Metaheuristic Algorithms Based Optimization for OLSR Routing in VANETs

Nandini G R

PG student,Dept. of CSE

CIT Tumkur,Karnataka, India

Mahesh N

Asst.Professor,Dept. of CSE

CIT Tumkur,Karnataka, India

Abstract— **In vehicular adhoc networks (VANETs) routing packets through the network is a difficult task because of the limited coverage of Wi-Fi and the high mobility of the nodes make regular topology changes and network fragmentations. For these reasons, and there is no central manager entity. Deployment of VANETs is crucial, therefore offering an efficient routing strategy. This paper deals with the optimal parameter setting of optimized link state routing (OLSR), is a well-known mobile ad hoc network routing protocol, by defining an optimization problem. To find automatically optimal configurations of this protocol, a series of metaheuristic algorithms (particle swarm optimization, differential evolution, genetic algorithm, and simulated annealing) are studied.**

Keywords—Vehicular adhoc networks(VANET),optimized link state routing (OLSR), metaheuristic, optimization algorithms.

I. INTRODUCTION

Vehicular Ad Hoc Networks (VANETs) [1] pursues the concept of ubiquitous computing for future. Nodes are vehicles, elements of roadside infrastructure, pedestrian personal devices and sensors. Deploying such networks uses Wi-Fi technologies. VANETs make the travel safer and fun as well.Vehicles can be turned into a network by incorporating the wireless communication and data sharing capabilities, VANETs are considered as an development of Mobile Ad hoc Networks (MANETs); however they have some distinct characteristics too.

VANETs are also related to MANETs in many ways. For example, both networks are multi-hop mobile networks and having dynamic topology. There is no central entity, and nodes route data themselves across the network. Both MANETs and VANETs are quickly deployable, without the need of an infrastructure. Although, MANET and VANET, both are mobile networks, however, the mobility pattern of VANET nodes is such that they move on specific paths (roads) and hence not in random direction. This gives VANETs some advantage over MANETs as the mobility pattern of VANET nodes is predictable. MANETs are often characterized by limited storage capacity and low battery and processing power. VANETs, on the other hand, do not have such limitations. Sufficient storage capacity and high processing power can be easily made available in vehicles. Moreover, vehicles also have enough battery power to support long range communication.

Another difference is highly dynamic topology of VANETs as vehicles may move at high velocities. This makes the lifetime of communication links between the nodes quite short.

Node density in VANETs is also unpredictable; during rush hours the roads are crowded with vehicles, whereas at other times, lesser vehicles are there. Similarly, some roads have extra traffic than other roads.

In VANETs, the high mobility of the nodes, the Wi-Fi boundaries in coverage and capability of the channel, and the occurrence of obstacles create packet loss, regular topology changes and network fragmentation. Thus, design efficient routing protocols [3], [4].And great deal of effort is committed to offer new medium access control access strategies [5]. Routing is a difficult task in such networks to find the routing paths among the nodes since there is no central entity. Different routing strategies have been defined by targeting the specific VANET needs of scenarios and applications. These protocols can be grouped into position based (e.g., greedy perimeter stateless routing (GPSR) and greedy perimeter coordinator routing), cluster based (e.g., clustering for open IVC network and location-based routing algorithm with cluster-based flooding), broadcasting (e.g., BROADCOMM and history enhanced vector based tracing detection) protocols And topology based (proactive, e.g., destination-sequenced distance-vector and optimized link state routing (OLSR), reactive, e.g., ad hoc On demand distance vector (AODV) and dynamic source routing (DSR) [4].

In the present paper, we aim at automatically tune OLSR [6], and solving an offline optimization. Although specific routing protocols are rising for VANET networks, a number of authors are currently using OLSR to deploy vehicular networks [7].This protocol has been preferred because it exhibits very competitive delays in the broadcast of data packets in large networks (which is an important feature for VANET applications), it adapts well to continuous topology changes, and the OLSR has simple operation that allows it to be easily integrated into different kind of systems and it presents a series of features that make it well suited for VANETs.

In summary, the main assistance of this paper are the following.
- We propose an optimization strategy in which a number of metaheuristic algorithms are (separately) coupled with a network simulator (ns - 2) to find quasi-optimal solutions.
- This optimization strategy is used in this paper to find as fine-tuned as possible configuration parameters of the OLSR protocol although it could directly be used also for a number of other routing protocols (AODV, PROAODV, GPSR, FSR, DSR, etc.).

- We obtain OLSR configurations that automatically out perform the standard one and those used by human experts in the current state of the art.
- We generate a set of realistic VANET scenarios based in the real area of Malaga, Spain. These instances are publicly available online for the sake of future experiments.

The remainder of this paper is organized as follows. In Section II, the OLSR routing protocol is introduced. Section III describes the optimization design followed to tackle the problem.Results are shown in Section IV. Finally, conclusion is considered in Section V.

II. PROBLEM OVERVIEW

Currently, there are several routing protocols for mobile ad hoc networks (MANETs) which may be used to find the best route to the destination vehicle in VANETs. However, due to the difference in the characteristics of the two networks, mobility characteristics that are present in VANETs could be added into current MANET routing protocols, so as to allow the routing protocol to achieve a better routing performance by finding the best route to the destination vehicle. Therefore, offering an efficient routing strategy is crucial to the deployment of VANETs. This project deals with the optimal parameter setting of the optimized link state routing (OLSR), which is a well-known mobile ad hoc network routing protocol, by defining an optimization problem. The main drawback of OLSR is the necessity of maintaining the routing table for all the possible routes. Such a drawback is negligible for scenarios with few nodes, but for large dense networks, the overhead of control messages could use additional bandwidth and provoke network congestion. This constrains the scalability of the studied protocol.

Thus, computing an optimal configuration for the parameters of this protocol is crucial before deploying any VANET, since it could decisively improve the QoS, with a high implication on enlarging the network data rates and reducing the network load. In addition, the target application is not considered to be particular, although there are more interested end-user services like infotainment, vehicle-to-vehicle multiplayer gaming, content distribution and sharing, etc. Such services rely on peer-to-peer communications and therefore unicast routing protocols like OLSR.

This protocol has been chosen since it presents a series of features that make it suitable for highly dynamic ad hoc networks and concretely for VANETs. These features are the following.

- Using OLSR, the status of the links is immediately known. Additionally, it is possible to extend the protocol information that is exchanged with some data of quality of the links to allow the hosts to know in advance the quality of the network routes.
- The simple operation of OLSR allows easy integration into existing operating systems and devices Without changing the format of the header of the IP messages.
- OLSR is a routing protocol that follows a proactive strategy, which increases the suitability for ad hoc networks with nodes of high mobility generating frequent and rapid topological changes, like in VANETs.

- The OLSR protocol is well suited for high density networks, where most of the communication is concentrated between a large number of nodes (as in VANETs.
- Its capability of managing multiple interface addresses of the same host, VANET nodes can use different network interfaces (Wi-Fi, Bluetooth, etc.) and act as gateways to other possible network infrastructures and devices (as drivers and pedestrian smart phones, VANET base stations, etc.).
- OLSR is particularly appropriate for networks with applications that require short transmission delays .

A. OLSR Protocol

In OLSR protocol, three levels of optimization are achieved. First, few nodes are selected as Multipoint Relays (MPRs) to broadcast the messages during the flooding process. This is in contrast to what is done in classical flooding mechanism, where every node broadcasts the messages and generates too much overhead traffic. Second level of optimization is achieved by using only MPRs to generate link state information. This results in minimizing the "number" of control messages flooded in the network. As a final level of optimization, an MPR can chose to report only links between itself and those nodes which have selected it as their MPR.

OLSR daemons periodically exchange different messages to keep the topology information of the complete network in the presence of mobility and failures. The core functionality is performed mainly by using three different types of messages: topology control (TC),multiple interface declaration (MID) messages and HELLO messages.

TC messages are generated periodically by MPRs to indicate which other nodes have selected it as their MPR. This information is stored in the topology information base of each network node, which is used for routing table calculations. Such messages are forwarded to the other nodes through the entire network. Since TC messages are broadcast periodically, a sequence number is used to distinguish between recent and old ones.

MID messages are sent by the nodes to report information about their network interfaces employed to participate in the network. Such information is needed since the nodes may have multiple interfaces with distinct addresses participating in the communications.

HELLO messages are exchanged between neighbor define a solution vector of real variables, each one representing nodes (one-hop distance). They are employed to accommodate link sensing, neighborhood detection, and MPR selection signaling. These messages are generated periodically, containing information about the neighbor nodes and about the links between their network interfaces.

The OLSR mechanisms are synchronized by a set of parameters predefined in the OLSR RFC 3626 [6] (see Table I).These parameters have been tuned by different authors without using any automatic tool in [8] and [9], and they are the timeouts before resending HELLO, MID, and TC messages(HELLO_INTERVAL,REFRESH_INTERVAL and TC_INT RA respectively); the "validity time" of the

information received via these three message types, which (MID), and TOP_HOLD_TIME (TC); the WILLINGNESS of a node to act as an MPR (to carry and forward traffic to other nodes); and DUP_HOLD_TIME, which represents the time during which the MPRs record information about the forwarded packets.

TABLE I : Main OLSR Parameters And RFC 3626 Specified Values

Parameter	Standard Configuration	Range
HELLO_INTERVAL	2.0s	$R \in [1.0, 30.0]$
REFRESH_INTERVAL	2.0s	$R \in [1.0, 30.0]$
TC_INTERVAL	2.0s	$R \in [1.0, 30.0]$
WILLINGNESS	3	$R \in [1.0, 30.0]$
NEIGHB_HOLD_TIME	3×HELLO_INTERVAL	$R \in [1.0, 30.0]$
TOP_HOLD_TIME	3× TC_INTERVAL	$R \in [1.0, 30.0]$
MID_HOLD_TIME	3× TC_INTERVAL	$R \in [1.0, 30.0]$
DUP_HOLD_TIME	30.0 s	$R \in [1.0, 30.0]$

B. OLSR Parameter Tuning

The standard configuration of OLSR offers a moderate QoS when used in VANETs. Hence, taking into account the impact of the parameter configuration in the whole network performance, we tackled here the problem of the optimal OLSR parameter tuning to discover the best protocol configuration previously to the deployment of VANET. The standard OLSR parameters are defined without clear values for their ranges.

Table I shows the standard OLSR parameter values, as specified in the OLSR RFC 3626. The range of values each parameter can take has been defined here by following OLSR restrictions with the aim of avoiding pointless configurations. According to that, we can use the OLSR parameters to define a solution vector of real variables, each one representing a given OLSR parameter. This way, the solution vector can automatically be fine-tuned by an optimization technique, with the aim of obtaining efficient OLSR parameter configurations for VANETs, hopefully outperforming the standard one defined in the RFC 3626.

III. OPTIMIZATION FRAMEWORK

As previously commented, the optimization strategy used to obtain automatically efficient OLSR parameter configurations is carried out by coupling two different stages: an optimization procedure and a simulation stage. The optimization block is carried out by a metaheuristic method, in this case one of those previously mentioned, i.e., PSO, DE, GA, and SA. These four methods were conceived to find optimal (or quasi-optimal) solutions in continuous search spaces.

We use a simulation procedure for passing on a quantitative quality value (fitness) to the OLSR performance of computed configurations in terms of communication cost. This procedure is carried out by means of the ns-2 network simulator widely used to simulate VANETs accurately . For this paper, ns-2 has been modified to interact automatically

with the optimization procedure with the aim of accepting new routing parameters, opening the way for similar future research.
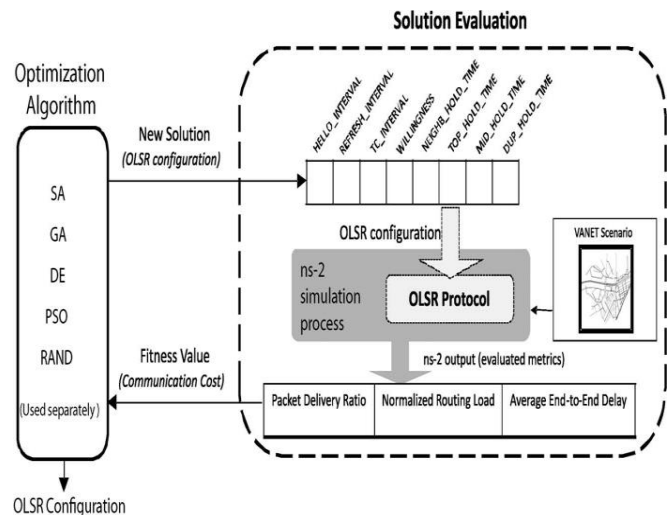

Fig 1: Optimization framework for automatic OLSR configuration in VANETs

As Fig. 1 illustrates, when the used metaheuristic requires the evaluation of a solution, it invokes the simulation procedure of the tentative OLSR configuration over the defined VANET scenario. Then, ns-2 is started and evaluates the VANET under the circumstances defined by the OLSR routing parameters generated by the optimization algorithm. After the simulation, ns-2 returns global information about the PDR, the NRL, and the E2ED of the whole mobile vehicular network scenario, where there were 10 independent data transfers among the vehicles.

A. Genetic Algorithm (GA)

Genetic Algorithms are possibly the most widespread subclass of EAs. As EA, it applies the natural selection principles to solve optimization problems. During successive generations, the solutions evolve to the optimum according to these principles. The evolution of these solutions depends largely on an adequate codification of them. Gas work with a population of individuals, each of which represents a feasible solution to a given problem. Each individual is assigned a value associated with the goodness of this solution (fitness). Similarly, in Nature, the fitness could be seen as a value of the adaptation and the effectiveness to compete for certain resources for biological organisms. The higher individual adaptation to the problem, the greater the probability to be selected to breed, i.e., crossing its genetic material with another individual. This crossing produces new individuals descendants of the above ones, which share some characteristics of their parents. So, spread the genetic material of the best individuals in successive generations. If the algorithm is designed properly, the population will converge to an optimum solution to the problem.

The algorithm manipulates a collection p of individuals (the population), each of which comprises one or more chromosomes. These chromosomes allow each individual

represent a potential solution for the problem under consideration. An encoding/decoding process is responsible for performing this mapping between chromosomes and solutions. Chromosomes are divided into smaller units termed genes.

The algorithm is an iterative process which starts by generating an initial population of solutions. This is typically addressed by randomly generating the desired number of solutions. When the alphabet used for representing solutions has low cardinality, this random initialization provides a more or less uniform sample of the solution space. Then, applying the genetic operators, recombination, mutation, and replacement on this population will produce a new gene.

Algorithm 1 : Genetic Algorithm (GA)

---

1. $p^0 \leftarrow$ initializePopulation()
2. Evaluation (p^0)
3. While g < maxGeneration or stopCondition () do
4. $p_s^g \leftarrow$ selection(p^g)
5. $p_r^g \leftarrow$ recombination(p_s^g)
6. $p_m^g \leftarrow$ mutation (p_r^g)
7. Evaluation(p_g^m)
8. $p^{g+1} \leftarrow$ replacement(p_m^g)
9. end While

IV. RESULTS

This section show metaheuristic and RAND performances when solving the OLSR optimization problem.

A. Performance Analysis

Table II shows the mean and standard deviation of the communication cost values obtained (out of 30 independent executions) running all the evaluated metaheuristic and the RAND for the VANET scenario instance. The best, median, and worst values are also provided. We can clearly observe in this table that SA outperformed all the other algorithms in terms of mean (-0.450297), median (-0.457451), and worst (-0.406932) communication cost values. According to these measures, SA obtained the best results, followed by DE, PSO, and GA, respectively. Finally, as expected, the RAND obtains worse results than all the metaheuristic algorithms. In terms of the best OLSR configuration returned by the algorithms (third column), PSO computed the solution with the lowest communication cost. The best OLSR parameter settings obtained by DE, SA, and GA are the second, third, and forth, respectively. The RAND best OLSR configuration is the least competitive one.

With the aim of given that these comparisons with statistical confidence, we have applied the Friedman and Kruskal–Wallis tests to the distributions of the results. We have used these nonparametric tests since the resulting distributions violated the conditions of equality of variances several times. The confidence level was set to 95% ($p - value = 0.05$), which allows us to ensure that all these distributions are statistically different if they result in $p - value < 0.05$. In effect, confirming the previous observations, the results of Friedman (see sixth column of Table II) test ranked Sas the algorithm with the best global performance followed by DE, PSO, and GA, respectively. The random search (RAND) showed the worst rank among the compared techniques. Moreover, the multicompare test of Kruskal–Wallis applied to the median values of the distributions resulted in $p - values$ _0.05 (last column of Table II). Therefore, we can claim that all the compared algorithms obtained statistically different results. According to the behavior of the optimization algorithms, we now study the evolution of the best solution (communication cost value) during the whole evolutionary process.

Finally, concerning the mean run time that each algorithm spent in the experiments, Table III shows the mean time in which the best solution was found T_{best} (second column) and the global mean run time T_{run} (third column).

GA shows the shortest time ($2.04E + 04$ s) to find its best solutions, and it seems that this algorithm fast falls in local optima, hence obtaining weak results (see Table II). Globally, PSO needed the second shortest time ($3.05E + 04$ s) to compute its optima followed by DE, RAND, and SA, respectively. In terms of mean run time T_{run}, random search takes shorter times ($4.36E + 04$ s) than the other algorithms since it has less internal operations. However, this algorithm converges to the weakest solutions. PSO is the metaheuristic that spent the shortest mean running time ($5.38E + 04$ s) followed by DE, SA, and GA, respectively.

According to Table III ,the analyzed algorithms use between $4.36E + 04$ and $1.18E + 05$ seconds (12.11 and 32.66 h, respectively) to finish each execution. This effort in the protocol design is completely justified by the subsequent benefits obtained in the global quality of service once the VANET is physically deployed.

.Table IV summarizes three rankings of the compared algorithms in terms of the mean fitness quality Mean fitness when solving the OLSR optimization problem, the mean time in which the best solution was found Tbest, and the mean run time Trun. In light of these results, we can claim that SA performs the best in terms of quality of solutions, although spending a higher time than other algorithms like PSO and DE to converge to such accurate solutions. In contrast, PSO offers the best tradeoff between the solution quality and the time required to find it. This is in fact a typical behavior of PSO, where the early convergence to successful solutions it performs makes this algorithm useful for time consuming problems as the choice for the present paper. This way, we can offer accurate OLSR configurations to experts in reasonable design times.

TABLE II: Results Obtained By Metaheuristic And RANDs In The Optimization Of OLSR For VANET Scenario. Results Of The Statistical Tests Of Friedman And Kruskal-Wallis(KW)

TABLE III: Mean Execution Time Per Independent Run Of Each Algorithm

Algorithm	T_{best}(seconds)	T_{run}(seconds)
PSO	3.05E+04	5.38E+04
DE	4.29E+04	7.95E+04
GA	2.04E+04	1.18E+04
SA	5.78E+04	1.04E+04
RAND	3.73E+04	4.36E+04

TABLE IV: Ranking In Terms Of Best Returned Solutions, Time To Find The Optimum T_{best} And Mean Execution Time T_{run}

Rank	$Mean_{fitness}$	T_{best}	T_{run}
1	SA	GA	RAND
2	DE	PSO	PSO
3	PSO	RAND	DE
4	GA	DE	SA
5	RAND	SA	GA

Alg	$Mean_{std}$	Best	Median	Worst	Fried	KW (p-value)
SA	-0.450±0.024	-0.478	-0.457	-0.407	1.40	3.059E-6
DE	-0.437±0.030	-0.480	-0.435	-0.393	2.10	3.066E-6
PSO	-0.432±0.033	-0.482	-0.420	-0.392	2.50	3.067E-6
GA	-0.351±0.023	-0.437	-0.345	-0.327	4.33	3.159E-6
RAND	-0.3308±0.050	-0.410	-0.330	-0.217	4.50	3.358E-6

V. CONCLUSION

The deployment of vehicular networks is a field with about ten years of intense research activity and progress. Nowadays, the widely adopted approach is equipping vehicles with WLAN devices (IEEE 802.11 family). If vehicles can directly communicate with each other and with roadside infrastructure, an entirely new paradigm for traffic safety and transport efficiency can be created.

In such networks, vehicles communicate within a limited range. In turn, VANETs are composed with high mobility nodes. Thus, they exhibit a topology that may change quickly and in unpredictable ways complicating the communication tasks. Therefore, it is crucial to provide user with an efficient configuration of the communication protocols in order to offer the best quality of service (QoS) possible previously to its deployment.

The proposed problem can be solved by employing metaheuristic algorithms (For examples SA, GA, PSO, and DE) using the Sumo/Ns-2 vehicular network simulator to evaluate the solutions.

REFERENCES

[1] Jamal Toutoch,Jose Garcia-Nieto and Enrique Alba, "Intelligent OLSR routing protocol optimization for VANETs",vol.61.no.4.May 2012.

[2] U. O. M. S. Corson, J. Macker, "MANET: Routing protocol performance issues and evaluation consideration", 1999. http://www.faqs.org/rfcs/rfc2501.html

[3] T. Taleb, E. Sakhaee, A. Jamalipour, K. Hashimoto, N. Kato, and Y. Nemoto, "A stable routing protocol to support ITS services in VANET networks," IEEE Trans. Veh. Technol., vol. 56, no. 6, pp. 3337–3347,Nov. 2007.

[4] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey,"IEEE Veh. Technol. Mag., vol. 2, no. 2, pp. 12–22, Jun.2007.[Online].Available:http://dx.doi.org/10.1109/MVT.2007.912927

[5] N. Lu, Y. Ji, F. Liu, and X. Wang, "A dedicated multi-channel MAC protocol design for VANET with adaptive broadcasting," in Proc. IEEEWCNC, Apr. 2010, pp. 1–6

[6] T. Clausen and P. Jacquet, "Optimized link state routing protocol (OLSR)," IETF RFC 3626, 2003. [Online]. Available: http://www. ietf.org/rfc/rfc3626.txt

[7] T. Chen, O. Mehani, and R. Boreli, "Trusted routing for VANET," in Proc.9th Int. Conf. ITST, M.Berbineau,M.Itami.and.G.Wen.Eds ,Oct.2009.pp.647-652.

[8] C. Gómez, D. García, and J. Paradells, "Improving performance of a real ad hoc network by tuning OLSR parameters," in Proc. 10th IEEE ISCC, 2005, pp. 16–21

[9] Y. Huang, S. Bhatti, and D. Parker, "Tuning OLSR," in Proc. IEEE 17thInt. Symp. PIMRC, Helsiniki, Finland, 2006, pp. 1–5.