International Journal Of Engineering Research & Management Technology
ISSN: 2348-4039
September- 2014 Volume 1, Issue-5
Email: editor@ijermt.org www.ijermt.org

Performance Evaluation of Vehicular Data Transfer Protocol Using Metaheuristic Techniques

Manish Suroliya
School of Engineering & Technology
Jaipur National University

V.S. Dhaka
School of Engineering & Technology
Jaipur National University

ABSTRACT—
Vehicular ad hoc networks (VANETs) are the emerging field, which deals with a set of communicating vehicles that are able to deploy Intelligent Transportation Systems (ITS). In this kind of networks, it is difficult to make an ideal configuration of the communication protocols. In this work, we consider a problem which lies in configuring the File Transfer Protocol configuration with the aim of optimizing the amount of data transferred, the transmission time and the number of lost packets in realistic VANET scenarios. In order to do this we have used metaheuristic algorithms, which lie in searching efficient parameters setting of VDTP protocol. The VDTP protocol has been tested by employing five different parameter settings: PSO, DE, GA, ES, and SA. For our tests, two different environment instances of VANETs for Urban and Highway scenarios have been taken. The experiments using ns-2 prove that PSO outperforms all the compared algorithms.

Keywords— VANETs, VDTP, Metaheuristics, optimized configurations, performance analysis

I. INTRODUCTION

Vehicular Ad hoc Networks (VANETs) [1] [2] are fluctuating networks composed of a set of communicating vehicles (nodes) equipped with contrivances which are able to spontaneously interconnect each other without any pre-subsisting infrastructure. This designates that no accommodation provider is present in such kind of networks as it is conventional in traditional or in mobile cellular communication networks. The most popular wireless networking technology available now days for establishing VANETs is the IEEE 802.11b WLAN, also known as Wi-Fi (wireless fidelity). Incipient standards such as the IEEE802.11p and Wi-Fi direct are promising but still not available to perform authentic tests with them. This imperatively insinuates that vehicles communicate with in a constrained range while moving, thus showing a topology that may change expeditiously and in capricious ways. In such kind of networks, antecedent to its deployment, it is critical to provide the user with an ideal configuration of the communication protocols in order to increment the efficacious data packet exchange, as well as to minimize the transmission time and the network use (with their implicative insinuations on higher bandwidth and lower energy consumption). This is especially true in certain VANET scenarios (as shown in Fig. 1) in which buildings and distances discontinue communication channels usually, and where the available time for connecting to vehicles could be just 1s. The efficient protocol configuration for VANETs without utilizing automatic perspicacious design implements is virtually infeasible because of the large number of possibilities. It is especially arduous (e.g. for a network designer) when considering multiple design issues, such as highly dynamic topologies and decreased coverage. In integration, the utilization of exact techniques is also impracticable due to the time spent during the great number of simulations required. All this motivates the utilization of met heuristic techniques [3] which arise as well-suited implements to solve this kind of quandaries.

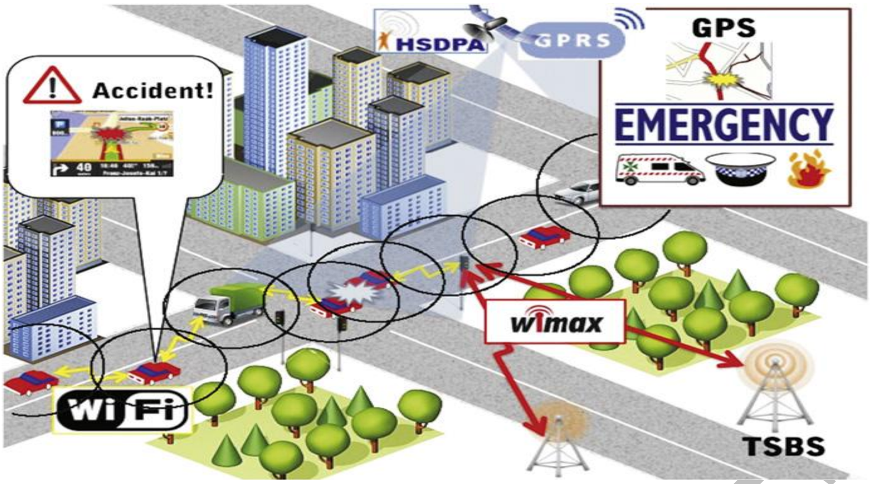


Fig. 1 Typical urban VANET scenario Circles represent the Wi-Fi coverage of vehicles.

In this paper, we face the File Transfer protocol Configuration in VANETs by betokens of five different optimization techniques. This problem lies in the core of any VANET application, and thus ideal configuration is main concern. Also, we utilize many optimization algorithms because this is an incipient field, and their relative advantages are still obscure. Indeed, we cannot find results for comparisons in the literature since only manual (human expert) VDTP configurations were made so far. These algorithms are two swarm astuteness techniques: Particle Swarm Optimization (PSO) [4] and Differential Evolution (DE) [5], two evolutionary algorithms: Genetic Algorithm (GA) [3] and Evolutionary Strategy (ES) [6] and a trajectory search technique, Simulated Annealing (SA) [7]. We have chosen these algorithms because they constitute a representative subset of well-kenned metaheuristics (population and trajectory predicated algorithms), with opportune operators for authentic parameter optimization, and with heterogeneous schemes of population and evolution. This way, we offer a set of initial results sanctioning future comparisons with other modern techniques.

For our experiment, two typical car-to-car environment instances have been taken: Urban and Highway VANETs. We rely both on a flexible simulation structure utilizing ns-2 (The Network Simulator Project—Ns-2) [8] [9], and authentic tests for optimizing the transmission time, the number of lost packets, and the volume of data transferred. One addition contribution of this work is to provide the specialist with an utilizable platform, embedded within ns-2, to configure network protocols and hence obtaining a fair QoS control in VANETs.

II. RELATED WORK

Few cognate works can be found in the specialized literature concerning the utilization of metaheuristics for the optimization of Mobile Ad hoc Networks (MANETs). Vanhatupa [10] proposed a flexible Genetic Algorithm for optimizing channel assignment in mesh wireless networks. In that work, the network capacity was increased by 20% while keeping the coverage above 80%. In Alba [11], a specialized Cellular Multi-Objective Genetic Algorithm (cMOGA) was utilized for finding an optimal broadcasting strategy in Urban MANETs, obtaining in this case three objectives fronts with coverage, bandwidth, and duration as performance metrics. The application of multi-objective techniques in this kind of works provides the specialists with a range of non-dominated solutions which can help them in the decision making process. Nevertheless, the use of (mono-objective) aggregated functions allows us the possibility of weighting the objectives and assigns more (or less) importance to them for better guiding the search. This way, in Dorronsoro [12], six versions of Gas (panmictic and decentralized) were evaluated and prosperously utilized in the design of ad hoc injection networks. From a different point of view, and due to its particular design, ant colony optimization (ACO) has been successfully adapted for implementing new routing protocols for MANETs [13], as well as for resource management [14]. Nevertheless, in these two last cases, the routing load incited by the internal operations of ACOs makes these approaches infeasible for astronomically immense networks. More recently, Huang [15] proposed an incipient routing protocol predicated on a PSO to make scheduling decisions for reducing the packet loss rate in a theoretical VANET scenario.

In our work, besides of utilizing the optimization technique itself as a protocol algorithm, our main contribution consists of ameliorating the performance of a subsisting protocol by optimally tuning its parameters. This way, we will optimistically obtain ideal configurations in the network design phase without incorporating extra management load to the authentic network operation.

III. PROBLEM OVERVIEW

The ideal File Transfer Configuration consists in optimizing the main parameters required by an application communication protocol. This protocol, called VDTP (vehicular data transfer protocol) [16], operates on the transport layer protocols of VANETs, sanctioning the end-to-end file transfer. This implies that considerations about the multi-hop interconnection mode and routing issues can be avoided, since they are carried out by the previous down layer protocols (e.g., UDP, DSR, IP, etc.). Hence, the different vehicles that constitute the nodes in a given VANET can exchange consummate files of information to each other by utilizing VDTP. In this section, we briefly describe the VDTP, detailing the main parameters to be optimized.

A A Brief Overview of Vehicular Data Transfer Protocol

VDTP is an application layer protocol that allows the end-to-end file transfer to be used in VANETs. It operates on Dynamic Source Routing protocol [17]. In VDTP, the communication process is carried out by both a file petitioner, which endeavours to download a file, and a file owner, which stores the file. This communication protocol works by utilizing these packets: FIRQ (File Information Request), FIRP (File Information Reply), DRQ (Data Request), and DRP (data reply). As presented in Fig. 2(a), once the file petitioner kens the designation and the location of a given file, it commences the communication by utilizing the FIRQ packet in order to obtain the file size. Then, the petitioner waits for this information which is sent by the owner by denotes of a FIRP packet. After receiving the information about the file size, the petitioner computes the number of segments in which the file will be split, dividing the file size by the chunk size. The petitioner commences the transfer by sending a DRQ (1) packet asking for the first segment of the file; then it waits for the first data chunk sent by the owner which utilizes the DRP (1) packet. This operation is reiterated by both, petitioner and owner, until transferring the last chunk DRP (n), and hence making up the consummate file. In VANETs, it is customary to work in a truculent medium which can incite a high number of lost packets during the communication process. In such situation, Vehicular Data Transfer Protocol uses various procedures on the basis of timers and counters to solve such issues.

The timeout mechanism controls the waiting time until a concrete DRQ or FIRQ packet has to be resent (retransmission time). Fig. 2(b) shows an example of how the DRQ and the DRP packets are disoriented (and retransmitted) after an established timeout. The counters are used to control the number of times a packet has been retransmitted. As shown in Fig. 2(c), after an antecedent designated number of retransmissions (total attempts) of the same DRQ/FIRQ packets, the transmission between the vehicles is refused.

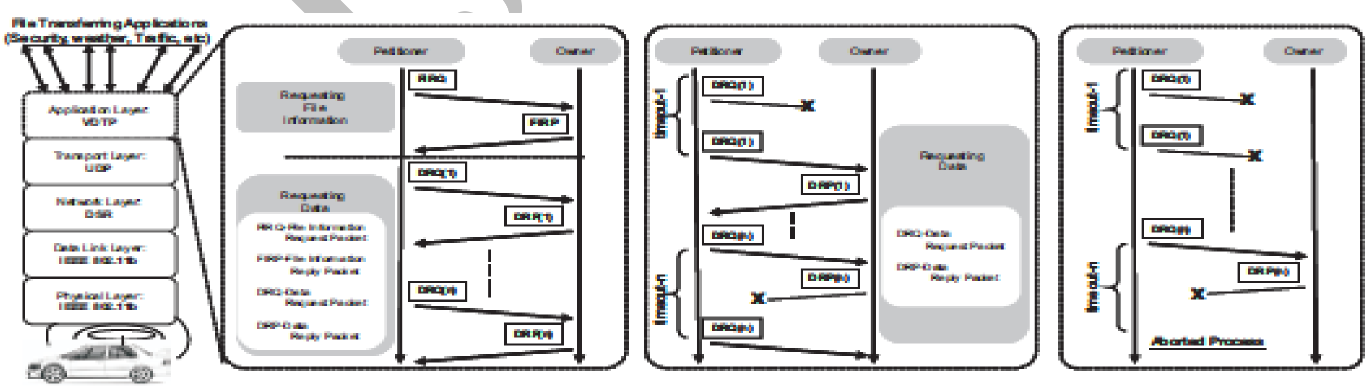


Fig. 2 VDTP operation modes: (a) a complete file exchange is done; (b) timeout expiration and retransmission; (c) communication refused. urban VANET scenario.

Problem design variables

Since we are fascinated with finding the best possible configuration of VDTP, we have fixated on the three aforementioned parameters: chunk size, retransmission time and number of total attempts. Therefore, a given configuration (representing a solution of the problem) is a vector of three authentic values (chunk size, total attempts and retransmission time). The scope of each parameter is:

- chunk size: R+ ϵ [128…524,288] bytes (524,288 bytes = 512 k Bytes),
- total attempts: R+ ϵ [1….250] attempts,
- retransmission time: R+ ϵ [1…. 10] s.

IV. THE ALGORITHMS

Major In this section we describe a brief overview of the five metaheuristic algorithms used in our study. Concretely, they are Particle Swarm Optimization, Evolutionary Strategy, Differential Evolution, Genetic Algorithm, and Simulated Annealing. These techniques were selected with the aim of experimenting with different population structures, as well as different reproduction mechanisms. We have verbally expressed the same stop condition (reaching a certain number of generations) in all algorithms in order to simplify the following descriptions.

B **Particle Swarm Optimization (PSO)**

Particle Swarm Optimization [4] is a population based metaheuristic inspired in the social behavior of birds with in a flock, and initially designed for continuous optimization problems. In PSO, each potential solution to the problem is called particle and the population of particles is called swarm. SO, each potential solution to the quandary is called particle and the population of particles is known as swarm. Each particle position x^i updated each generation g by betokens of the following equation:

$$x^i_{g+1} \leftarrow x^i_g + v \qquad (1)$$

Where factor x^i_{g+1} is the velocity of the particle and is given by

$$v^i_{g+1} \leftarrow \omega \cdot v^i_g\, \varphi_1 \cdot (p^i_g - x^i_g) + \varphi_2 \cdot (b_g - x^i_g) \qquad (2)$$

In this formula, p^i_g is the best solution that the particle i has stored so far, b_g is the best particle (also known as the leader) that the entire swarm has created, and ω is the inertia weight of the particle (it controls the trade-off between global and local experience). Finally, φ_1(cognitive component) and φ_2(social component) are specific parameters which control the relative effect of the personal and global best particles ($\varphi_1 = \varphi_2$ = 2· UN(0, 1)).

Algorithm1 describes the pseudo-code of PSO. The algorithm commences by initializing the swarm (Line1), which includes both the positions and velocities of the particles. The corresponding pi of each particle is randomly initialized, as well as the leader g (Line 2).Then, during a maximum number of iterations, each particle moves through the search space updating its velocity and position (Lines5and 6), it is then evaluated (Line7), and its pi is also calculated (Lines8). At the end of iteration, the leader b is updated.

Algorithm1. Pseudo code of PSO.

1: initialize Swarm ()
2: locate Leader (b)
3: **while** g < max Generations **do**
4: **for** each particle x^i_g **do**
5: update Velocity (v^i_g) //Equation 2
6: update Position (x^i_g) //Equation1

7: evaluate (x_g^i)
8: update (p_g^i)
9: **end for**
10: update Leader (b_g)
11: **end while**

C Differential Evolution (DE)

Differential Evolution [5] is a stochastic population based algorithm designed to solve optimization problems in continuous domains. The population resides of a set of individuals which evolve simultaneously through the search space of the problem. The task of producing new individuals is performed by differential operators such as the differential mutation and crossover. A mutant individual w_{g+1}^i is generated by the following equation:

$$w_{g+1}^i \leftarrow v_g^{r1} + \mu \cdot (v_g^{r2} - v_g^{r3}) \qquad (3)$$

Where r1, r2, r3 ϵ {1, 2, . . . , i − 1, i + 1, . . . ,N} are random integers mutually dissimilar, and also different from the index i, the mutation constant μ > 0 stands for the amplification of the difference between the individuals v_g^{r2} and v_g^{r3}, and it avoids the stagnation of the search process.

In order to increase even more the diversity in the population, each mutated individual bears a crossover operation with the target individual v_g^i, by means of which a trial individual u_{g+1}^i is produced. A randomly chosen position is taken from the mutant individual to prevent that the trial individual replicates the target individual.

$$u_{g+1}^i(j) \leftarrow \begin{cases} w_{g+1}^i(j) & if\ r(j) \leq Cr\ or\ j = j_r, \\ v_{g\,(j)}^i & otherwise. \end{cases} \qquad (4)$$

As shown in Eq. (4), the crossover operator randomly chooses a uniformly distributed integer value j_r and a random real number r ϵ (0, 1), also evenly distributed for each component j of the trial individual u_{g+1}^i. Then, the crossover probability Cr and r are compared just like j and j_r. If r is less than or equal than Cr (or j is equal to j_r) then we select the jth element of the mutant individual to be allocated in the jth element of the trial individual u_{g+1}^i. Otherwise, the jth element of the target individual v_g^i becomes the jth element of the trial individual. Finally, a selection operator chooses the acceptance of the trial individual for the next generation if and only if it yields a reduction in the value of the evaluation function (also called fitness function f()), as shown by the next Equation:

$$v_{g+1}^i \leftarrow \begin{cases} u_{g+1}^i & if\ f(u_{g+1}^i) \leq f(v_g^i), \\ v_{g\,(j)}^i & otherwise \end{cases} \qquad (5)$$

Algorithm 2 shows the pseudocode of DE. After initializing the population (Line1), the individuals evolve during a number of generations (maxGenerations). Each individual is then mutated (Line 5) and recombined (Line6).The new individual is selected (or not) following the operation of Eq. (5) (Lines 7and 8).

Algorithm2. Pseudo code of DE.
1: initializePopulation()
2: **while** g < maxGenerations **do**
3: **for** each individual v_g^i **do**
4: choose mutually different (r1, r2, r3)
5: w_{g+1}^i ← mutation (v_g^{r1}, v_g^{r2}, v_g^{r3}, μ)
6: u_{g+1}^i← crossover (v_g^i ,w_{g+1}^i, cp)

evaluate (u^i_{g+1})
$v^i_{g+1} \leftarrow$ selection (v^i_g, u^i_{g+1})
end for
end while

D Genetic Algorithm (GA)

Genetic Algorithms [3] are the most popular metaheuristic algorithms. A GA iterates a process in which two parents are selected from the whole population with a given Selection criterion, they are then recombined, the obtained offsprings are mutated, and finally they are evaluated and inserted back into the population following a given criterion. The mutation process is carried out by randomly (uniformly) selecting one of the elements in the solution, and assigning (randomly) a new value in the range as stated in Section 3.1.1. As recombination operator we use here a polynomial crossover defined for continuous variables [3]. Algorithm 3 summarizes the operations of a canonical GA.

Algorithm3. Pseudo code of GA.
$P_0 \leftarrow$ initializePopulation()
while g < maxGenerations **do**
p'_g recombine(P_g)
p''_g mutate(p'_g)
evaluate(p''_g)
p_{g+1} select(p''_g U p'_g)
end while

There are two main versions of GA: steady state GA (ssGA) and generational GA (genGA). The difference between the ssGA and the genGA is the way in which the population is being up dated with the new individuals generated during the evolution. In the case of the ssGA ,new individuals are directly inserted into the current population while in the case of the genGA, a new auxiliary population is built with the obtained offsprings and then, once this auxiliary population is full, it entirely replaces the current population. Thus, in ssGAs the population is a synchronously being updated with the newly generated individuals, while in the case of genGAs all the new individuals are updated simultaneously, in a synchronous way.

E Evolutionary Strategy (ES)

Evolutionary Strategy [6] is a metaheuristic algorithm, designed by Rechenberg and Schwefel, also based on the ideas of adaptation and evolution. As common with evolutionary algorithms, the mutation and selection operators are applied to the individuals through a given number of generations. The selection in evolutionary strategies is deterministic and only based on the fitness rankings, not on actual fitness values. We used here a mutation operator as explained in GA.

Algorithm4. Pseudo code of ES.
$c_0 \leftarrow$ initializeParent()
while g < maxGenerations **do**
$o_g \leftarrow$ mutate(c_g)
evaluate (o_g)
if f(o_g) is better than f(c_g) **then**
$c_g \leftarrow o_g$
end if
end while

The canonical ES (Algorithm 4) operates on a population of size two: the current individual (parent c) and the result of its mutation (offspring o). After the parent initialization (Line1), ES starts the evolution process by generating a mutated offspring (Line 3) which is evaluated (Line 4). Only if the offspring has a better fitness than the parent, it takes place the parent of the next generation (Lines 5 and 6). Otherwise the offspring is ignored. This is version of ES is called (1+1)-ES. More generally, in (1+λ) - ES, a population with more than one offsprings (λ) can be generated for being compared with the same parent. In a(1, λ)-ES the best offspring becomes the parent of the next generation while the current parent is always ignored. The most generalized version, (μ +/, λ) - ES, often uses a population of parents (μ) and also recombination as an additional operator.

F **Simulated Annealing (SA)**

SA was first presented as a trajectory based optimization technique in [18]. It is inspired in the metallurgy processes of annealing, and basically lies in a local search method with a mechanism that eventually promote solutions of worse quality than the current ones (uphill moves), in order to escape from local minima. The probability of performing such a movement decreases during the search process. The pseudocode of the canonical SA is showed in Algorithm 5.

Algorithm5. Pseudo code of SA.

1: initialize(T, S)
2: evaluate(S)
3: **while** g < maxGenerations **do**
4: **while** not coolingCondition(g) **do**
5: S' ← chooseNeighbor(S)
6: evaluate(S')
7: if accept(S, S', T) then
8: $S_{\square} \leftarrow S'$
9: **end if**
10: **end while**
11: coolDown(T)
12: **end while**

The algorithm works iteratively keeping a single tentative solution S at any time. In every iteration, a new solution S' is generated from the previous one, S (Line 5), and either replaces it or not depending on an acceptance criterion (Lines7–8).The acceptance criterion works as follows: both the old(S) and the new (S') solutions have an associated quality value, determined by a fitness function (f()). If it is worse, it replaces it with probability prob (Eq. (6)). This probability depends on the difference between their quality values and control parameter T named temperature. This acceptance criterion provides the way of escaping from local optima.

$$prob = \frac{2}{1 + e^{(f(S_o) - f(S_n))/T}} \qquad (6)$$

As iterations go on, the value of the temperature(T) is reduced following a cooling schedule(Line11), thus biasing SA towards accepting only better solutions. In this work, we employ the geometric rule T(n+1) = α, where 0 < α < 1, and the cooling is performed every k iterations (k is the Markov chain length).
For then neighbor selection, we use a mutation operator (as in GA and ES). The initial value of temperature T is automatically generated in such a way that any movement from the initial (random) solution will be accepted with a certain probability.

V. OPTIMIZATION STRATEGY

Our optimization strategy for this problem is composed of rudiment ally two main components: an optimization algorithm and a simulation process. The optimization part is carried out by (independently) one of the algorithms described in Section 4. All of them are specially adapted to find ideal (or cuasi- ptimal) solutions in continuous search spaces (which is the case in this work). The simulation activity is a way of allocating a quantifiable quality value to the factors regulating VDTP, thus leading to ideal configurations of this protocol customize to a given scenario. This procedure is carried out by betokens of the ns-2 simulator in which we have implemented the VDTP protocol for sending files in VANETs.

In each optimization algorithm, the evaluation of each solution is carried out by denoting of the simulation component. As Fig. 3 illustrates, when a given algorithm engenders an incipient solution it is immediately utilized for configuring the VDTP. This configuration evaluates the quality of the solution by utilizing the received retransmission time, chunk size, and total number of endeavors, as expounded in Section3.1. Then, ns-2 is commenced and maps a given VANET scenario instance, taking its time in evaluating the scenario with buildings, signal loss, obstacles, vehicles, velocity, covered area, etc., under the circumstances defined by the three control parameters optimized by the algorithm. After the simulation, ns-2 returns the global information about the transmission time required for sending the file, the number of lost packets engendered during the simulation, and the amount of data transferred between vehicles. This information is utilized to compute the fitness function.

A **Fitness function**

Since ns-2 operates by simulating (and averaging) many potential variations scenario all fitting the genuine vehicle system, there is a possibility of obtaining different fitness values even utilizing the same VDTP configuration (solution). Hence, in order to provide each solution with a fitness value as sure as possible, a single estimation of one solution requires N=10 internal simulations, computing the universal fitness (F) as the mean of all ns-2 results:

$$F = \frac{1}{N}\sum_{i=1}^{N}\frac{transmission\_time_i + lost\_packets_i}{\log(data\_transferred_i + C)} \qquad (7)$$

In this equation, I ϵ [1…. 10] is the number of simulations per solution evaluation. The factor C=2 avoids division zero if there is no data transference, preventing a possible error in the fitness calculation. The data transferred is presented in logarithmic scale in order to make up for the difference in the range of values. This way, the algorithm looks for minimizing the global fitness.

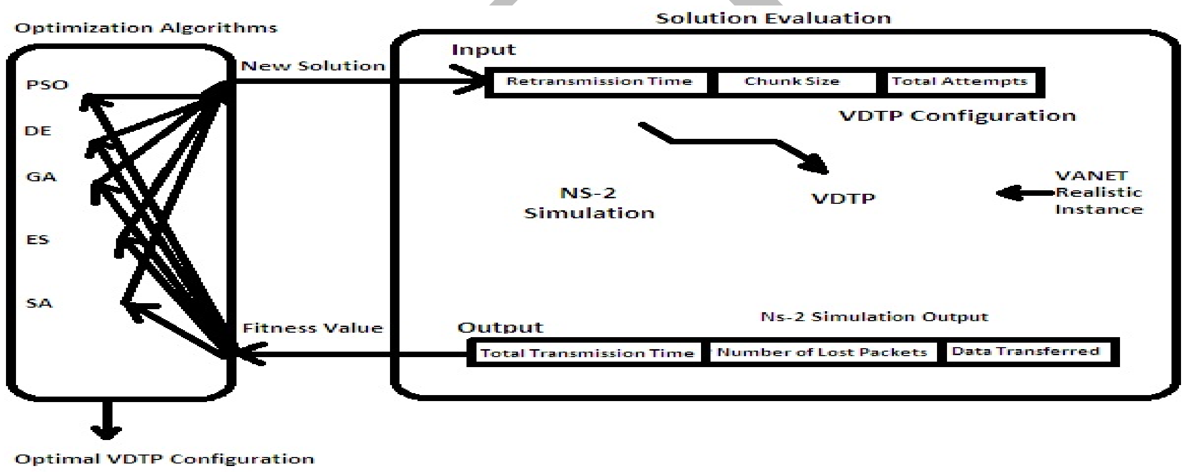


Fig. 3: Optimization strategy for VDTP configuration in VANETs. The algorithms invoke the ns-2 simulator each solution Evaluation

VI. EXPERIMENTS

We have utilized the implementation of the five algorithms provided by MALLBA [19], a C++ predicated framework of metaheuristics for solving optimization problems. The simulation phase is carried out by running ns-2 simulator v-2.31.For the experiments; we made 30 independent runs of each algorithm on machines with intel core i3 2.2GHz core, 4 GB of RAM and O.S Linux Ubuntu 12.04.

A **Instances: VANET scenarios**

We have engendered two simulations VANET scenarios (instances) from authentic Urban and Highway areas of Jaipur, India (selected areas in Fig. 4). These instances have been engendered following the authentic tests carried out by experts with the aim of obtaining as different as possible conditions of speed, quantity of vehicles, obstacles, signal noise, network use, etc. Hence, we can analyze in both scenarios the comportment and performance of the compared algorithms, as well as the differences in the resulting VDTP configurations in terms of communication efficiency.

Furthermore, we can compare these automatically engendered configurations against the ones utilized in the authentic experiments by human experts in [20] [21].

I Urban

The Urban scenario covers an area: Approximately 120,000 m2. It includes no buildings and semaphores. We have utilized VanetMobiSim [22] for engendering an authentic simulation mobility model where vehicles move desultorily according to authentic traffic rules. Around 30 vehicles move with a speed between 30 and 50 km/h, and 20 of them endeavouring to send and receive a file of 1024kBytes.

II Highway

The Highway scenario covers a distance of 1km with two directions. In this case, the absence of obstacles is compensated for the handicap of the high speed of vehicles, which additionally interferes the communication among vehicles. Engendering an authentic simulation mobility model where vehicles move desultorily according to authentic traffic rules. In the Highway VANET, an amount of 30 vehicles move with a speed between 80 and 110 km/h, and 20 of them endeavouring to send and receive a file of 1024kBytes size. The resulted communication environments of Urban and Highway instances, including directions and mobile nodes (vehicles), were mapped in the ns-2 simulator following the VANET designations of contrivances and protocols summarized in Table 1.

TABLE1: VANET instance specification

Parameter	Value
Propagation model	Two ray ground
Carrier frequency	2.472 GHz
Channel bandwidth	5.5 Mbps
Wifi channel	13
Link layer: transceiver	PROXIM ORiNOCO PCMCIA (IEEE 802.11b)
Link layer : antenna gain	7 dBi (Omnidirectional)
Mac protocol	802.11-b
Routing protocol	DSR
Transport protocol	UDP
Application protocol	VDTP
File transfers	20 sessions

. TABLE2: Parameterization of the optimization algorithms.

Algorithm	**Parameter**	**Symbol**	**Value**
PSO	Local coefficient Social coefficient Inertia weigh	φ_1 φ_2 w	2.rand(0.1) 2.rand(0.1) 0.5
DE	Crossover probability Mutation factor	Cr µ	0.9 0.1
GA	Crossover probability Mutation probability	Pcros Pmut	0.8 0.2
ES	Crossover probability Mutation probability	Pcros Pmut	0.9 0.1
SA	Temperature decay	T	0.8

B **Parameter settings**

In our experiments, all studied algorithms were configured in order to perform 1000 solution evaluations per run. At each one of these solution evaluations, ns-2 performs 10 independent simulations of the target scenario with the same protocol configuration as stated in Section5.1.Therefore, the population based algorithms (PSO, DE, GA, and (µ, λ) - ES) were configured with 20 individuals, accomplishing 50 generational steps.

Table 2 summarizes the remaining parameters categorical to each algorithm. These parameters were culled as the most precise after a set of initial tuning experiments. In these, a number of five coalescences of parameters per algorithm and VANET instance were tested performing 10 independent runs per amalgamation, hence results a number of 500 supplemental executions.

C **Result and comparisons**

In this section we compare the five studied algorithms when solving the ideal File Transfer Configuration quandary on VDTP. The results of fitness values regarding the Urban and Highway VANET scenarios in terms of the mean, the standard deviation, the minimum (best fitness), the median, and the maximum (worst fitness) found in 30 independent runs of every algorithm can be seen in Table3.

TABLE3: Final fitness values regarding the Urban and Highway VANET scenarios.

Instance	**Algorithm**	**Mean ± Std. dev.**	**Minimum**	**Median**	**Maximum**
Urban	PSO DE GA ES SA	**1.6346 ± 0.2899** 1.7423 ± 0.3717 1.9086 ± 0.2260 2.1517 ± 0.1266	0.9077 **0.7389** 0.8799 1.8862 0.8730	**1.7809** 1.8658 1.9731 2.1222 2.1663	**1.8918** 2.0228 2.1614 2.4246 3.8025

Instance	**Algorithm**	**Mean ± Std. dev.**	**Minimum**	**Median**	**Maximum**
		2.7850 ± 0.8718			
Highway	PSO	**4.1761± 0.2556**	3.3301	4.2513	**4.4554**
	DE	4.6631 ± 0.9328	2.7145	4.2272	7.0531
	GA	4.3805 ± 0.8695	**2.5345**	4.1918	5.8608
	ES	5.7833 ± 0.9705	3.8836	6.1347	6.9421
	SA	4.4246 ± 0.7401	3.1498	**4.0855**	5.7922

For the Urban scenario, we can detect (in Table 3) that PSO getting the best result with regard to the mean fitness. This most diminutive mean value leads us to believe that utilizing the PSO the resulting VDTP ends in an efficient communication which is expeditious and precise between vehicles. In advisement, the best median and maximum values were additionally obtained by PSO, albeit the best minimum (e.g., the best VDTP configuration found for Urban) was reached by DE. This is an expected value, since DE generally shows a pronounced exploitative comportment (utilizing a parameterization proximate to the standard one) [5], while PSO inclines to have an explorative performance utilizing a high inertia (as in this study w=0.5) [23]. Homogeneous results can be observed for the Highway scenario, in which PSO obtained the best mean fitness value again. For this instance, PSO additionally showed the lowest value of standard deviation. This implicatively insinuates a considerable advantage, since it provides our model with a high robustness, which is a crucial issue when designing VANETs. From the results, we found that GA and DE obtained the best VDTP organizations for the Highway instance. The worst organization was obtained by ES.

TABLE4: PSO versus other algorithms Signed Rank test with confidence level 95% (p-value=0.05).

	Urban		**Highway**	
Algorithm	Test	p-value	Test	p-value
DE	▲	0.047	▲	0.001
GA	▲	0.001	Δ	0.453
ES	▲	0.001	▲	0.001
SA	▲	0.001	Δ	0.371

TABLE5: Friedman rank test with confidence level 95%.

Urban			**Highway**	
Algorithm	Rank		Algorithm	Rank
PSO	**1.27**		SA	**1.83**
DE	1.83		GA	1.97
GA	3.07		PSO	2.17
ES	4.33		DE	3.67
SA	4.50		ES	4.97

In order to provide such comparison with statistical meaning, we have applied a Signed Rank [24] statistical test to the distributions of the aforementioned results. We have used this non-parametric test with confidence level of 95% (p-value=0.05), which leads us to ensure that these results are statistically different if they result in p-value < 0:05. Table 4 contains the resulted p-value of applying the Signed Rank test to PSO (the one with the best mean fitness) in comparison with the remaining of algorithms, hence substantiating the differences in results. In this table, the symbol ▲ denotes that PSO is statistically more preponderant than the compared algorithm, whereas the symbol Δ betokens that PSO has a more preponderant rank than the compared algorithm, but without statistical difference.

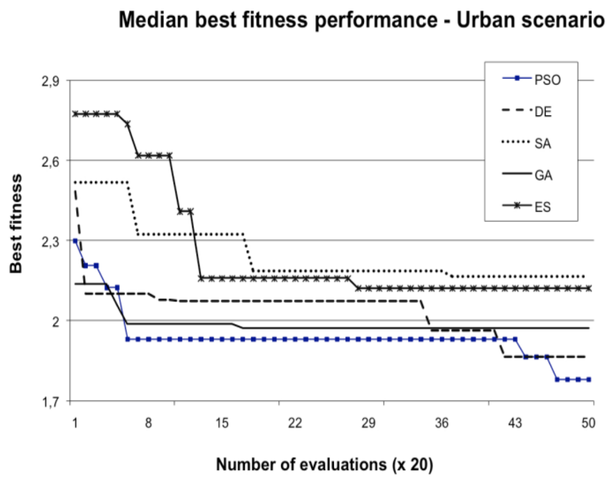
As we can observe in Table 4, PSO is statistically more preponderant than all compared algorithms for the urban instance. Only DE shows a p-value (0.047) proximate to 0.05, being lower in any case. Concerning the Highway instance, PSO presents the best rank, not far from GA and SA. A general comparison can be made utilizing the Friedman [25] statistical test by betokens of which the algorithms are sorted in a ranked list. Table 5 exhibits the Friedman ranking of the compared algorithms in Urban and Highway scenarios (the best ranked algorithm is in the top). For Urban scenarios, PSO and DE are the best ranked algorithms, but exhibiting SA the last position. However, for Highway scenario, SA obtains the best rank, whereas PSO is situated in the third position. These statistical results lead us to cerebrate that, in spite of the global best deportment of PSO, the different requisites implicit to both instances implicatively insinuates that each algorithm can show quite different results depending on the VANET scenario on which it operates. For example, DE exhibits a moderate performance in urban scenario while it is the second worst in Highway. The antithesis example can be observed in GA and SA which show impotent results in Urban but highly competitive ones in Highway. Therefore, the VANET designer can cull the optimization model more suited to his/her requisites, and opt ate the best option for each studied VANET scenario.

D Performance Analysis

We present now a performance study which fundamentally lies in analysing the best fitness value, resulted from each function evaluation, during the whole evolution process of a given algorithm Figs. 5 and 6 demonstrate the graphs of the best fitness values (communicate on cost) acquired through the median execution in Urban and Highway instances, properly.

We can observe in both graphics that PSO and DE tend to converge in the same range of solution evaluations, although they could improve their fitness even in the final steps of the evolution process. GA shows a kindred trend as the former ones but it is subjected to an early slowness.

Conclusively, the different deportments observed in ES, and categorically in SA, for Urban and Highway instances substantiate us the high dependency of such algorithms to each different VANET instance (they are not robust in this application). Regarding the mean run time that each algorithm spent in the experimentation, Table 6 shows both the mean time in which the best solution was found T_{best}, and the global mean run time T_{run} for Urban and Highway scenarios. In general, SA shows the shortest times to find the best solution for the two VANET instances. We suspect that despite its temperature mechanism, SA expeditiously falls in local optima hence obtaining impotent results in urban scenario. Nevertheless, this demeanor can be an advantage for Highway scenario where SA obtained precise solutions with an expeditious performance as expected in PSO and DE; they spent closed executions times for the two VANET instances since they have homogeneous internal operations this likeness in time consumption was withal registered in GA and ES.



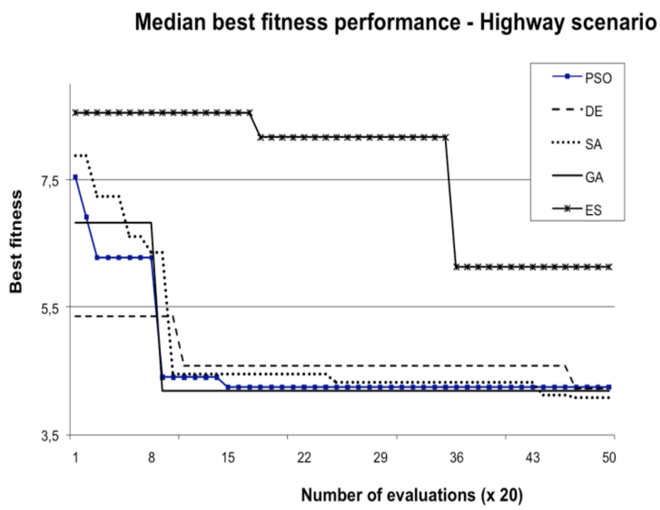


Fig. 5: Median fitness performance in urban scenario. Fig.6: Median fitness performance in highway scenario.

As a summary, the algorithms use between 9.00E+03 and 4.76E+03s for the Urban scenario (150 and 80 minutes, respectively), and between 2.19E+03 and 8.45E+02s for Highway scenario (60 and 23 minutes, respectively). This relative low effort in the protocol design is thoroughly justified by the subsequent benefits obtained in the global data transmission time and loss of packets once the VANET is physically deployed as observed in the following analysis.

E **Scalability Analysis**

Once we have analyzed the performance of the five algorithms in two different VANET scenarios, we study in this section how sundry network sizes affect the performance of these optimization techniques. For this purpose, we have generated two new VANET instances from the initial urban scenario (of Jaipur) by enlarging the metropolitan area considered. Hence, as Fig. 7 shows, the initial urban area (A1) has been augmented to A2 and A3 VANET areas. We have set the traffic flow as described in Section 6.1, also increasing the number of vehicles as follows:

TABLE6: Mean execution time (seconds) per independent run of each algorithm for both, urban and highway, scenarios.

Instance	**Algorithm**	T_{best}	T_{run}
Urban	PSO DE GA ES SA	4.68E + 03 4.37E + 03 3.48E + 03 5.46E + 03 **2.18E + 03**	7.95E + 03 7.12E + 03 6.68E + 03 9.00E + 03 **4.76E + 03**
Highway	PSO DE GA ES SA	1.39E + 03 9.82E + 02 8.83E + 02 9.84E + 02 **5.85E + 02**	2.19E + 03 2.10E + 03 1.56E + 03 1.47E + 03 **8.45E + 02**

TABLE7: Performance comparison in terms of mean fitness and mean optimization time (Tbest) of the three scaled Urban VANETs.

Algorithm	Mean fitness			T_{best}		
	Urban$_{A1}$	Urban$_{A2}$	Urban$_{A3}$	Urban$_{A1}$	Urban$_{A2}$	Urban$_{A2}$
PSO	**1.6346±0.2899**	**1.3920±0.2831**	**3.6763±0.4435**	7.95E+03	5.93E+03	1.20E+04
DE	1.7423±0.3717	1.4504±0.1885	3.9186±0.7419	7.12E+03	1.10E+03	1.43E+04
GA	1.9086±0.2260	1.4100±0.1235	3.6829±0.5063	6.68E+03	9.81E+03	1.41E+04
ES	2.1517±0.1266	1.5462±0.6023	3.7799±0.6227	9.00E+03	8.99E+03	1.50E+04
SA	2.7850±0.8718	2.3880±1.0207	3.8143±0.1260	**4.76E+03**	**3.40E+03**	**5.36E+03**

TABLE8: VDTP configurations and simulation output values for the ideal fitness achieved (in the median execution) by all studied algorithms.

Instance	Algorithm	VDTP configuration			Simulation results		
		Chunk size (Bytes)	Retransmission Time (s)	Attempts	Transmission Time (s)	Lost Packets	Data Transferred (kBytes)
Urban	PSO	41.358	10.00	3	**3.41**	**0.27**	1.024
	DE	28.278	6.00	9	3.59	0.63	1.024
	GA	31.196	3.83	9	3.61	**0.27**	1.024
	ES	23.433	10.00	8	3.50	**0.27**	1.024
	SA	19.756	6.43	3	4.22	0.36	1.024
	Human Experts	25.600	8.00	8	4.24	1.60	1.024
Highway	PSO	29.257	6.42	9	**24.67**	3.18	1.024
	DE	19.810	6.91	8	27.66	3.45	1.024
	GA	34.542	9.54	10	26.96	2.72	1.024
	ES	38.490	8.15	12	33.99	3.36	1.024
	SA	32.002	8.21	4	25.43	**2.54**	1.024
	Human Experts	25.600	10.00	10	33.08	3.27	1.024

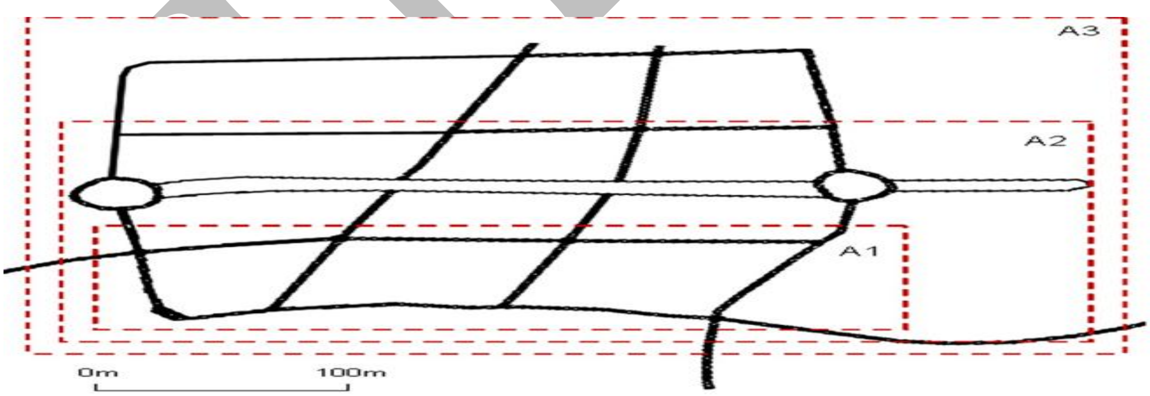


Fig. 6: Three urban areas from jaipur.

- $Urban_{A1}$ with 30vehicles in 120,000m2,
- $Urban_{A2}$ with 40vehicles in 240,000m2,
- UrbanA3 with 50vehicles in 360,000m2

Component From the point of view of the mean fitness acquired by each algorithm (out of 30 independent runs), we can observe in Table 7 that PSO keeps the best performance for $Urban_{A2}$ and $Urban_{A3}$.
Supplement ally, one of the most fascinating results can be observed in GA, which appears as the second best algorithm in amending its comportment with the VANET size ES obtains reasonable mean fitness values for all network instances, keeping a low standard deviation. The worst results are enlisted by SA in $Urban_{A2}$, and DE in $Urban_{A3}$. Concerning DE, the initial choice of its parameters (Cr=0.9 and μ= 0.1) could lead the algorithm to perform an exploitative search, hence obtaining good results in minuscule instances (the second best for $Urban_{A1}$) but damaging its demeanor in more immensely colossal VANETs (the worst for $Urban_{A3}$). In summary, excepting for GA and DE, we can attest that for the scaled VANET instances the performance of the algorithms are kindred to their performances in $Urban_{A1}$ (the initial Urban VANET instance) being PSO always the best procedure.

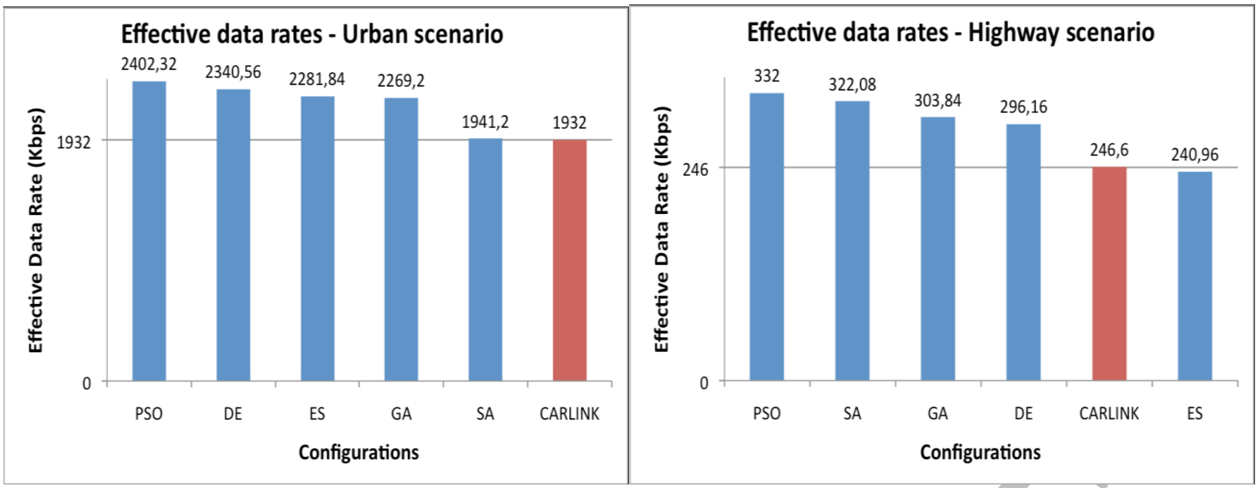
A trivial but withal intriguing observation lies in the mean fitness values, which are in $Urban_{A2}$ smaller than in $Urban_{A1}$. We suspect that, in spite of the more astronomically immense dimension of $Urban_{A2}$, the proportion of communicating vehicles (perm2) in this VANET avails the protocol operation especially for intermediate nodes, hence ameliorating the efficacious ratio of distribution packets and the overall retransmission time. This proportion could not be enough for $Urban_{A3}$ where the cost of transmissions is the more astronomically immense one.

Regarding the execution time, Table 7 shows in the three last columns the time required to gain the best solution (T_{best}) for each VANET instance. Especially, for PSO, ES, and SA the time required to unify in $Urban_{A2}$ is lower than in $Urban_{A1}$. This demeanor can be explicated by the fact of obtaining good solutions more expeditious in $Urban_{A2}$ than in $Urban_{A1}$, where the lower number of vehicles could ruin the communications conditions. On the contrary, the global runtime (T_{run}) always increases with the network size. This is of course a wonted result.

F **Quality of Service (QoS) Analysis**

Determinately, from the perspective of the worked VDTP configurations (solutions), we analyze the results in terms of the QoS be speakers considered here: the transmission time, the number of lost packets, and the quantity of data transferred induced in the designed VANET. In this sense, Table 8 exhibits the results after simulating the best solutions found by the studied algorithms. In advisement, the last row of this table contains the results of simulating the configuration of VDTP that has been utilized in the scope of the CARLINK project.

For the Urban VANET, the VDTP configuration obtained by PSO (Chunk Size=41,358Bytes, Retransmission Time =10s, and number of Attempts=3) achieves the best performance in terms of transmission time and mean number of lost packets. Concretely, in comparison with the human experts configuration of CARLINK, PSO obtains a reduction in the transmission time of 0.83s (19.5%) registering withal a lower number of lost packets.



a) Urban scenario b) Highway scenario

Fig. 8: Effective transmission data rates (throughput) in Kbps achieved during the simulations of the final VDTP

However, it is in the Highway scenario was PSO gets the higher time decline of 8.41s (25%) regarding the human experts configuration (from 33.08 to 24.67s).We must notice that, in spite of achieving the PSO a higher reduction in the transmission time than SA and GA, the fact of losing more packets (3.18 in PSO, 2.71 in GA and 2.54 in SA) in the global transference leads SA and GA to calculate a more preponderant fitness value (as shown in Table3). A final analysis can be done concerning one main QoS designator: the efficacious transmission data rate (throughput) achieved. As we can observe in Fig. 8, the VDTP configuration obtained by virtually all algorithms in the two VANET scenarios obtained higher efficacious data rates than the human configured VDTP. Concretely, PSO achieves the highest efficacious data rate (300.29kBytes/s in Urban and 41.54kBytes/s in Highway).This clearly claims for the utilization of these automatic algorithms to help human designers. We again remind that the genuine rectification of efficacious data rates between cars are in the order of tens of k Bytes/s, so our savings (58.79kBytes/s iUrban and 10.5kBytes/s in Highway) are genuinely consequential in current authentic applications such as safety, traffic control, and weather presages.

VII. CONCLUSIONS

In this paper, we tackle the ideal File Transfer protocol Configuration in VANETs by designates of five popular metaheuristic algorithms. For this, we require an intricate system accounting for a flexible simulation structure targeted for optimizing the transmission time, the number of lost packets, and the quantity of data transferred in simulated and additionally authentic VANET scenarios.

The experiments, utilizing ns-2 (well-kenned VANET simulator), disclose that all algorithms are capable of efficiently solve the Ideal File Transfer problem. In the comparisons, PSO performs statistically more preponderant than all algorithms in Urban and statistically more preponderant than DE and ES in Highway. In additament, GA and SA show a competitive performance in Highway. The scalability analysis shows that GA ameliorates with the network size, whereas DE decreases its performance with astronomically immense VANET instances. PSO keeps the best result even for more sizably voluminous instances. From the perspective of its authentic world utilization, PSO can reduce 19% of the transmission time in Urban and 25.43% in Highway with regards to human experts configuration of CARLINK, while transmitting the same amount of data (1024kBytes). The highest efficacious data rates obtained by PSO (of 300.39kBytes/s in comparison with 241.5kBytes/s of human experts) and DE (292.57kBytes/s) in Urban lead us to advise the final utilization of our design algorithms.

As a matter of further work we are presently elongating our benchmark with incipient VANET authentic instances (e.g. consummate cities and highway knots). In integration, we are orchestrating to define incipient optimized configuration schemes for other communication protocols such as: UDP, DSR, etc. which should effectively support authentic VANET design.

REFERENCES

1. Liming Zheng, Wanlei Li, Bo Xie, "Research on Communications over VANET under Different Scenes and Implementation of Vehicle Terminal", Wireless Communications, Networking and Mobile Computing (WiCOM), 2012 8th International Conference, pp. 1-7, sep. 2012.
2. Harri, J., Filali, F., Bonnet, C., "Mobility models for vehicular ad hoc networks: a survey and taxonomy", Communications Surveys & Tutorials, IEEE (Volume:11 , Issue: 4) pp. 19-41, Dec. 2009.
3. Blum, C., Roli, A., "Metaheuristics in combinatorial optimization: overview and conceptual comparison", ACM Computing Surveys (CSUR) 35(3), pp. 268–308, 2003.
4. Kennedy, J., Eberhart, R., "Particle swarm optimization", IEEE International Conference on Neural Networks 4, 1942–1948, 1995.
5. Price, K. V., Storn, R., Lampinen, J., "Differential Evolution: a practical Approach to Global Optimization", Springer, London, UK, 2005.
6. Beyer, H., Schwefel, H., "Evolution strategies: a comprehensive introduction", Journal Natural Computing 1(1), 3–52, 2002.
7. Rodriguez, F.J., Garcia-Martinez, C., Lozano, M.,"Hybrid Metaheuristics Based on Evolutionary Algorithms and Simulated Annealing: Taxonomy, Comparison, and Synergy Test", Evolutionary Computation, IEEE Transactions, Dec. 2012.
8. Alba, E., Luna, S., Toutouh, J., "Accuracy and efficiency in simulating vanets". In Modelling, Computation and Optimization in Information Systems and Management Sciences (MCO), Metz, France, pp.568–578, 2008.
9. The Network Simulator Project—Ns-2. [online] Available in URL (http://www.isi. edu/nsnam/ns/).
10. Vanhatupa, T., Hannikainen, M., Hamalainen, T., "Optimization of mesh WLAN channel assignment with a configurable genetic algorithm", In 1st International Workshop on Wireless Mesh (WiMeshNets'06), Waterloo, Ontario, Canada, 2006.
11. Alba, E., "A cellular multi-objective genetic algorithm for optimal broadcasting strategy in metropolitan MANETs", Computer Communications 30(4), pp. 685–697, 2007
12. Dorronsoro, B., Danoy, G., Bouvry, P., Alba, E., "Evaluation of different optimization techniques in the design of ad hoc injection networks", In: Workshop on Optimization Issues in Grid and Parallel Computing Environments, part of the HPCS, Nicossia, Cyprus, pp.290–296, 2008.
13. Di Caro, G. A., Ducatelle, F., Gambardella, L. M., AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks. European Transactions on Telecommunications 16 (5), pp. 443–455, 2005.
14. Chiang, F., Chaczko, Z., Agbinya, J., Braun, R., Ant-based topology convergence algorithms for resource management in vanets. In Lecture Notes in Computer Science of EUROCAST, vol.4739, pp.992–1000, 2007.
15. Huang, C., Chuang, Y., Hu, K., "Using particles warm optimization for QoS in ad-hoc multicast", Engineering Applications of Artificial Intelligence 22 (8), 1188–1193, 2009.
16. Alba, E., Toutouh, J., Luna, S., VDTP: a file transfer protocol for vehicular ad-hoc networks. Technical Report D2006/10, University of Malaga, Spain, 2006.
17. Johnson, D. B., Maltz, D. A., Broch, J., DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks. In: Perkins, C.E. (Ed.), Ad Hoc Networking, Chapter5, Addison-Wesley, pp.139–172, 2001.
18. Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P., "Optimization by simulated annealing", Science 220 (4598), pp. 671–680, 1983.
19. Alba, E., Luque, G., Garcıa-Nieto, J., Mallba: a software library to design efficient optimization algorithms. International Journal of Innovative Computing and Applications 1 (1), pp. 74–85. 2007.
20. Alba, E., Toutouh, J., and Luna, S., MEUs ad-hoc communications performance evaluation in highway scenarios (DYNAMIC tests in the CARLINK-UMA scenario) Technical Report D.2.2.5,University of Malaga, Spain, 2008.
21. Alba, E., Toutouh, J., and Luna, S., MEUs ad-hoc communications performance evaluation in urban scenarios (DYNAMIC tests in the CARLINK-UMA scenario). Technical Report D.2.2.4., University of Malaga, Spain, 2008.
22. Alba, E., Toutouh, J., and Luna, S., VanetMobiSim: the vehicular mobility model generator tool for CARLINK. Technical Report D1.3.1, University of Malaga, 2007.
23. Eberhart, R., Shi, Y., "Comparing inertia weights and constriction factors in particle swarm optimization", In Proceedings of the IEEE CEC, vol.1, La Jolla, CA, USA, 2000, pp. 84–88, 2000.
24. Wilcox, R., "New Statistical Procedures for the Social sciences", Hillsdale , 1987.
25. Iman, R. L., Davenport, J. M., "Approximations of the critical region of the Friedman statistic", Communications in Statistics 18, pp. 571–595, 1980.