Optimizing Communications in Vehicular Ad hoc Networks Using Evolutionary Computation and Simulation

Wahabou Abdou
Computer Science Laboratory of the University of Franche-Comté
1, cours Louis Leprince-Ringuet
25201 Montbéliard France
+ 33 (0) 3 81 99 47 77
wahabou.abdou@univ-fcomte.fr

Adrien Henriet
Computer Science Laboratory of the University of Franche-Comté
1, cours Louis Leprince-Ringuet
25201 Montbéliard France
+ 33 (0) 3 81 99 47 43
adrien.henriet@univ-fcomte.fr

Dominique Dhoutaut
Computer Science Laboratory of the University of Franche-Comté
1, cours Louis Leprince-Ringuet
25201 Montbéliard France
+ 33 (0) 3 81 99 47 78
dominique.dhoutaut@univ-fcomte.fr

François Spies
Computer Science Laboratory of the University of Franche-Comté
1, cours Louis Leprince-Ringuet
25201 Montbéliard France
+ 33 (0) 3 81 99 47 13
francois.spies@univ-fcomte.fr

Christelle Bloch
Computer Science Laboratory of the University of Franche-Comté
1, cours Louis Leprince-Ringuet
25201 Montbéliard France
+ 33 (0) 3 81 99 47 77
christelle.bloch@univ-fcomte.fr

ABSTRACT
Broadcasting efficiently in a Vehicular Ad hoc Network (VANET) is a hard task to achieve. An efficient communication algorithm must take into account several aspects such as the neighboring density, the size and shape of the network, the use of the channel, the priority level of the message. Some studies [6,12,13] have proposed new solutions of broadcasting on such a network, but it is quite hard to evaluate their performance in various contexts. In order to determine the best repeating situation for each node in the network according to its environment, we developed a tool combining a network simulator (NS2) and an evolutionary algorithm. In this paper, we study four types of context and we tackle the best behavior for each node to determine the right input parameters. These studies are necessary to develop efficient broadcast algorithms in VANET.

Categories and Subject Descriptors
C.2.2 [**Network Protocols**]
I.2.8 [**Problem Solving, Control Methods and Search**]: Heuristic methods.

General Terms
Algorithms, Design, Experimentation.

Keywords
VANET, Broadcast, Flooding, Context dissemination, Multi-objective Evolutionary Algorithm.

1. INTRODUCTION
Inter-vehicular communications assist drivers by giving them road traffic (e.g. traffic jam) and security (e.g. accidents) information. Due to the high mobility of vehicles, unicast transmissions are not appropriate to such a network [6]. Communication protocols such as existing routing protocols are based on a simplistic form of broadcasting known as "Flooding", in which each node (or all nodes in a localized area) retransmits each received unique packet exactly once [12]. The resulting problem is the consumption of the bandwidth by useless retransmissions. It should be noted that the use of the bandwidth is closely linked to the density of vehicles in the considered area.

In a dense environment, if each vehicle retransmits each message as soon as it is received, the number of collisions will grow quickly, preventing potential highly relevant and time-critical messages from getting access to the shared wireless medium. In low density, if vehicles rarely relay the transmissions, the rebroadcasting chain might be broken (while using a realistic propagation model). The behavior of the vehicles (regarding message retransmission) must depend on the context. The problem is how to communicate efficiently without unnecessarily saturating the channel.

Many researches focus on message dissemination strategies in Vehicular Ad hoc Networks (VANETs). [11] and [12] present mechanisms to reduce redundancy, contention and collision in Mobile Ad hoc Networks using probability, area or neighbor knowledge-based methods. In [6] an altruistic communication scheme which differentiates messages by their relevancy is proposed.

In this paper, we suggest the use of four parameters to adapt the message dissemination to the context:

- the probability for each vehicle to rebroadcast a message;
- the number of times each vehicle retransmits each message;
- the delay between two retransmissions;
- the TTL (Time To Live) of messages.

We intend to establish a platform that will allow various vehicles to detect the density of their environment and adjust their communication parameters automatically. This paper presents the choice of parameters depending on a given context. It is the first step of the project. Automatic detection of the context will be our future work. An evolutionary algorithm (EA) is used to explore the possible settings. Then considering each of these settings (i.e. a set of parameters) the behavior of a set of vehicles is simulated using NS2 (Network simulator 2) [1]. Our goal is to reduce the channel utilization and the time spent for a complete transmission (all vehicles receive the message) in a given area when broadcasting.

The remainder of this paper is organized as follows. Section 2 presents different methods of broadcasting in wireless ad hoc networks. Section 3 describes our approach which consists in using an evolutionary algorithm to determine the optimal parameters of message dissemination for a given context. Section 4 evaluates this proposition and presents simulation results. Section 5 presents concluding remarks and outlines future work.

2. RELATED WORK

Some work has been done to optimize broadcasting in wireless ad hoc networks. The objective is to reduce the number of broadcasts without decreasing reachability or increasing latency. The proposed techniques can be classified into five groups [11,12,13]: simple flooding, probabilistic, counter-based, area-based, and neighbor-knowledge-based methods.

In simple flooding methods, each node rebroadcasts a message only the first time it receives it. The message references are stored to avoid later rebroadcasting. If this technique gives interesting results in a sparse environment, when the density is high, many relays are redundant and waste the channel bandwidth [13].

The probabilistic methods are proposed to improve the simple flooding. When a node receives a message, it rebroadcasts or drops it depending on a given probability. If this probability is set to 1, this technique is equivalent to the simple flooding. The problem of this method is the determination of the appropriate value of the probability. Even if values between 0.6 and 0.8 as rebroadcast probability are considered optimal [8], it is clear that these values are not likely to be globally optimal [13].

The counter-based methods are based on the fact that the most a message is received by the same node, the less an additional area will be reached if this node rebroadcast it as demonstrated in [11]. When a node receives a message for the first time, it initializes the counter with a value of 1 and sets a Random Assessment Delay (RAD) which is a time chosen in a given interval. Before the expiration of the RAD, each time the same message is received, the counter is incremented by 1. When the RAD is over, if the counter is less than a given threshold value, the message is rebroadcast. Otherwise the message is discarded. In [11] Tseng et al., showed that this threshold could be lower than 6. It should be noted that this implies an additional latency.

When using an area-based method, before rebroadcasting a message, the node evaluates the additional coverage area which will result upon this retransmission. In [12] it is mentioned that this technique does not consider whether nodes exist within that area. To evaluate the additional coverage area, the node can use the distance between itself and each node that has previously rebroadcast the message (distance-based scheme) or the geographical coordinates (location-based scheme). The GPS is not indispensable in the distance-based scheme (the signal strength can be used to calculate the distance). In both distance-based and location-based schemes, a RAD is assigned before the message is rebroadcasted (if the additional coverage area is higher than a fixed threshold) or dropped. It is possible for VANET to improve this method by using knowledge of way, and direction of node, and not only area, which can change.

In the neighbor knowledge-based approaches, the nodes exchange a "Hello" packet periodically and build a 1-hop neighbors' list. When transmitting a message, the node adds its list. So the receiver compares the sender list to its own list and retransmits the message only if all his 1-hop neighbors are not included in the sender's list. This is the self-pruning method. Another approach known as Scalable Broadcast Algorithm uses a 2-hop neighbors' list in the same way. This method is inadvisable for vehicular ad-hoc network because of the relevancy about the information on neighbors. In fact, with high-velocity mobile network, the information about neighbors is quickly inaccurate. In this case, the performance of this algorithm is very poor. However, for static or low mobility network, it is a very fair method.

The first step of our project deals with probabilistic broadcast, by taking into account other parameters like the delay between two retransmissions of a message by a vehicle, the number of retransmissions and TTL. Since this increases the complexity of the problem, an EA is used to solve it. We have not found in literature any work applying EAs to VANET communications. The next section describes the proposed approach.

3. HYBRID APPROACH TO OPTIMIZE COMMUNICATIONS

The proposed approach is based on two main modules, an optimization engine and a simulation engine, which cooperate to identify the best choice(s) of parameters for a given context. Figure 1 illustrates these two sub-systems and their interactions.

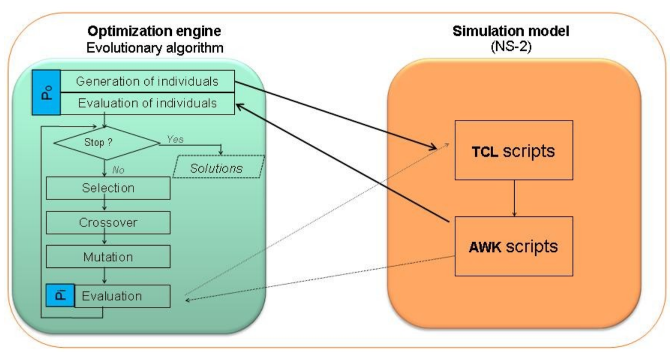

Figure 1: Interaction between the two sub-systems

The optimization engine uses an evolutionary algorithm (EA) and the simulation engine uses NS2. This section first presents some basic elements about EA and NS2. Then, it details how each of them has been used to optimize the communications in VANET.

3.1 Introduction to Evolutionary Algorithms

Evolutionary algorithms (EAs) draw an analogy between the solutions of an optimization problem and individuals in nature. They first build an initial population P_0, containing *n* randomly generated individuals, which are the possible solutions to the problem. An evaluation process assigns a fitness value to each of them. The more interesting a solution is, the greater its fitness. Thereafter, in the selection phase, the probability to select an individual is proportional to its fitness. Thereby, the best individuals are most likely to become parents. The recombination of two parents uses a crossover operator to generate new individuals, called children. By analogy with natural selection and reproduction, children inherit qualities from their parents. Repeating these steps to create several successive populations P_i (also called generations), the algorithm evaluates and compares many solutions, while continuously increasing their quality. An additional operator, called mutation also generates new solutions to prevent the EA from being trapped by local optima. Crossover and mutation are applied with given crossover and mutation rates. Finally, according to a given stop criterion, the algorithm returns the best solution(s) it has found.

3.2 Introduction to Packet based Network Simulators

VANETs may contain thousands of nodes (Vehicles). Communication solutions cannot be evaluated nor by representative real hardware experiments nor in a deterministic way. This is why simulation is very important. There are lots of network simulators (NS, GloMoSim, OMNeT++,...) but NS2 [1] is the most reliable in our case, the most active open-source network simulator, and the most frequently used for wireless simulation. It uses discrete event simulation, which is based on the change of events and states, because it is a faster more realistic method. It is also packet-based, which means that we simulate the transport of data packets on network topology. For our experiments, we want to evaluate the packet propagation in a wireless scenario. There is some limits to NS2 about the propagation model for wireless communication, it is why we use an internal improvement with a Shadowing Pattern model [5] to have a more realistic error model. The simulator works with scenarios which are evaluated by the program. A scenario is a set of start parameters, explicit nodes for communication, and time events. Each new event (in time) changes the state of the simulator, and can modify some parameters. The scenarios used here are either a single or a double line of nodes which transmit one packet or repeat transmitted packets.

3.3 Context-Oriented Broadcasting Strategy into VANET

The proposed system hybridizes EAs and NS2 to determine the best set of parameters (or settings) of a probabilistic broadcasting strategy for different given environments. The EA explores a set of possible settings**.** NS2 is used to assess the quality of each considered setting by simulation. The following sections detail this cooperation between these two sub-systems. Section 3.3.1. presents the studied kind of strategy. It particularly specifies the settings (based on four parameters) that are used to make this broadcasting strategy context-oriented. Finally, it indicates which kinds of simulation environments have been used to compute the values of these criteria for each examined setting. Section 3.3.2 shows how these parameters and criteria are used in the EA to determine the best settings for each of the considered environments. It describes the different elements of the algorithm: the genotypic representation of each considered solution (i.e. setting), the evaluation of its fitness according to the value of the different performance criteria it obtained by simulation, the selection process, the recombination operators (crossover and mutation) and the stop criterion.

1.1.1 Parameters and performance criteria in a multi-hop model

As previously explained, our context-aware diffusion protocol is based on a probabilistic broadcasting strategy. Each node is a smart repeater. When receiving a message, it decides whether it must rebroadcast it or not using four parameters to adapt its behavior to the current environment. These parameters have been chosen so that they remain as few as possible in order to enable fast genetic computations. Yet they can combine and express optimized solutions in an extremely large array of environments. These parameters are:

- P : The probability to start repeating a packet when receiving it for the first time. Inherited from probabilistic flooding algorithms, this parameter enables the tuning of the contention for the radio channel, especially in high density environments.
- Nr : The total number of repeats (applied only when a node has decided it should repeat a packet). It enables the protocols to cope with low to extremely low node densities. In mobile environments, when repeating a packet, one cannot be sure there will be a neighbor to repeat and process it further. Repeating this packet again over time maximizes the probability a passing node will receive it, at a price of channel over-use. This parameter is also needed to limit the diffusion over time.
- Dr : The delay between repeats (also only applied if the node initially chooses to repeat the packet). This parameter is used to tune the channel use and, in conjunction with the total number of repeats to tune how long a packet will be broadcast.
- TTL : The Time To Live of a packet, expressed as a number of hops. This parameter also limits the diffusion over time. This parameter could be complemented or replaced by time and / or geographic limitations. However a lot more computations would be required to optimize and should be explored in future work.

In the proposed approach, EAs and NS2 are used to optimize these four parameters in any given environment. The EA explores the set of possible values for each of them. A given combination of values is called a setting. NS2 is used to assess the quality of each considered setting by simulation. Several objectives are used to determine whether a solution is interesting or not. The given values of the parameters must enable to use the communication network efficiently without saturating it. Four criteria are used to characterize this. Three of them must be minimized:

- Nc : the number of collisions,
- T : the time spent until the last node receives the message,
- and R : the number of retransmissions during the simulation run.

As simulation is a stochastic process, five hundred simulations are run to evaluate the mean values of these objectives for each setting. In each of these simulations, if the channel is saturated, the number of collisions may prevent the message from being normally transmitted to all the nodes. In this case, the simulation is stopped, and this is stored as a failed simulation. This permits to compute a fourth criterion for each setting, called full reception ratio in a limited area (FR), which equals the ratio between the number of successful simulations and the total number of simulations. This criterion must be maximized. Besides, if it is less than a given threshold, the considered setting is said to be unfeasible. Otherwise, the setting is feasible and the successful simulations are used to compute the mean values of the three other criteria.

1.1.2 Optimizing broadcasting settings

The previous section described a multi-objective optimization problem. Evolutionary algorithms (EAs) are known to solve such problems efficiently in various fields of application [2,10]. That is why we chose this kind of heuristics to solve the current problem. The first step of the project uses a very simple multi-objective EA. It does not include any mechanisms of diversification such as those used in classical multi-objective EAs like the very famous NSGA2 [4]. Indeed, the main goal of this step was to estimate the feasibility quickly and to check that EAs are tools adapted to this problem. Besides, the evaluation phase based on NS2 simulation is rather time-consuming. Therefore the developed version of EAs had to be all the more simple to limit the computation time required by optimization operations. Each individual represents a setting of the broadcasting strategy by encoding each of the considered multi-hop model parameters (P, Nr, Dr and TTL) as integer numbers. For instance, the individual presented in figure 2 corresponds to the following setting of these decision variables: the probability P equals 0.5264603, the number of retransmissions is 2, the delay between two successive retransmissions is 0.2454129 seconds and TTL equals 27.

P	Nr	Dr	TTL
5264603	2	2454129	27

Figure 2. Example of individual

In accordance with the variation ranges of decision variables set in Table 2, the search space contains 12.10^{15} possible combinations (9.10^{16} in the rural area context). Simulate all possible cases with NS2 would take a lot of time. Hence, an EA is an appropriate tool for solving this problem.

The initial population is generated by choosing the values of each decision variable randomly in its given variation range. Each individual provides four input values for the NS2 simulator, which determines the mean values of the objectives NC, T, R and FR. These values are used by the EAs to compute the fitness of individuals using the principle of non-dominated sorting, based on Pareto dominance [3]. Pareto solutions are those for which improvement in one objective implies the worsening of at least one other objective. In the proposed approach, the EA uses the evaluation results returned by the NS2 simulator to build three Pareto fronts. First, all the individuals for which FR is less than a given threshold are said to be unfeasible. The remaining feasible individuals are compared, using Pareto dominance and all the non-dominated individuals are put in the first Pareto front R_1. This process is repeated successively twice with the remaining subset of individuals to build the second and the third Pareto fronts (R_2 and R_3). Finally, the remaining subset and the unfeasible individuals are gathered in the set of dominated solutions R_4. These sets are then used to compute the fitness in this way:

$$P(R_i)=\frac{\delta(R_i)*Card(R_i)}{\sum_{i=1}^{4}(\delta(R_i)*Card(R_i))},\forall\ i\ s.t.\quad 1\leq i\leq 4$$

where $Card(R_i)$ is the cardinality of R_i, and δ (R_i) is a fixed probability ratio between the dominated solutions and those of R_i.

This permits to define a roulette wheel in which the part of each front depends on its cardinality. Figure 3 gives an instance of such a roulette wheel built from the sets presented in table 1. This prevents the dominated solutions from having very low values of fitness and preserves the diversity of the successive populations.

Table1: Example of sets and associated selection probabilities

Sets	δ (R_i)	$Card(R_i)$	$P(R_i)$
R_1	4	19	0.5984252
R_2	3	8	0.1889764
R_3	2	5	0.0787402
R_4	1	17	0.1338583

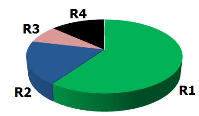

Figure 3. Example of roulette wheel.

This roulette wheel permits to select parents to be recombined. Each parent is chosen by two random selections. The Pareto front is first selected, then an individual is selected randomly among the individuals belonging to this front, using equal probabilities for all these individuals.

Each pair of selected parents is then recombined using a binary 10-point crossover. The four variables of each parent are first converted into binary strings and concatenated. The resulting binary string is then separated into several parts by choosing ten crossover points randomly. Then two children are built by alternatively copying the odd parts and the even parts of the selected parents. Then, the resulting binary string of each child is converted into four new integer values of parameters. This crossover operator has been chosen because k-point crossover is a classical method of recombination [7]. It has been tested with 1 point, but there was not enough diversity in the successive evolutionary populations. After several tests a 10-point crossover has been chosen because it seemed more efficient. However, additional experiments should be done by comparing various methods of crossing to determine which of them is the most suitable.

Finally, mutation chooses one of the four variables randomly and changes its value randomly. These operators permit to generate a new population containing n children, whose fitness is again estimated using NS2 simulations. All these steps (selection, reproduction, mutation, evaluation) are repeated until a given stop criterion. For the moment, since simulation requires quite a large computing time, this criterion is a given number of generations, in order to keep reasonable solving times. The EA finally returns the set R_1.

4. EXPERIMENTATION

4.1 Test environment

The simulations are done under the all-in-one distribution of NS2 running on a multiprocessor computer: four 2600 MHz AMD Opteron CPUs with 32 GB of Random Access Memory. For each individual, 500 independent NS2 iterations are run to obtain statistically reliable results. This number of iterations was determined empirically. Each iteration has a duration of 1000 simulated seconds, allowing the complete sending of one

emergency message on a dedicated channel, the processing and accounting of all the repetitions of this message that may be scheduled due to retransmission policy. The variation ranges of decision variables are given in table 2.
In the next sections, the TTL of each individual is not used in the interpretation because the size of the simulated networks is not high enough.

Table 2: variation ranges of decision variables

Parameter	P	Nr	Dr (in seconds)	TTL
Lower bound	0	1	0	10
Upper bound	1	5[1]	2	40

Table 3: configuration parameters of the EA

Parameter	Value	Parameter	Value
Size of population	32	Number of generations	15
Crossover rate	1	Mutation rate	0.02

The presented tests used the values of δ (R_i) given in table 1, the configuration parameters of the EA given in table3 and a feasibility threshold equal to 0.75. The population size and the number of generations are low to limit computation time. The other parameters were defined empirically.

4.2 Tests in a highway context

To illustrate vehicle communications on a highway, the chain topology presented on figure 4 has been set. 50 nodes were simulated, with a distance of 200 meters between two nodes. This illustrates cars lined up on 10 km with a medium density. Each vehicle is able to communicate regularly with a dozen peers, but more occasionally packets are received by vehicles up to a few thousands meters away (as observed in real experiments).

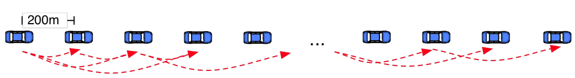

Figure 4 : The simple chain topology

The results are presented in figures 5 to 8. The EA provides a few dozen solutions belonging to R1 among the 12.10^{15} possible combinations. All these solutions are feasible (FR > 0.75). A number of unfeasible solutions that were not selected by the EA were taken into account to draw the provided figures. This permit to better observe the shape of the curves. The aim of this first step of the project was mainly to check that this approach could identify adapted settings for each given context. Therefore, only simple broadcasting methods were taken into account and the obtained results will not be compared with other known broadcasting methods. For some cases the simple flooding will simply be taken as reference. The comparison between complex strategies (and the associated settings) returned by this approach and classical broadcasting methods will be performed in the second step of the project.

Figure 5 presents the full reception ratio. It shows that retransmitting a packet only once, whatever the retransmission probability (P), leads to incomplete coverage. Out of 500 simulations for a given setting, the Nr=1 curve never goes over than 90% of FR. And this ratio quickly decreases with the decrease of P. In this scenario and for this criterion, individuals

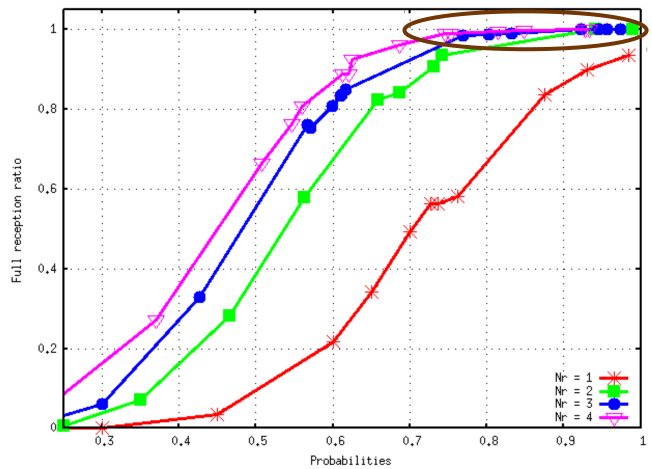

Figure 5: Full reception ratio for the medium density scenario

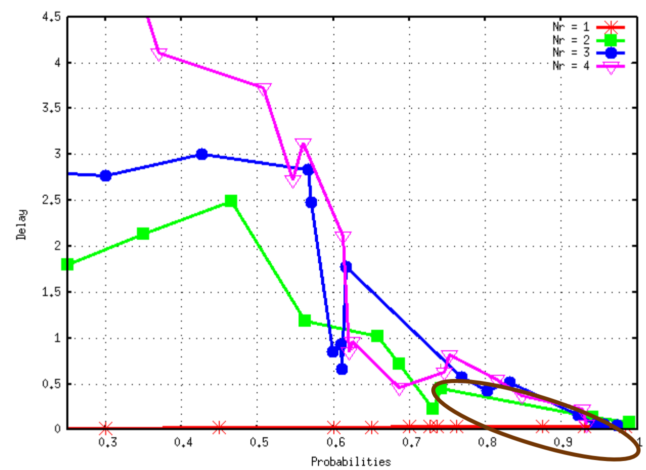

Figure 6: Delay for the medium density scenario

with Nr = 2; P > 0.9 and Nr ∈ {3,4}; P > 0.7 prevail (highlighted individuals in figures).

From figure 6, it appears that only individuals for which Nr=1 maintain a very short delay for any P. It also appears that for P>0.7, the other individuals behave essentially the same, increasing delays as P decreases.

Although this result was expected, it validates our approach and confirms its practicality.

Considering those two first criteria, only individuals for which Nr = 2; P > 0.9 and Nr ∈ {3,4}; P > 0.7 are selected, as shown with still highlighted individuals on figure 6. The figure 7 leads to eliminate individuals for which Nr ∈ {3,4}, as they cause to much collisions.

Finally, figure 8 shows a cost expressed as a number of sent packets, and leads to select the individuals that have the lowest cost (R) which are Nr=2 and P > 0.9. The observed delay time between repeats (Dr) for these individuals is in [0.7 ; 1.2]. For this context, it is clear that the simple flooding (P=1 and Nr=1) will be efficient since it allows only one retransmission for each packet.

[1] Except for the rural area experiment where the upper bound is 30.

4.3 Tests in an urban context

Here the focus is on the increased density of vehicles. The used topology is similar to the previous chain, but with 134 nodes for the same 10-km length (one VANET capable node every 75 meters). Each message may be received by tenths of nodes. The results are presented in figures 9 to 12.

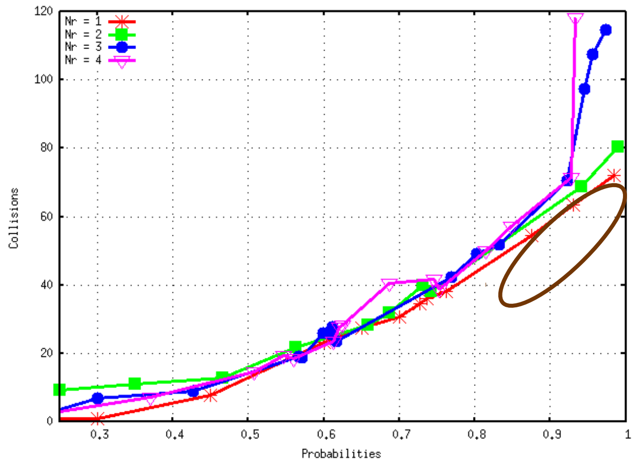

Figure 7: Collisions for the medium density scenario

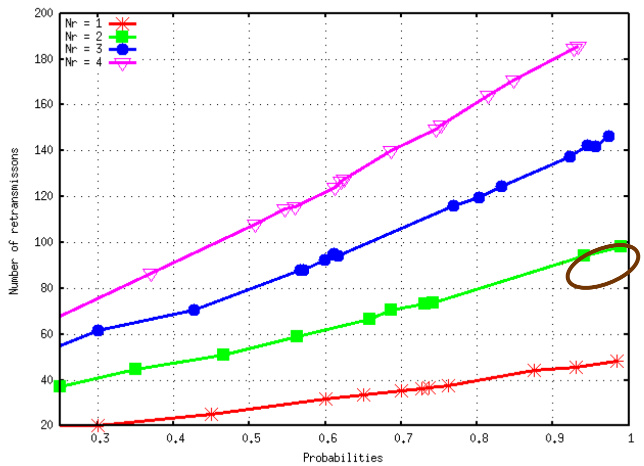

Figure 8: Number of retransmissions for the medium density scenario

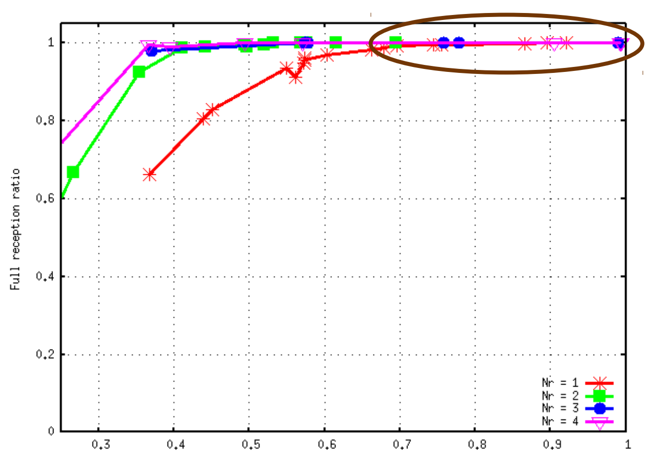

Figure 9: Full reception ratio for the high density scenario

As far as full reception ratio is concerned (see figure 9), all individuals are equivalent if P is high enough (P>0.7). Concerning delay, the individuals all behave equally well as long as P >0.55 (see figure 10). The still selected individuals set is thus not reduced. Figure 11 shows that all individuals behave also essentially the same for a given P.

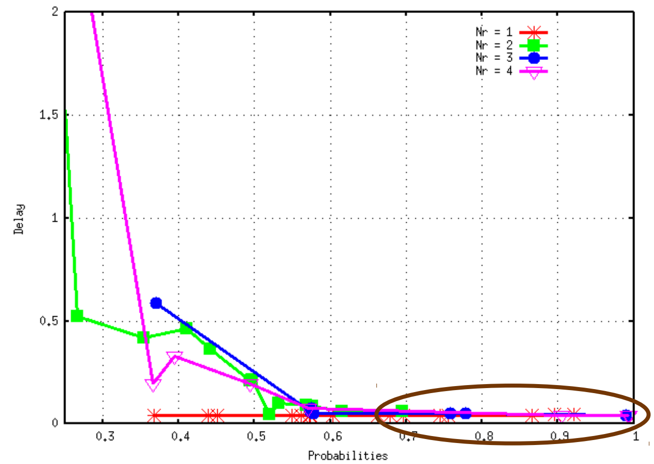

Figure 10: Delay for the high density scenario

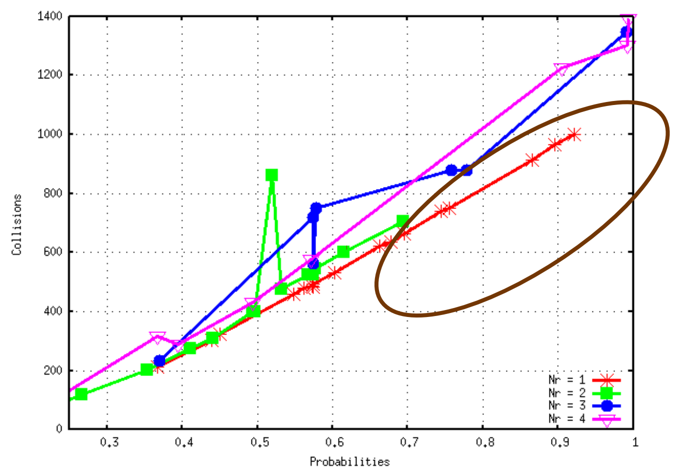

Figure 11: Collisions for the high density scenario

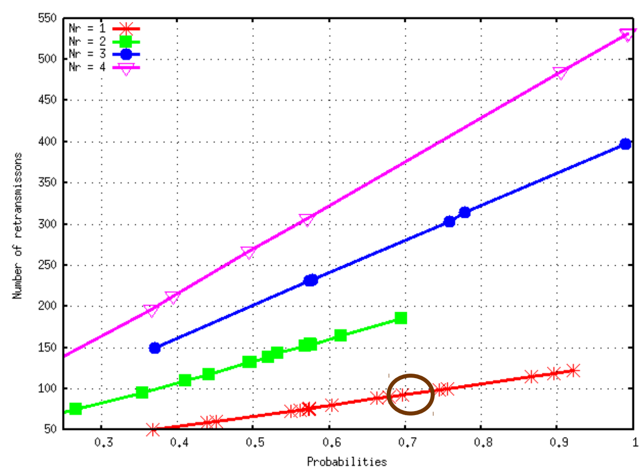

Figure 12: Number of retransmissions for the high density scenario

Figure 12 finally enables to discern an optimal setting. The best individuals correspond to Nr=1 (as they have the lowest channel occupation) and P=0.7, which is the minimum below which information would not always reach all the nodes. The Dr time has no sense here because when Nr = 1 there is no second repeat on nodes.

The simple flooding will unnecessarily consume the bandwidth if it is used in this context.

4.4 Tests in a mobile node context

The topology used in this scenario is derived from the previous one, as shown in figure 13. There are two chains of 67 vehicles, each driving in opposite directions. The results are not shown, as they are very similar to those presented in figures 9 to 12. From the packet diffusion protocol point of view, this topology and the previous one are very similar. In such dense scenarios, the vehicle density largely prevails over mobility patterns, i.e.: when the matters less average number of neighbors is high how they move.

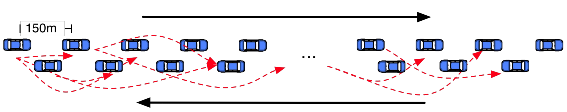

Figure 13: The double chains topology

4.5 Tests in a rural area

This scenario focuses on a very low-density topology where vehicles are often too far away from each other to communicate. The topology is basically similar to the one presented in figure 4. But in order to simulate a scarcer vehicle distribution, the radio propagation model mimics the very intermittent presence of neighbors: a given vehicle can communicate only for periods accounting for about 20% of the total simulation time. Figure 8 shows that the EA selected only individuals with high Nr values. This is understandable as only high redundancy enables reliable enough hop-to-hop communications in such a scarce network.

Figure 14 shows that for an FR value close to 1, Nr > 15 and P > 0.8 are necessary. This appears in figure 15 as a high overall number of retransmissions. Hopefully this network overhead is of course distributed over a longer period and a large area (tenths of seconds and 10km line in this scenario). So the collision ratio remains very low (around an average of 20 collisions for the selected individuals). In such an environment, as information can only occasionally have the opportunity to jump from one vehicle to the next one, the delay before every vehicle has been reached is quite high, in the order of tenths of seconds, because the Dr time is in [0.3;2].

Due to the fact that the message should be retransmitted many times, the simple flooding is not applicable in this context.

5. CONCLUSION

The goal of this work was to emphasize efficient parameters in the inter-vehicle communication context. However, the number of potential values for this communication context is huge. The identification of right parameters for this type of ad hoc network is extremely complex using an analytical approach or a classical simulation approach. The originality of our work is to use an evolutionary algorithm (EA) to track local optima which offer good properties. Based on the results developed in the previous section, we highlighted the need for an adaptive strategy to design broadcast strategies. The results have enabled us to measure the

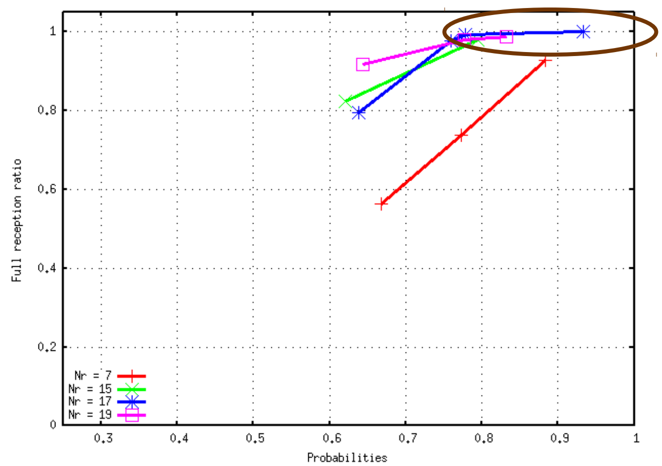

Figure 14: Full reception ratio for a low density scenario

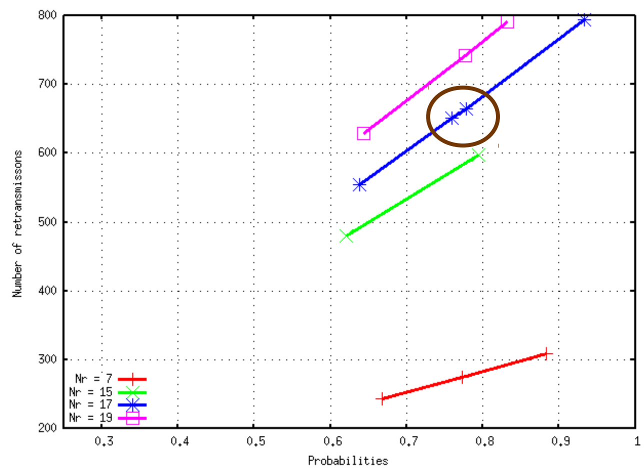

Figure 15: Number of retransmissions for a low density scenario

differences between an appropriate strategy and an inefficient strategy in a particular context. Even if some results were expected, they validate the proposed approach and confirms its practicality. Now, we know the effect of input parameters on the efficiency of the broadcast function linked to the environment. The next step is to integrate these results into an adaptive strategy. However, we must design mechanisms as implicit as possible to assess the vehicle environment. One of the challenges is to identify the density of neighbors by transmitting as few messages as possible. The second main prospect concerns the EA. Its improvement will mainly focus on the development of an adaptive distributed version of EAs, to facilitate the configuration phase and to reduce computation time. This step will rest on the adaptation of concepts we already used in other application fields [9]. A comparative study of crossover methods must also be done. The goal is to increase performance. More complex and realistic network topologies (such as grid topologies) could be considered. Simulating others communication strategies, such as location-based methods and various classes of messages to take into account priorities is possible.

6. REFERENCES

[1] Lee Breslau, Deborah Estrin, Kevin Fall, Sally Floyd, John Heidemann, Ahmed Helmy, Polly Huang, Steven McCanne, Kannan Varadhan, Ya Xu, and Haobo Yu. Advances in Network Simulation. IEEE Computer, 33 (5), pp. 59-67, May, 2000. Superceeds USC tech report 99-702b.

[2] Coello Coello, C. A.; Lamont, G. B. & Van Veldhuizen, D. A. Evolutionary Algorithms for Solving Multi-Objective Problems Springer, 2007.

[3] Y. Collette and P. Siarry, *Optimisation multiobjectif*. Eyrolles, 2002,

[4] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II." *IEEE Trans. Evolutionary Computation*, vol. 6, iss. 2, pp. 182–197, 2002.

[5] Dhoutaut, D., Régis, A., and Spies, F. 2006. Impact of radio propagation models in vehicular ad hoc networks simulations. In Proceedings of the 3rd international Workshop on Vehicular Ad Hoc Networks, Los Angeles, CA, USA, September 29 - 29, 2006. VANET '06. ACM, New York, NY, 40-49

[6] Stephan Eichler, Christoph Schroth, Timo Kosch, Markus Strassberger, Strategies for Context-Adaptive Message Dissemination in Vehicular Ad Hoc Networks in IEEE Computer Society, 2006

[7] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Professional, 1989

[8] L. Li, J. Halpern, and Z. Haas. Gossip-based ad hoc routing. In Proceedings of the IEEE INFOCOM, IEEE Computer Society, 2002.

[9] J.-L. Hippolyte, C. Bloch, P. Chatonnay, C. Espanet, and D. Chamagne. A self-adaptive multiagent evolutionary algorithm for electrical machine design. In Dirk Thierens et al., editor, *GECCO'07: Proceedings of the 9th annual conference on Genetic and evolutionary computation*, volume II, pages 1250–1255, New York, NY, USA, 2007. ACM Press.

[10] A. Liefooghe, M. Basseur, L. Jourdan, E.-G. Talbi: ParadisEO-MOEO: A Framework for Evolutionary Multi-objective Optimization. EMO 2006, pp. 386-400

[11] Tseng, Y., Ni, S., Chen, Y., and Sheu, J. 2002. The broadcast storm problem in a mobile ad hoc network. Wireless Network 8, 2/3 (Mar. 2002), 153-167

[12] Williams, B. and Camp, T. 2002. Comparison of broadcasting techniques for mobile ad hoc networks. In Proceedings of the 3rd ACM international Symposium on Mobile Ad Hoc Networking & Computing, Lausanne, Switzerland, June 09 - 11, 2002. MobiHoc '02. ACM, New York, NY, 194-205.

[13] Qi Zhang, Dharma P. Agrawal, Dynamic probabilistic broadcasting in MANETs, Journal of Parallel and Distributed Computing, Vol. 65, Issue 2, February 2005, Pages 220-233.