Mobility-aware Ant Colony Optimization Routing for Vehicular Ad Hoc Networks

Sérgio Luis O. B. Correia
Graduate Program in Computer Science
State University of Ceara (UECE)
Fortaleza, Brazil 60.740–000
Email: sergio@larces.uece.br

Joaquim Celestino Júnior
Graduate Program in Computer Science
State University of Ceara (UECE)
Fortaleza, Brazil 60.740–000
Email: celestino@larces.uece.br

Omar Cherkaoui
Département d'informatique
Université du Québec à Montréal
Montréal, Canada QC H3C 3P8l
Email: cherkaoui.omar@uqam.ca

***Abstract*—Vehicular Ad hoc Networks (VANETs) are a special type of Mobile Ad hoc Networks (MANETs), made by vehicles communicating among themselves, and by vehicles communicating to devices located in the margins of roads and highways. The main characteristic of a VANET is the high speed of network nodes – that can go up to 200 km/h –, and that impacts directly on the ability the network has to deliver data, given we might have a network formed for just a small amount of time. It has been shown in several works that ant-based routing can be successfully applied to both wired and wireless networks. This work proposes Ant Colony Optimization (ACO) procedures that take advantage of information available in vehicular networks – such as the vehicles' position and speed –, in order to design an ant-based algorithm that performs well in the dynamics of such networks. The authors have also adapted the Dynamic MANET On-demand (DYMO) routing protocol to make use of the ACO procedures proposed in this paper, and the resulting bio-inspired protocol, MAR-DYMO, had its performance evaluated in an urban scenario and compared against a few other routing protocols. The obtained results suggest that making use of environmental information can make ACO algorithms more suitable for routing in vehicular ad hoc networks.**

I. INTRODUCTION

Vehicular Ad hoc Networks (VANETs) are a special type of Mobile Ad hoc Networks (MANETs), made by vehicles communicating among themselves (Vehicle-to-vehicle communication, V2V), and by vehicles communicating to devices located in the margins of roads and highways (Vehicle-to-infrastructure communication, V2I). The devices installed in the vehicles are called OBUs (on-board units), and the ones located in the roads, RSUs (roadside units). Figure 1 illustrates the V2X communications.

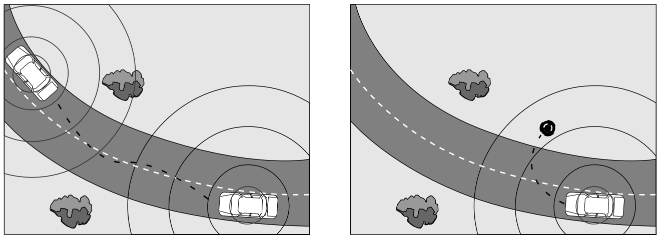


Figure 1. V2X communications: V2V on the left and V2I on the right.

In V2V, each OBU works in ad hoc mode, being able to forward messages through multiple hops, but in this mode the network connectivity is highly dependant on the vehicle's density and mobility pattern.

In short, the main characteristics of a VANET are the high speed of network nodes and their reasonably predictable mobility pattern, since the vehicles are supposed to follow the transit conventions, their trajectory is also supposed to follow the road lines.

The goal of this work is to devise and evaluate bio-inspired procedures that take into account some of the information available in vehicular networks, such as the vehicles' position and speed. The idea is to use this information to help the routing decisions, having, in the end, procedures that adapt well to VANETs.

The remainder of this paper is divided as follows: section II presents the related work; section III presents the Ant Colony Optimization metaheuristic; section IV presents our proposed bio-inspired procedures; section V develops, as a study case, a new DYMO variant that makes use of the proposed bio-inspired procedures; in section VI we propose a VANET scenario and evaluate a few routing protocols; and finally, section VII concludes this paper.

II. RELATED WORK

Härri, Bonnet and Filali devised a location-aware framework called Kinetic Graphs [1] for predicting and managing mobility. This framework is able to model a vehicle's trajectory at a given time instant. It assumes that, over a relatively short period of time, an arbitrary network node i will be following a linear trajectory, and hence its position can be modeled as a function of time by the means of 1st order kinetics, as shown in Equation (1).

$$\mathbf{Pos}_i(t) = \begin{bmatrix} x_i + v_{x_i} \cdot t \\ y_i + v_{y_i} \cdot t \end{bmatrix}, \tag{1}$$

where $Pos_i(t)$ is the position of the node i at time t, the vector $[x_i, y_i]^T$ denotes the initial position of the node i, and the vector $[v_{x_i}, v_{y_i}]$, its instantaneous velocity. In this work, we will apply the Kinetic Graphs framework to make information such as the vehicles' position and speed available to the routing protocol, so that it will be able to use it for making its forwarding decision, aiming at a better overall routing performance.

Sommer and Dressler evaluated the DYMO [2] routing protocol in a VANET [3] and concluded that, under lightweight vehicular and data traffic, it performs well. On the other hand, they also found out DYMO tends to add a considerable communication overhead, when there is a large amount of data to be transmitted.

In [4], a bio-inspired variant of the DYMO protocol – called Ant-DYMO – was developed, and it obtained satisfactory results with regards to data delivery and end-to-end delay when applied in some MANET scenarios. Those good results motivated the investigation of that algorithm in a VANET scenario, and in this work we will evaluate Ant-DYMO – along with other routing protocols – in the urban scenario proposed in section VI.

III. COMBINATORIAL OPTIMIZATION AND ANTS

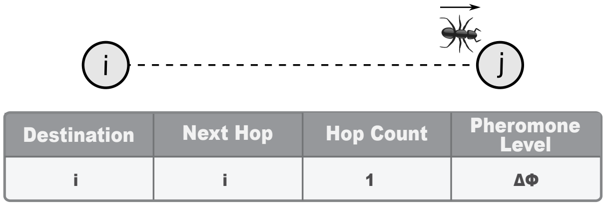
In the beginning of the 1990s, the combinatorial optimization based on ant colonies (ACO, *Ant Colony Optimization*) [5]–[7] emerged as a novel bio-inspired technique intended to solve hard combinatorial optimization problems. ACO is a metaheuristic [8], i.e., an approximate algorithm used for obtaining good enough solutions for hard combinatorial optimization problems within a reasonable computational time. The source of inspiration for the ACO metaheuristic was the foraging behavior of real ants. When looking for food, the ants initially explore the area nearby their nest, in a random way, but leaving – on the path they pass through – a chemical substance called pheromone This substance is later used by an ant to find the way back to the nest, as ants are able to follow pheromone tracks. As soon as an ant finds a food source, it evaluates the source and brings some of the food back to the nest. In the way back to the nest, it again deposits pheromone on the path it passed by. The amount of pheromone that was deposited in the path, that might depend on the quantity and quality of the food, will guide other ants to the food source. This indirect communication through pheromone trails enables the ants to find the shortest path between the nest and the food source [9]. This characteristic of the real ant colonies is exploited in artificial colonies, being the basis for the ACO technique.

IV. THE PROPOSAL: MOBILITY-AWARE ANT COLONY OPTIMIZATION ROUTING

The central component of an ACO algorithm is a parametrized probabilistic model, which is called the pheromone model [10], and in this context, two mechanisms are of the most importance: the way the pheromones are deposited in the paths where the ants pass through and the way these evaporate within the time. These two mechanisms will be discussed in the following subsections.

A. Pheromone deposit

The level of pheromone on a path indicates how good that path is, i.e., it reflects the quality of the path, its cost. Considering the link ℓ_{ij}, an ant walking on it – going from node i to node j –, deposits an amount of pheromone on the path. In terms of implementation, when the ant arrives at node j, the pheromone level of the entry – in the routing table of j – having i as both *destination* and *next hop*, is increased of a $\Delta\phi_{ij}$. Figure 2 shows the pheromone table of the node j after receiving an ant from the neighbor node i.



Destination	Next Hop	Hop Count	Pheromone Level
i	i	1	ΔΦ

Figure 2. Ant walking through the link ℓ_{ij}. The routing table at j is shown after the ant reaches j, and illustrates the pheromone deposit process.

$\Delta\phi_{ij}$ is defined in Equation (2) as being the amount of pheromone to be deposited in the route for the link ℓ_{ij}, and it will sum up with the previous pheromone level (assuming that route already existed), or just be the actual level, if it is a newly acquired route.

$$\Delta\phi_{ij} = P_R + \frac{t_{link}}{t_{MAX}} \tag{2}$$

In Equation (2), P_R is the expected probability of successfully receiving a message sent through a given distance, i.e, the probability of j receiving a message from i or vice versa, given the distance between them is d meters. t_{link} is the route lifetime, given by the Kinetic Graphs framework, and t_{MAX} is a value defined as the maximum route lifetime. This upper bound exists to avoid having too large values for the route lifetime estimation.

Considering the Nakagami fading model captures well the dynamics of a VANET in highways [11] and also urban scenarios [12], we are going to use it in our analytical and simulation models.

From [13], we have that, for the Nakagami distribution with a positive integer value for the fading parameter m, we obtain

$$P_R(d, CR) = e^{-m(d/CR)^2} \sum_{i=1}^{m} \frac{\left(m\,(d/CR)^2\right)^{i-1}}{(i-1)!} \tag{3}$$

which gives the probability of successfully receiving a message sent through a distance d between the sender and receptor and an intended communication range CR. Equation (3) does not consider the effects interference. Our pheromone deposit equation then becomes

$$\Delta\phi_{ij} = e^{-m(d_{ij}/CR)^2} \sum_{i=1}^{m} \frac{\left(m\,(d_{ij}/CR)^2\right)^{i-1}}{(i-1)!} + \frac{t_{link}}{t_{MAX}} \tag{4}$$

Equation (4) will determine the amount of pheromone to be deposited in every link visited by the ants, which will be the probability of reception of a message through this path summed with the ratio between the lifetime estimation of that path and the maximum allowed lifetime. The probability of reception is a good indicative of the path quality, but only in cases where the distance between the sender and the destination nodes is less than the wireless transmission range. For multi-hop routes whose length is bigger than the transmission range, the only indicative of quality we have is the estimated lifetime for that path.

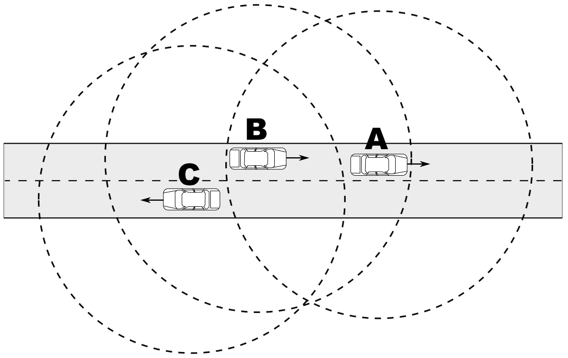


Figure 3. Road with three cars showing their transmission coverage area. B is neighbor of both A and C, but A and C are not neighbor themselves.

In Figure 3, if we consider the route from A to C (through B), the pheromone deposit procedure would take into account only the estimated route lifetime of the whole path – the minimum value of the estimated lifetime from each part of the path, A to B, and B to C –, since the distance between A and C is bigger than the transmission range, in this case (i.e., the probability of reception here would be zero).

B. Evaporation process

Just like it happens with real ants, an ACO algorithm simulates the evaporation process the pheromone trails – left by the ants while traveling through the links – experience.

From a practical standpoint, the evaporation process is necessary to avoid a too fast algorithm convergence towards a suboptimal region. In other words, it is a way to escape from a local optimum, favoring the exploration of new areas in the search space [10].

From an implementation standpoint, periodically – every t_{ev} seconds –, the pheromone level of all links decrease following a mathematical model that tries to imitate the evaporation mechanism of the real ants. In general, a simple formula such as the one shown in Equation (5) is used to model this evaporation mechanism, where ρ is the so-called *pheromone evaporation rate*, and ϕ, the pheromone level associated with that path.

$$\phi \leftarrow (1-\rho)\cdot\phi \tag{5}$$

In several ACO algorithms, such as ARA [14], ARAMA [15] and Ant-DYMO [4], to name a few, the evaporation rate ρ is fixed and the same for all links, usually a value found in some empirical way. Since we have available more information about the links than the usual ad hoc routing protocols – such as, AODV (Ad hoc on-demand distance vector routing) [16] or DYMO (Dynamic MANET on-demand routing) [2] –, we are going to propose different evaporation rates for every link, based on the assumption they are actually different, and hence, should behave differently in regard the pheromone evaporation. Through the Kinetic Graphs, we have now an estimation on the duration of a path, which will be represented by t_{link}, the same from Equation (2). With that information, we have an idea on when two nodes will stop being neighbors, i.e., the link between them will break, and we will set up the evaporation mechanism so that it will completely evaporate after t_{link} seconds. In practice, the route will be removed – evaporated – after the link is supposedly broken, so the idea is to eliminate invalid routes in a more accurate way, exactly when they become invalid.

We will now proceed to calculate the rate needed to evaporate a link within t_{link} seconds. Let $\phi^{[k]}$ represent the pheromone level of an arbitrary link after the evaporation process described by Equation (5) is performed k times. Hence,

$$\phi^{[k]} = \phi\cdot(1-\rho)^k \tag{6}$$

Assuming there are no other interferences in the pheromone level of this arbitrary link, such as pheromone deposit, Equation (6) shows the amount of pheromone associated with this link after it suffers the evaporation process k times. Let ε be a smallest amount (different of zero) of pheromone possible to be associated with a link, for which we assume the following:

$$\phi = \begin{cases} \phi, & \text{if } \phi \geq \varepsilon \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

Therefore:

$$\begin{aligned} \phi\cdot(1-\rho)^k &= \varepsilon \\ \rho &= 1-\left(\frac{\varepsilon}{\phi}\right)^{\frac{1}{k}} \end{aligned} \tag{8}$$

As the evaporation mechanism is performed every t_{ev} seconds (IV-B), we can calculate k, that tells the number of times the evaporation process was performed:

$$k = \frac{t_{link}}{t_{ev}} \tag{9}$$

Finally, by combining (9) with (8), we get the evaporation rate a given link will be subject to, so that it will completely evaporate after its predicted duration:

$$\rho = 1-\left(\frac{\varepsilon}{\phi}\right)^{\frac{t_{ev}}{t_{link}}} \tag{10}$$

V. APPLICATION OF THE PROPOSED TECHNIQUES TO AN EXISTING ROUTING PROTOCOL

In this section, we will apply the techniques presented in the previous section to the DYMO routing protocol, and obtain the Mobility-aware Ant Colony Optimization Routing DYMO, or simply MAR-DYMO, for short. Initially, we are going to describe the basic operations of the DYMO routing protocol, and then we describe our adaptions to it.

A. *Dynamic Manet On-Demand (DYMO)*

DYMO is a reactive and unicast routing protocol for multihop wireless networks and is considered a successor of the popular AODV. Its basic operations are the route discovery and route maintenance.

1) Route Discovery: The route discovery procedure is performed when a node in the network wants to send data to another node for which it does not know a route for. At this point, the first node creates a route request (RREQ) message and floods the network with it. The RREQ messages keep a list of visited nodes, so that a node, upon receiving a RREQ message, can update its routing table with a route to the source of this message. After receiving an RREQ message, the node checks whether it has a route to its destination, and if it does, it sends a route reply (RREP) message back to the source of the RREQ, telling it about this route. An RREP is also sent if the receiving node is the actual destination of the RREQ message.

2) Route Maintenance: The route maintenance procedure takes care of eliminating invalid routes from routing tables, attributing to each route a duration time. When a route is successfully used in the forwarding of data packets, the lifetime associated with that route is extended. When the route lifetime expires, its associated route is removed from the routing table. This way, DYMO only keeps routes that are being used.

B. *Mobility-aware Ant Colony Optimization Routing DYMO (MAR-DYMO)*

As we have seen, DYMO is a reactive protocol. We are going to turn it into an ACO algorithm by adding the *pheromone deposit* and *pheromone evaporation* discussed in section IV. We start by modifying the HELLO message from the DYMO protocol to add the fields shown in Figure 4.

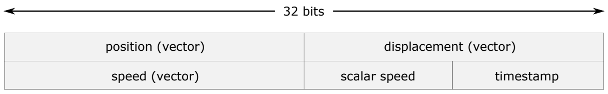


Figure 4. Data a vehicle needs to transmit in order to allow others to make predictions on its mobility.

Now our HELLO messages are not periodic anymore, as it is in DYMO, but they will be sent by the vehicles when needed, in an aperiodic fashion, managed by the Kinetic Graphs framework. That way the nodes will keep a table with updated info on their neighbors. This info allows a vehicle to make predictions on their neighbors, such as their position at a given time instant and the time they will be still neighbors, for instance.

Based on [17], we define the distance of $\frac{2}{3}$ of the transmission range to be covered before broadcasting our HELLO message – which contains the extra information displayed in Figure 4. With this distance and the speed of the car, one can easily predict the time a new broadcast will be performed. The interval between the initial and predicted times is then divided into seven equal time intervals, as shown in Figure 5, and after each interval t_i, a comparison between the vehicle's actual and predicted position is performed. If the absolute difference between the vehicle's positions is larger than a sufficiently small epsilon – the prediction error margin –, a new broadcast is performed, i.e., a new HELLO message will be sent. This broadcast method is used in order to avoid unnecessary broadcasts. When the vechicle itself notices its own information is outdated, it sends another HELLO message to give the other vehicles current info for making new and more accurate predictions on its mobility. To summarize, the HELLO messages will be sent either after every $\frac{2}{3}$ of the transmission range or when the vehicle notices the predictions are innacurate, when the difference between the actual and predicted positions is larger than the defined error margin.

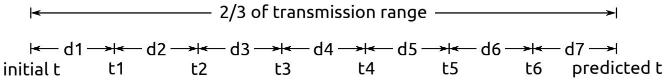


Figure 5. Time interval between the sending of HELLO messages

For turning the protocol into an ACO algorithm, the routing tables will be adapted to carry, for each route, the pheromone level associated with it, its evaporation rate – calculated through Equation (10), and the predicted lifetime. They are also changed to work in a multi-path fashion, i.e., they might have more than one route for the same destination – though with different nodes as the next hop –, and the actual route will be chosen among the existing ones based on their pheromone levels, by using the roulette-wheel selection method. Routes with a higher pheromone level will have a bigger probability of being chosen, but they will not necessarily be the selected ones.

$$p_i = \frac{\phi_i}{\sum_{j=1}^{N} \phi_j} \tag{11}$$

Equation (11) shows the probability a route i to a given destination will be chosen, among N routes for the same destination.

VI. PERFORMANCE EVALUATION

A. *Simulation*

The simulations were carried out using the Network Simulator 2.34 [18] and using the modules *Mac802_11Ext* and *WirelessPhy-Ext*, that are based on the original ns-2 wireless modules, but improve those in the sense they provide for a significantly higher level of simulation accuracy [19], due to a more accurate implementation of the IEEE 802.11 MAC and PHY layers. The Nakagami radio propagation model is used. The vehicular traffic was generated by the Vehicular Network Movement Generator [20] according to the car-following and lane-changing models proposed by Gipps [21], [22], which belong to the class of collision avoidance vehicular mobility models. In this mobility model, the cars move at the maximum safest speed that ensures that there will be no collision with the preceding vehicle. The following routing protocols were compared against each other: AODV [16] (ns-2 native implementation), DYMO [2] (DYMOUM [23] implementation), Ant-DYMO [4] (the implementation from [4]), and the one DYMO variant developed in the previous section, MAR-DYMO. The simulation parameters are summarized in Table I.

Table I
SUMMARY OF THE SIMULATION PARAMETERS

	Simulation Parameters
Scenario Area	$1600m \times 1500m$
Communication Range	$350m$
Propagation Model	Nakagami ($m = 1$)
Mobility Model	Gipps
Application	CBR (constant bitrate)
Transport	UDP
MAC and PHY	802.11p
Packet size	512 bytes
Transmission Rate	4 packets/second
Interface Queue	20 packets
Simulation Time	150 seconds
Number of vehicles	$25, 50, 75, 100, 125$ and 150
Routing Protocols	AODV, DYMO, Ant-DYMO and MAR-DYMO
Number of runs	15 times
Confidence Interval	95%

1) Scenario: The scenario is a $1600m \times 1500m$ urban area, made of grids as shown in Figure 6. The roads have two lanes each and the vehicular traffic flows in both directions.

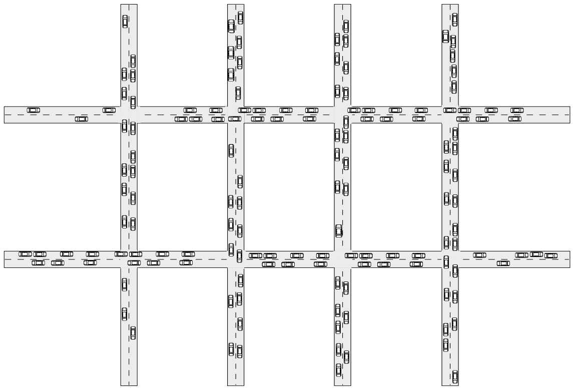


Figure 6. Scenario evaluated

2) Data Traffic: The traffic used in the simulations is UDP (CBR, constant bit rate). During the whole simulation, each node will be sending data to a random destination for a time chosen between 10 and 60 seconds. After that, it will not send for a time chosen between 0 and 5 seconds. After this pause period, it will pick another destination randomly and repeat the process until the simulation ends.

3) Metrics: We considered three metrics in order to evaluate the routing protocols in the proposed scenario: – the average delivery ratio, the average end-to-end delay and the routing overhead.

The *average delivery ratio* is defined as being the number of packets successfully received at the destinations per number of packets sent by the data traffic sources. This metric tells how good was the protocol in the task of sucessfully transmitting data end-to-end.

The *average end-to-end delay* is defined as being the average of the the sum of the time it took to send/receive each of the successfully delivered packets, and it tells about the latency of the protocol.

The *routing overhead* is defined as being the number of routing packets – such as the protocol messages – per number of data packets successfully received at the destinations. This metric tells about the extra traffic generated by the routing protocol in order to successfully deliver data packets.

B. Results

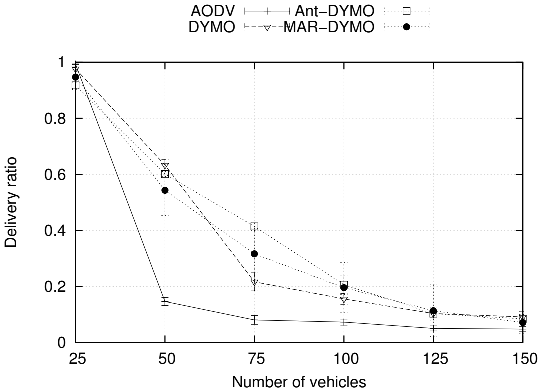


Figure 7. Average delivery ratio

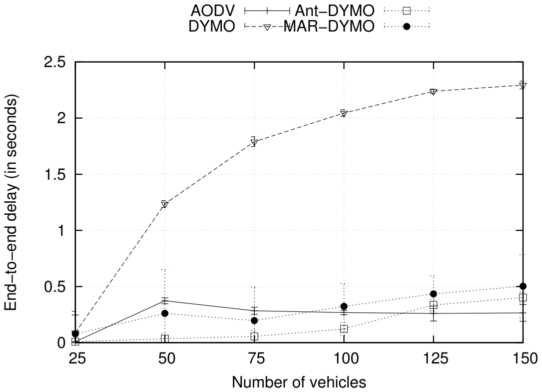


Figure 8. Average end-to-end delay

Figure 7 shows DYMO and its two bio-inspired variants performed better than AODV in regard to the data delivery ratio, i.e., they were able to deliver more data packets. Ant-DYMO performed noticeably better when we had 75 cars moving on the scenario.

With regard to the end-to-end delay, Figure 8 shows pure DYMO had the worst performance from the compared protocols. AODV had a very good performance in this metric. When

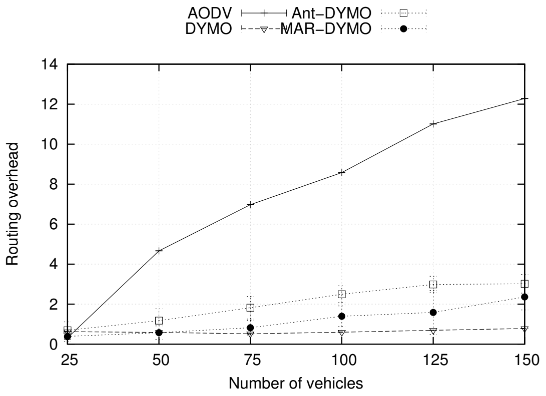


Figure 9. Routing overhead

the number of vehicles in the simulation was varying between 25 and 100, Ant-DYMO again had a better performance, with MAR-DYMO staying not too far behind. It is worth noting that Ant-DYMO has a proactive phase – with ants that are looking for new routes –, while all the other considered protocols do not, and work typically on-demand. The advantage Ant-DYMO got over its competitors come from these extra routes. MAR-DYMO had good performance as well, considering its reactive nature, and this good performance is believed to be due to the smart route evaporation that considers the predicted route lifetimes.

Figure 9 shows the routing overhead, and once more, AODV had the worst performance of the compared protocols. This metric considers the ratio of control packets over the delivered data. Since AODV had the worst delivery rate, it was expected its overhead was going to be high. Ant-DYMO had more overhead from the DYMO protocols, and that is again, due to the proactive ants. On the one hand, they are good for the network connectivity, meaning there will be more routes available, but on the other hand, they generate much more control traffic. MAR-DYMO was able to generate less overhead than Ant-DYMO, although it generated more than DYMO. This is due to the aperiodic HELLO messages described. Even if it is aperiodic, sometimes it is needed to send more update messages than a simple periodic protocol such as DYMO would.

VII. CONCLUSION AND FUTURE WORK

This paper proposed two ACO procedures to be applied to routing protocols intended to be used in VANETs. They use information available in such networks, such as the vehicles' position and speed, and through a framework to make predictions on the mobility of the neighboring vehicles, such procedures should add to the overall performance of such VANET-adapted routing protocol. We have developed a simple case study where the DYMO protocol was modified to implement the two bio-inspired proposed procedures and the results showed this new DYMO variant – called MAR-DYMO – performed pretty well in the proposed evaluation scenario. In regard to end-to-end delay, it performed much better than pure DYMO, and was good enough when compared to the hybrid protocol Ant-DYMO.

In the near future we envision making a MAR-DYMO variant that also employs a proactive phase, to evaluate the gain it potentially might bring to the total delay and delivery rate.

REFERENCES

[1] J. Härri, C. Bonnet, and F. Filali, "Kinetic mobility management applied to vehicular ad hoc network protocols," *Computer Communications*, vol. 31, no. 12, pp. 2907–2924, 2008.

[2] I. Chakeres and C. Perkins, "Dynamic manet on-demand (DYMO) routing," *draft-ietf-manet-dymo-21 (work in progress)*, 2010.

[3] C. Sommer and F. Dressler, "The DYMO routing protocol in VANET scenarios," in *2007 IEEE 66th Vehicular Technology Conference, 2007. VTC-2007 Fall*, 2007, pp. 16–20.

[4] J. Martins, S. Correia, and J. Celestino, "Ant-DYMO: A bio-inspired algorithm for MANETS," in *Telecommunications (ICT), 2010 IEEE 17th International Conference on*. IEEE, 2010, pp. 748–754.

[5] M. Dorigo, "Optimization, learning and natural algorithms (Italian)," Ph.D. dissertation, Politecnico di Milano, Milan, Italy, 1992.

[6] M. Dorigo, V. Maniezzo, A. Colorni, M. Dorigo, V. Maniezzo, and A. Colorni, "Positive feedback as a search strategy," *Dipartimento di Elettronica e Informatica, Politecnico di*, 1991.

[7] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 26, no. 1, pp. 29–41, 1996.

[8] F. Glover and G. Kochenberger, *Handbook of metaheuristics*. Springer, 2003.

[9] J. Deneubourg, S. Aron, S. Goss, and J. Pasteels, "The self-organizing exploratory pattern of the argentine ant," *Journal of Insect Behavior*, vol. 3, no. 2, pp. 159–168, 1990.

[10] M. Dorigo and C. Blum, "Ant colony optimization theory: A survey," *Theoretical Computer Science*, vol. 344, no. 2-3, pp. 243 – 278, 2005.

[11] V. Taliwal, D. Jiang, H. Mangold, C. Chen, and R. Sengupta, "Empirical determination of channel characteristics for DSRC vehicle-to-vehicle communication," in *Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks*. ACM, 2004, p. 88.

[12] L. Rubio, J. Reig, and N. Cardona, "Evaluation of Nakagami fading behaviour based on measurements in urban scenarios," *AEU-International Journal of Electronics and Communications*, vol. 61, no. 2, pp. 135–138, 2007.

[13] M. Killat and H. Hartenstein, "An empirical model for probability of packet reception in vehicular ad hoc networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, pp. 1–12, 2009.

[14] M. Gunes, U. Sorges, and I. Bouazizi, "ARA-the ant-colony based routing algorithm for manets," in *proceedings of the ICPP international Workshop on ad hoc networks (IWAHN)*. Citeseer, 2002.

[15] O. Hussein and T. Saadawi, "Ant routing algorithm for mobile ad-hoc networks (ARAMA)," in *Performance, Computing, and Communications Conference, 2003. Conference Proceedings of the 2003 IEEE International*. IEEE, 2003, pp. 281–290.

[16] C. Perkins, E. Belding-Royer, and S. Das, "RFC3561: Ad hoc on-demand distance vector (AODV) routing," *Internet RFCs*, 2003.

[17] J. Härri, "Modeling and predicting mobility in wireless ad hoc networks," Ph.D. dissertation, EPFL, Lausanne, 2007. [Online]. Available: http://library.epfl.ch/theses/?nr=3836

[18] N. Simulator, "ns-2," 1989.

[19] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, and H. Hartenstein, "Overhaul of IEEE 802.11 modeling and simulation in ns-2," in *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*. ACM New York, NY, USA, 2007, pp. 159–168.

[20] J. Nzouonta, "Vehicular network movement generator." [Online]. Available: http://web.njit.edu/~borcea/invent/

[21] P. G. Gipps, "A behavioural car-following model for computer simulation," *Transportation Research Board*, vol. 15, pp. 105–111, 1981.

[22] ——, "A model for the structure of lane-changing decisions," *Transportation Research Board*, vol. 20B, no. 5, pp. 403–414, 1986.

[23] P. Ruiz and F. Ros, "DYMOUM – A DYMO implementation for real world and simulation," *URL http://masimum. dif. um. es*, 2005.