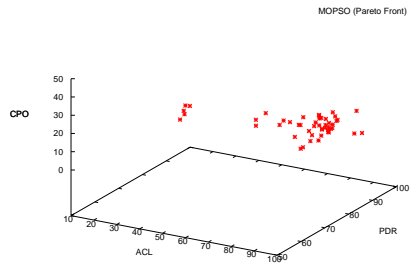


Fig. 12. Obtained Pareto front by using MOPSO approach

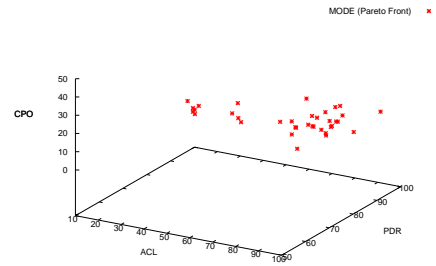


Fig. 13. Obtained Pareto front by using MODE approach

VANET Scenario	MOEA	Spacing		Spread		Generational Distance		Ratio of non-dominated		Avg. Time (s)
		Average	Std. Dev.	Average	Std. Dev.	Average	Std. Dev.	Average	Std. Dev.	
U1	NSGA-II									
	MOPSO									
	MODE									
U2	NSGA-II									
	MOPSO									
	MODE									
U3	NSGA-II									
	MOPSO	21.952747	12.682711	67.150199	12.523893	0.557143	0.752174	0.5571428	0.133911	79271.766
	MODE									
U4	NSGA-II									
	MOPSO									
	MODE									

TABLE V. COMPARISON WITH OTHER THREE MULTI-OBJECTIVE OPTIMIZATION APPROACHES OVER REALISTIC VANET

results show that the protocol proposed significantly improve the cluster lifetime and reduce the communication overhead while keeping or improving the performance in terms of packet delivery than WCA, Lowest-ID, Highest-Degree and the PCM algorithms.

Future works include the development of cross-layer architecture (MAC/clustering) including not only information about vehicles direction and speed but also channel quality. Moreover, the division of the channel access time into a Topology Management Frame (TBF) and a Data Transmission Frame (DTF) can improve the performance of AWCP protocol.

REFERENCES

- [1] Federal Communications Commission, FCC 99-305, FCC Report and Order, October 1999.
- [2] The FCC DSRC (Dedicated Short Range Communications), web site. <http://wireless.fcc.gov/services/its/dsrc/>.
- [3] O. Tonguz, N. Wisitpongphan, F. Bai, P. Mudalige, and V. Sadekar, *Broadcasting in VANET*, in INFOCOM MOVE Workshop 2007, Anchorage, Alaska, USA, May 2007.
- [4] Z. Y. Rawashdeh and S. M. Mahmud, *A novel algorithm to form stable clusters in vehicular ad hoc networks on highways*, EURASIP Journal on Wireless Communications and Networking, 2012.
- [5] T. Song, W. Xia, T. Song, and L. Shen, *A cluster-based directional routing protocol in vanet*, in IEEE International Conference on Communication Technology, 1172–1175, 2010.
- [6] M.S. Almalag, S. Olariu and M.C. Weigle, *TDMA cluster-based MAC for VANETs (TC-MAC)*, Proc. IEEE WoWMoM, pp. 1-6, 2012.
- [7] R. L. Devi, C. Maheswari and L. Maria, *A Cluster Based Authentic Vehicular Environment for Simple Highway Communication*, International Conference on Information and Network Technology, Singapore, 2012.
- [8] M. S. Almalag and M. C. Weigle, *Using Traffic Flow for Cluster Formation in Vehicular Ad-hoc Networks*, IEEE Workshop On User MObility and VEhicular Networks, Denver, 2010.
- [9] S. C. Lo, Y. J. Lin, and J. S. Gao, *A Multi-Head Clustering Algorithm in Vehicular Ad Hoc Networks*, International Journal of Computer Theory and Engineering, Vol. 5, No. 2, April 2013.
- [10] A. Daeinabi, A.G.P. Rahabar, A. Khademzadeh, *VWCA: an efficient clustering algorithm in vehicular ad hoc networks*, *Journal of Network and Computer Applications*, vol. 34, no. 1, 207222, 2011.
- [11] A. Ahizoune, A. Hafid: *A new stability based clustering algorithm (SBCA) for VANETs*, *LCN Workshops*, 843-847, 2012.
- [12] M. Chatterjee, S. K. Das, and D. Turgut, *WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks*, *Cluster Computing*, 193204, 2002.
- [13] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, *A fast and elitist multiobjective genetic algorithm: Nsga-ii*, IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182197, 2002.
- [14] M. Gerla and J. C. Tsai, *Multicluster; mobile, multimedia radio network*, ACM Wireless Networks, vol. 1, 1995.
- [15] A. Ramalingam, S. Subramani, and K. Perumalsamy. *Associativity based cluster formation and cluster management in ad hoc networks*. In Proceedings of the 9th International Conference On High Performance Computing, 2002.
- [16] W. Zhuang, H. A. Omar and L. Li, *VeMAC: A novel multichannel MAC protocol for vehicular ad hoc networks*, INFOCOM WKSHPs, 413-418, 2011.
- [17] G. H. EkbataniFard, R. Monsefi, M-R. Akbarzadeh-T and M. H. Yaghmaee, *A multi-objective genetic algorithm based approach for*

energy efficient QoS-routing in two-tiered Wireless Sensor Networks, TSWPC'10, 80-85, 2010.

- [18] W. Abdou, A. Henriët, C. Bloch, D. Dhoutaut, D. Charlet, F. Spies, *Using an evolutionary algorithm to optimize the broadcasting methods in mobile ad hoc networks*, Journal of Network and Computer Applications. 34, 1794–1804, 2011.
- [19] D. Krajzewicz, M. Bonert, and P. Wagner, *The open source traffic simulation package SUMO*, in RoboCup'06, Bremen, Germany, 2006, pp. 110.
- [20] F. Karnadi, Z.H. Mo, K. chan Lan, *Rapid generation of realistic mobility models for VANET*, in: Wireless Communications and Networking Conference, 2007, WCNC'2007, IEEE, 2007, pp. 2506-2511.
- [21] C. W. Wu, T. C. Chiang and L. C. Fu, *An ant colony optimization algorithm for multi-objective clustering in mobile ad hoc networks*, 2014 IEEE Congress on Evolutionary Computation (CEC), Beijing, China, July 2014, pp. 2963-2968.
- [22] H. Yetgin, K. T. K. Cheung and L. Hanzo, *Multi-objective routing optimization using evolutionary algorithms*, WCNC'2012, IEEE, Shanghai, April 2012, pp. 3030-3034.
- [23] H. Ali, W. Shahzad, F. Khan, *Energy-efcient clustering in mobile ad-hoc networks using multi-objective particle swarm optimization*, Applied Soft Computing 12 (2012), pp. 1913-1928.
- [24] J. R. Schott. *Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization*. PhD thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, Massachusetts, 1995.