SUMMARIZING SENSORS DATA IN VEHICULAR \textit{AD HOC} NETWORKS

DORSAF ZEKRI$^1$, BRUNO DEFUDE$^1$ AND THIERRY DELOT$^2$

Abstract. This article focuses on data aggregation in vehicular \textit{ad hoc} networks. In such networks, sensor data are usually produced and exchanged between vehicles in order to warn or inform the drivers when an event is detected \textit{(e.g., accident, emergency braking, parking space released, vehicle with non-functioning brake lights, etc.)}. In the following, we present a solution to aggregate and store these data in order to have a history of past events. We therefore use Flajolet-Martin sketches. Our goal is to generate additional knowledge to assist drivers by providing them useful information even if no event is transmitted by vehicles in the vicinity.

Keywords. Vehicular \textit{ad hoc} networks (VANET), event streams, sensor data, spatio-temporal data, data aggregation.

Résumé. Dans cet article, nous nous intéressons à l’agrégation de données dans les réseaux \textit{ad hoc} inter-véhicules. Dans ces réseaux, des données sont produites par des capteurs et échangées entre véhicules pour informer ou avertir les conducteurs lorsqu’un événement survient \textit{(e.g., accident, freinage d’urgence, place de stationnement libérée, un véhicule avec les feux stop défectueux, etc.)}. Dans la suite, nous proposons d’agréger et de stocker ces données afin de conserver un historique des événements précédemment observés. Nous utilisons pour ce faire des sketches Flajolet-Martin. Notre objectif est ici de générer des connaissances additionnelles afin d’assister les conducteurs en leur proposant des informations pertinentes, y compris lorsqu’aucun événement n’est communiqué par les véhicules à proximité.
1. Introduction

Inter-vehicle communication is a recent topic of research and many interesting contributions have already been proposed, particularly concerning the exchange of information between vehicles [4,14,17,20–22,29]. In Vehicular ad hoc networks (VANETs), inter-vehicle communications rely on the use of short-range networks (about a hundred meters), like IEEE 802.11 or Ultra Wide Band (UWB) standards, providing bandwidth in the range of Mbps [19]. Using such communication networks, drivers can receive information from neighboring vehicles.

Our work takes place in the VESPA project1 [9], a system designed for vehicles to share information in inter-vehicle ad-hoc networks. The originality of VESPA is to support the exchange of many types of event in the network (e.g., available parking space, accident, emergency braking, obstacle in the road, real-time traffic information, information relative to the coordination of vehicles in emergency situations, etc.). Therefore, VESPA proposes a dissemination protocol based on the concept of Encounter Probability (EP), used to estimate the relevance of events for vehicles [4]. VESPA is thus complementary to existing navigation systems supporting only static information such as points of interest, since it allows drivers to be informed of ephemeral events occurring on the roads.

In VESPA as well as in the other existing systems, messages representing events (e.g., traffic congestion, emergency braking, parking space released, etc.) are generated and exchanged between vehicles using various protocols, in order to warn or inform drivers. Data is here only considered as an object to transmit and deleted once used. Our contribution in this article consists in collecting, summarizing and storing data about observed events on each vehicle. The concept of data summarization in inter-vehicular ad hoc networks has been considered in many works as a simple method of compressing data to reduce bandwidth requirements. Our approach is quite different since our goal is here not only to produce instantaneous warnings to the driver, but also to extract environmental knowledge usable to provide relevant information to the driver. For example, a summary of available parking spaces previously observed can be used to define the area having the highest probability of finding free places at a given day and hour. In another context, thanks to the correlation of safety related messages received by a vehicle (e.g., accident, emergency braking, etc.), dangerous areas can be dynamically detected and indicated to the driver. Such an approach can be applied not only to the detection of permanently dangerous areas but also to temporarily ones due to

---

1For more information, see http://www.univ-valenciennes.fr/ROI/SID/tdelot/vespa/
bad weather conditions for example. Different spatio-temporal aggregation techniques have been proposed in the literature [18], both in the contexts of vehicular networks and wireless sensor networks (WSN). In a preliminary work [8], we have specified a first aggregation structure based on a simple count of events in a spatial temporal cell. This article extends this work by proposing a new data structure that is more efficient and able to avoid counting several times the same occurrence of events observed by two different vehicles. Moreover, in order to increase the environmental knowledge exploited by each vehicle, we introduce mechanisms that allow vehicles to exchange the data they collected. Therefore, a two levels spatio-temporal model is proposed to simplify the exchange algorithms by creating a common repository for all vehicles. Hence, vehicles do not lose their autonomy while choosing their interest areas. Finally, each driver can improve the performances of the exchange process by establishing his/her priorities in terms of data to collect. The remainder of this article is organized as follows. Section 2 presents our overall vision. Section 3 introduces our spatio-temporal model. Section 4 describes the proposed aggregation structure. Section 5 presents the principle of our exchange protocol as well as an evaluation of its performances. In Section 6, we compare our approach with related works. Finally, we conclude and present the perspectives of our work in Section 7.

2. Global architecture

In the following, we consider smart vehicles able to provide alert services and decision support to drivers. Thus, one vehicle \( i \) (see Fig. 1) can acquire information about events observed either by itself (via sensors for example) or diffused by other vehicles (using a dissemination protocol). Obviously, the information available on one vehicle is partial since that vehicle cannot perceive all events or receive all messages transmitted by other vehicles using short-range wireless networks. Vehicles can also acquire information from a fixed infrastructure deployed along roads. For example, in urban areas, the infrastructure may correspond to a central parking management or central traffic information server providing information to vehicles driving in its vicinity.

Usually, events broadcasted in the vehicular network have a quite short lifetime, ranging from a few seconds to several hours depending on the type of event. Table 1 presents an example of message exchanged among vehicles to advertise an available parking space. This message contains a unique identifier, a priority, the

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Priority</th>
<th>Position</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>27092010102517191591N305111EABCD</td>
<td>low</td>
<td>50°19’15.91 N 3°30’51.11 E</td>
<td>Available parking space</td>
</tr>
</tbody>
</table>
position and a description of the resource considered. Thanks to one of the existing dissemination protocols, this message is transmitted to the vehicles driving in the vicinity of the parking place during a limited period of time \cite{4,29}. The solution presented in this article does not depend on the dissemination algorithm used. We only assume that the diffusion of messages among vehicles does not rely on any central server but rather relies on direct exchanges between close vehicles. Moreover, to avoid losing information related to the events observed (e.g., the available parking space), we propose in this article to aggregate events considered obsolete (i.e., previously observed and possibly used to produce a warning) in order to keep a summary to estimate whether an event can happen even without any further observation. Thus, when many accidents are observed in a particular geographical area, it is possible to conclude that this area is dangerous enough to warn drivers, even if no accident has been signaled by a neighboring at this time.

Data aggregation is defined by \cite{16} as a technique used to overcome two problems: implosion (data sensed by one node is duplicated in the network due to data routing strategy) and overlap (two different nodes disseminate the same data). Therefore, aggregation functions have to be duplicates insensitive (see Sect. 4.1 for a detailed discussion).

An alarm management module or decision support system for drivers can benefit from events observed by the vehicle (or others), from information delivered by an infrastructure and from summaries built on the vehicles (or exchanged with others). Obviously, the confidence in the information is also an important parameter which may change since the summaries do not contain real but probabilistic information. For instance, the enhancement/reduction of the confidence value affected to a summary may depend on the drivers’ feedback.

FIGURE 1. Global architecture.
3. Two levels spatio-temporal model

In our approach, each car has to save a summary of previously observed events. Our solution is based on a simple aggregation of many events depending on both the spatial and temporal dimensions. Based on the needs expressed in Section 2, we not only need a spatio-temporal model allowing everyone to choose his/her own interest area but also a suitable model for the exchanges of (parts of) summaries between vehicles minimizing the loss of information. For instance, we should not have to split an area because the number of events observed should then be distributed between the different parts of this area, what leads to an increase of the imprecision.

To address these needs, we propose the spatio-temporal model illustrated in Figure 2. This model is composed of two parts:

- The physical level: this lowest level consists in a repository shared between all vehicles which goal is to allow information exchanges without loss. The physical level is divided into fixed size squares that form a full partition. The same idea is used for the temporal dimension. Time is so divided into segments that form a full partition. We assume here that we want to emphasize the seasonal nature of the event production. We therefore propose to split the time in 7 days, themselves sliced in 2 h segments, providing a total of 84 time segments. The couple \{square, time segment\} is the smallest unit that can count occurrences of events. This physical space is very large: assuming that the size of a cell space is 1 km\(^2\) and 10 time segments, the coverage of France would represent about 6 million pairs. This number could be reduced by structuring the space using unfixed size...
areas, which allow having a better spatial resolution in urban areas (and greater accuracy). However, this requires a little more complex algorithm to implement.

- The logical level: Based on this physical level, each vehicle can build its own logical splitting, defined as a set of rectangles (or intervals). Those rectangles are themselves sets of squares (or intervals) of the physical layer (the rectangles with dotted lines in Fig. 2). Indeed, a driver may not be interested in the whole space but only in a subset. We impose that a physical cell belongs at most to one interest logical area. The number of squares (intervals) actually observed at the logical level is so (much) smaller than the whole physical level. For example, if a driver wants to monitor a hundred of spatial areas covering an average surface of 20 km², the number of couples to consider is approximatively equal to 2000.

4. Summarization of spatio-temporal events

The aim of our research consists in using past collected data to estimate the probability of occurrence of an event in the absence of fresh observation. The definition of the summarization process in our work is to aggregate past events to provide a knowledge base to estimate whether an event might occur even without observation. There are a variety of techniques that can be used to build summaries of spatio-temporal events. The important criteria to be provided by a summary are:

1. promote basic dimensions such as location and time;
2. be incrementally constructible and inexpensive in both computing time and storage space;
3. let each driver define the types of events s/he is interesting in, as well as the spatial and temporal scales s/he wants to use for the aggregation process;
4. allow the exchange of (parts of) summaries between vehicles in order to enrich their knowledge base.

The third criterion is provided by our two levels spatio-temporal structure. The first criterion requires a compact representation. The last criterion implies that the aggregation mechanism detects duplicates. Therefore, it has to recognize when the same event has been observed by two different vehicles in order not to consider it as two different events to aggregate.

Sketches are the basis of our proposal and are described in Section 4.1. Then, we detail our proposed data structure and its theoretical evaluation.

4.1. Sketches

The Flajolet-Martin Sketch [12] provides a compact representation to estimate the number of occurrences of distinct events. The sketch contains a set of binary arrays initially filled with 0. The size of the sketch is defined according to the size
of the set (greater the chain, better the accuracy of the estimate). A hash function \( h \) is then applied to each element \( x \) of the set. Let \( pf\,p(h(x)) \) be the lowest weight position of 1 in the binary representation of \( x \). The bit index \( pf\,p(h(x)) \) is set to 1 in the sketch if it is still at 0. Once the sketch has been constructed, the number of distinct values \( p \) can be estimated by the lowest weight position of 0 in the binary table using the estimate function \( E(p) = \log_2(\phi n) \) in [12].

Such a sketch is interesting in the sense that it detects duplicate by construction. Indeed, two instances of the same event have the same image computed by the hash function. Figure 3 illustrates the insertion of an event \( X1 \) in the sketch. \( X1 \) may correspond to a free parking space or an accident for example.

This sketch has been used in [28] which proposes a method for spatio-temporal indexing based on a R-tree for the spatial part and a B-tree for the time part. The value stored in a tree cell is a sketch and not a simple integer. This structure can properly be applied in our case, but can be simplified in our particular context of regular split of space and time.

4.2. Proposed data structure

We assume that each vehicle \( V_i \) can observe a set of events \( E \). An event \( e \) of \( E \) is characterized by (these are only considered in the summary, but other information can be useful for managing alarms or dissemination of messages):

1. \( ty_e \): the type of observed event (e.g., accident, release parking...).

2. \( lo_e \): the location of the event and its timestamp. This information is provided by GPS like positioning systems.

3. \( idf_e \): the unique identifier of event \( e \). This unique identifier is the basis for the detection of duplicates. We assume that an instance of an event always produces the same identifier on the vehicle \( V_i \) and other vehicles. Such a unique identifier can be generated by combining the current time and the GPS location of the event with a randomly-generated sequence\(^2\).

\(^2\) The generation of a unique identifier for events “observed” by several vehicles (e.g., different vehicles stuck in a traffic congestion) is still an open problem. Interesting ideas to solve it have been proposed in the field of information fusion [11,15].
In the following, we assume the following notations. The physical space is divided into squares of size $C_{NP}$ ($N$ squares on the $X$ axis and $P$ ones on the $Y$ axis). The coordinates of the origin point are $(x_{\text{origin}}, y_{\text{origin}})$. An interest area is defined as a rectangle defined by a number of physical cells. Areas can be disjoint or overlap. An interest area is defined by a pair of physical cells: bottom left of coordinates $(i, j)$ and upper right of coordinates $(i, l)$. We assume $g$ temporal granularities (ordered from Monday from 0:00 AM to 2:00 AM to Friday 10:00 PM to 12:00 PM). We describe a summary with the data structure described in Figure 4. It consists of a set of interest areas defined by a unique identifier ($idf$), two physical cells $(i, j, k, l)$ and aggregates many types of events (boolean $b$ indicates if the type of the event is aggregated in this area and $p$ is a pointer to a linked list of sketches). The table of interest areas is sorted by increasing values of $i$.

Each element of the linked list of sketches is characterized by a physical cell $(i, j)$ and by $g$ sketches aggregating the events observed in the cell for a given temporal granularity. The list has as many elements as there are physical cells in the interest area. The sketch is constructed using a hash function applied on the $idf$ identifier of each event. The following operations are defined on this data structure:

1. **Add an interest area**: create a new entry in the table of areas.

2. **Remove an interest area**: remove the corresponding entry in both the table of areas and the associated linked lists.

3. **Add an event**: the area(s) of interest corresponding to the location of the event have first to be identified. Then, if there is (at least) one, the associated physical cell and the corresponding sketch have to be updated.
These three operations are local to a vehicle. An intersection operation between two areas of interest is also needed to exchange summaries between two vehicles (see Sect. 5).

4.3. Theoretical evaluation

Let us assume that a vehicle observed $P$ interest areas containing $M$ physical cells, with $K$ types of aggregated events over all temporal granularities. The summary then requires the following memory space:

\[
\text{Size} = P \times (5 \text{ bytes} + K \text{ bits} + K \text{ pointers}) + KM \times (2 \text{ bytes} + 1 \text{ pointer} + g \text{ sketches})
\]

If $P = 100$, $M = 100$, $K = 8$, pointersize = 4 bytes, sketch = 800 bits, $g = 84$

Then Size $= 100 \times (5 + 1 + 32) + 5 \times 100(2 + 6 + 84 \times 100)$
So Size $= 4207800$ bytes or 4.2 Mbytes.

The temporal granularity $g$ is an important factor of the size when 84 time segments (portions of 2 h) are used. If we reduce the number of segments to 7 (day granularity), the size drops to about 0.5 Mbytes. The access cost to the data structure for a specific physical cell is linear in the number of area and in type of event: $O(P + K \times M)$.

5. Inter vehicles exchange protocol

In the previous section, we have introduced an aggregation structure that can be used to store a set of events observed by each vehicle. In the following, we focus on the (partial) exchange of summaries built on the vehicles in order to enrich the local database of each vehicle that can be used to extract information for the driver.

5.1. Principle of the exchange protocol

Each vehicle decides to publish all or part of its summaries to other vehicles and can also be interested in all or part of the summaries of others. To simplify, we consider here only public publications and subscriptions (one publishes/subscribes to all the vehicles it is likely to meet). The publication process consists in defining which summaries should be published (possibly aggregating them by grouping cells).

The subscription process consists in defining filters specifying the events types that the driver is interested about, adding appropriate spatial and temporal criteria. For example, a driver can be interested by “accidents” in “Paris” over the last month. The exchange of information between vehicles can then be done through a relay (e.g. servers located along the roads), or directly. In both cases, the exchange process is unsure if the duration of the connection is not sufficient to allow the complete exchange of summaries. We therefore propose to use a mechanism based on priorities, which defines an order based on data utility, and use this order
to prioritize exchanges. Priorities are defined as a set of rules defining an order between several elements. We use as elements the various types of events, different time granularities for the temporal dimension and the different areas of interest to address the spatial dimension.

The following example defines exchange priorities of a vehicle $V_i$. The following expression describes the types of events $V_i$ is interested about (accident first, then available parking space, and the other types of event do not interest it):

\[(\text{Exp 1}) \text{ Accident} > \text{Parking.}\]

If we assume 10 temporal granularities ($g_1$ representing the most recent period and $g_{10}$ the earliest), the following expression indicates that $V_i$ is systematically interested in the most recent summary ($g_6$ to $g_{10}$ will not be considered for exchange):

\[(\text{Exp 2}) \ g_1 > g_2 > g_3 > g_4 > g_5.\]

Similarly, if we assume 10 areas of interest for $V_i$, the next expression defines an order between them:

\[(\text{Exp 3}) \ A_1 > A_3; \ A_2 > A_4; \ A_4 > A_6; \ A_6 > A_8.\]

In this example, we have a partial order with $A_1$ and $A_2$ which are prior areas, then $A_3$ and $A_4$ then $A_6$ and finally $A_8$. Non-mentioned areas are not affected by the exchange. \textit{Exp1}, \textit{Exp2} and \textit{Exp3} define the priorities to follow when vehicle $V_i$ receives data from another vehicle. When $V_i$ meets $V_j$ and needs to obtain new summaries, it starts sending information about its priorities. Then, $V_j$ calculates the intersection among its summaries and sent priorities. If that intersection is not empty, it sends data corresponding to requested priorities. Depending on the duration of the connection, all or part of the exchange will be realized.

The basic operation here is the intersection between two interest areas (one for each vehicle). This intersection returns either the empty set if the two areas are distinct or corresponds to a set of physical cells if they have an intersection. For these common cells, the result is just the “inclusive OR” of sketches. The cost of the calculation of intersection is logarithmic in the number of areas (to determine the $p$ intersecting areas) and linear in number of physical cells: $O(\log P + p \times M)$.

Obviously, $V_i$ should not exchange continuously with the same vehicles. Therefore, we store on each vehicle a list containing the identifiers of $N$ latest vehicles with which an exchange took place as well as their timestamps. Before initiating the exchange of summary with a vehicle, the system has so to verify that the identifier of the encountered vehicle does not already appear in this list.

Another problem to avoid in the exchange phase is the one of duplicates (i.e., counting several times the same event occurrences). This problem is solved by applying a hash function to the key of the events. Indeed, if two vehicles $V_i$ and $V_j$ observe the same occurrence of an event $idf_e$, the same hash function $h$ is applied
on both vehicles. Thus, $h_{V_i}(id_f_e) = h_{V_j}(id_f_e)$ and the use of the “inclusive OR” only retains one occurrence in the exchange of sketches.

The exchange process is modeled in Figure 5. In Figure 5a, we consider 6 vehicles close enough to communicate, knowing that the exchange will take place successively in times ($t_1 < t_2 < ... < t_6$). A directed edge between vehicles $V_i$ and $V_j$ describes the fact that $V_i$’s summary has been updated thanks to $V_j$’s sketches.

The exchange process between vehicles $V_i$ and $V_j$, where $V_i$ is the sender and $V_j$ the receiver, is composed by two steps described in the following in Figure 5b:

- **Step 1**: $V_j$ sends its priorities to $V_i$. $V_i$ compares $V_i$’s priorities with its own sketches and produces an ordered list of sketches to exchange. This implies to transform the partial order defined by priorities in a total one (for the space dimension we give priority to the areas which are closed to the current one and we favor the most recent ones for the time dimension).
- **Step 2**: This phase just consists in the exchange of the sketches selected according to the order previously computed and in merging the couples of sketches $(CP_i(V_i), CP_j(V_j))$ selected in Step 1 using an “inclusive OR”. If the connection time is sufficiently important, all selected sketches are exchanged. Otherwise, only the preferred sketches are exchanged.

Figure 6 illustrates the exchange between $V_1$ and $V_3$ at step $t_1$ and represented in Figure 5a. Let us consider that:

- $V_1$ holds a summary $S_1$ and is interested in two types of event (Accident and Available parking space) at $t_0$ in two logical areas $A_1$ and $A_2$, each one composed by two physical cells $A_1 (c_{11}, c_{12})$ and $A_2 (c_{21}, c_{22})$.
- $V_3$ holds a summary $S_3$ interested in three types of event (Accident, Available parking space and traffic congestion) at $t_0$ in two logical areas $A_3$ and $A_4$, each one composed by two physical cells $A_3 (c_{31}, c_{32})$ and $A_4$. 

![Figure 5. Exchange principle.](image)
These vehicles also have priorities about the types of events, the spatial zone (interest area) and the time windows (temporal granularity) they want to monitor. These priorities for $V_1$ and $V_3$ are expressed as follows:

$$V_1's\ priorities\ \left\{\begin{array}{c}
\text{Accident} > \text{Available parking space} \\
g_1 > g_2 > \ldots > g_{12} \\
A_2 > A_1
\end{array}\right.$$
SUMMARIZING SENSORS DATA IN VEHICULAR AD HOC NETWORKS

As shown in Figure 5a, $V_1$ initiates the exchange of summaries with $V_3$ at time $t_1$. In a first step, $V_1$ and $V_3$ exchange their respective priorities. Then, $V_3$ finds a match between its summaries and $V_1$’s priorities. The temporal granularities are indeed the same on both vehicles and the types of events required by $V_1$ (e.g., accident and available parking space) are also stored on $V_3$. Moreover, there is an intersection between $V_1$’s areas of interest ($A_1$ and $A_2$) and $V_3$’s ones ($A_3$ and $A_4$).

As shown in Figure 6a, $V_3$ identifies a single physical cell in common with $V_1$ since $A_1 \cap A_3 = C_{22} = C_{42}$. Then, $V_3$ identifies the sketches to exchange (i.e., those associated to either accident or available parking place for all time periods) and corresponding to cell $C_{42}$ (Fig. 6b). At the same time $V_3$ compares its priorities with those of $V_1$ but there is no match here since they are not interested in same types of event and there is no intersection between $A_1$ and $A_2$ on $V_1$ and $A_4$ on $V_3$.

In the second and final step, $V_3$ sends the selected sketches in a predefined order to $V_1$ (e.g. first accident and then available parking place). Then, a merging operation with an “exclusive or” is performed locally on $V_1$. The result of this operation is presented in Figure 6c. So, at $t_1 + \Delta t$ the common physical cell summary on $V_1$ changes from Sketches ($V_1$, $A_2$, $C_{22}$) to Sketches ($V_1$, $A_2$, $C_{22}$) $\oplus$ Sketches ($V_3$, $A_4$, $C_{42}$).

To generalize, we represent the sequence of summaries’ exchanges in Figure 7. A cell $(i, j)$ in the table contains the value summarized on vehicle $V_i$ at time $t_j$. This illustrates that exchanges improve the completeness of vehicles’ summary. For instance, $V_4$ improves its initial summary $S_4$ by merging the values of $S_3$, $S_5$ and $S_6$. The summary of $V_4$ and $V_5$ at $t_4$ changes from $S_4$ and $S_5$ respectively to $S_4 \oplus S_5$ and $S_5 \ominus S_4$ as shown in Figure 7. Let us note also that the exchange process can be bidirectional. This is illustrated by the two edges between $V_4$ and $V_5$ in Figure 5.

<table>
<thead>
<tr>
<th>$V_i$</th>
<th>$V_1$</th>
<th>$V_2$</th>
<th>$V_3$</th>
<th>$V_4$</th>
<th>$V_5$</th>
<th>$V_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_1$</td>
<td>$R_1$</td>
<td>$R_1$</td>
<td>$R_2$</td>
<td>$R_2$</td>
<td>$R_3$</td>
<td>$R_4$</td>
</tr>
<tr>
<td>$t_2$</td>
<td>$R_4 \ominus R_2$</td>
<td>$R_4 \ominus R_2$</td>
<td>$R_3 \ominus R_2$</td>
<td>$R_3 \ominus R_2$</td>
<td>$R_3 \ominus R_2$</td>
<td>$R_3 \ominus R_2$</td>
</tr>
<tr>
<td>$t_3$</td>
<td>$R_4 \ominus R_2$</td>
<td>$R_4 \ominus R_2$</td>
<td>$R_3 \ominus R_2$</td>
<td>$R_3 \ominus R_2$</td>
<td>$R_3 \ominus R_2$</td>
<td>$R_3 \ominus R_2$</td>
</tr>
</tbody>
</table>

$V_3$’s priorities

$g_1 > g_2 > \ldots > g_{12}$

$A_3$.  

Figure 7. Temporal representation of summary’s exchanges.
5.2. Dynamic of the exchange protocol

In this section, our goal is to evaluate our exchange protocol. Therefore, we study how a summary produced by a vehicle is disseminated among the set of vehicles. The following formula approximates the number of reached vehicles after \( i \) rounds of exchanges:

\[
U_i = \sum_{i=1}^{n} U_{i-1} \times N\alpha\beta - \gamma \times \left( \sum_{i=1}^{n} U_{i-1} \times N\alpha\beta \right)
\]

where

- \( N \) is the total number of vehicles in the environment;
- \( \alpha \) is the percentage of vehicles encountered;
- \( \beta \) is percentage of vehicles interested in the summary;
- \( \gamma \) is percentage of redundant exchange (\( \gamma \) incremented after each round);
- \( U_{i-1} \) is the number of vehicles that received information at round \( i - 1 \);
- \( U_0 = U_1 = 0 \).

In this model we formalize the dissemination of summaries as an iterative process. At each round, vehicles having a specific summary \( S \) exchange it with a constant number of vehicles defined by \( \alpha\beta N \). To deal with redundancies (a vehicle may get summary \( S \) from different vehicles), we introduce \( \gamma \) as a redundancy factor decreasing \( U_i \). Using this formula, we highlight the impact of two factors:

1. \( \alpha\beta \), the percentage of vehicles having the summary;
2. \( \gamma \), the percentage of redundant exchanges.

The result of this variation is represented in a logarithmic axis in Figures 8 and 9. In the following, we consider that the target of a summary is 80% of \( N \) (fixed to 8000 vehicles). Besides, each round corresponds to one week. Figure 8 shows the impact of the \( \alpha\beta \) parameter with the same redundancy factor \( \gamma \) (initially set to 20% and increases by 10% ). We note that the number of reached vehicles increases exponentially (linear in logarithmic axis). Furthermore, the higher value of \( \alpha\beta \), the lowest the number of rounds needed to reach interested vehicles. As shown in Figure 8, for \( N = 10000 \), all interested vehicles can be reached in 4 rounds (including low value of \( \alpha\beta = 0.1\% \)). The impact of \( \gamma \) is shown in Figure 9. We therefore set \( \alpha\beta \) as constant and study two different values for the \( \gamma \) parameter. In the first case, we choose a low value for this parameter which is initially set to 20% and increased by 10% after each round. In the second one, a higher percentage is used for \( \gamma \) (initially set to 30% and increased by 20% after each). The last scenario may correspond to the case of a parking lot where many vehicles are searching for a free space. The probability that a vehicle encounters several times a same vehicle is then high what may lead to a lot of redundancy. In Figure 9, we note that in
such configurations, the increase in the number of rounds is less important than the increase of the $\gamma$ parameter ($\gamma$ is multiplied by two and the number of rounds by 1.5). As shown in Figure 9, for a number of vehicles $N = 10000$, the number of rounds needed to reach all interested vehicles remains limited (i.e., 6 in our case) when a high value of $\gamma$ is used.

As a conclusion, our dissemination protocol is effective since it ensures the dissemination of the summary to all vehicles concerned with a limited number of rounds. Besides, the variation of the $\gamma$ parameter has a less important impact on the number of rounds needed to reach the vehicles interested than the variation of the $\alpha\beta$ parameters.
6. Related works

Our data aggregation scheme may be viewed as a simple data fusion method. Each vehicle directly (or indirectly using the dissemination protocol) observe a set of events which is aggregated into a local summary. This summary is then exchanged and fused with others received from close vehicles. According to data fusion classification schemes detailed in [23], our proposal is of type complementary (vehicles do not observe the same events) but with some redundancy to handle (some events may be observed by several vehicles). We are working at a low level of abstraction (measurement level) with a data in - data out approach according to the DFD model [7]. Another classical architecture for fusion systems is the JDL model [13] initially defined for military applications and structuring data fusion in four levels, from level 0 (source pre-processing) to level 3 (threat refinement). Our proposal is part of level 1 (object refinement).

Besides, aggregation in inter-vehicle networks has so far been considered as a way to optimize storage or to minimize the use of bandwidth. To the best of our knowledge, we are the first work focusing on summarization as a basic construction for decision-making even in the absence of specific information. Recently, many strategies have proposed aggregation in the context of sensor networks including an attempt to reduce energy consumption [25]. However, the high mobility and the large concentration of vehicles in certain geographical areas make very difficult the use of these proposed strategies in the context of inter-vehicle networks. Moreover, energy consumption is not a problem in our context. In the following, we present some works related to aggregation in inter-vehicle networks.

In [27], RLSMP (region based location service management protocol) are proposed. It is based on the aggregation of messages according to geographical areas. Their goal is to reduce the latest positions and the number of messages generated for the management of vehicle locations. The authors precise that aggregation improves scalability, but it can also lead to:

(1) more packet collisions and so more retransmissions (mainly because of the important size of packets exchanged);
(2) longer delays, because of the processing of data before they can be effectively delivered.

The work presented in [10] consider that vehicles aggregate warnings data if they receive multiple messages related to the same event. They also propose the use of invalidation messages when a vehicle did not detect a danger in an area defined as dangerous based on the aggregated information.

In [24], the authors discuss security aspects and more precisely detect attacks involving dissemination messages containing wrong aggregated data. If aggregation can reduce the bandwidth problems, it also can make security problems more complicated to manage. The proposed solution is a tamper-proof service deployed on vehicles. Two types of aggregation are also proposed: syntactic aggregation, which essentially reduces the overload of messages and semantic aggregation, which
is specific to information and saves more bandwidth. The solution presented in this article cannot be used for events such as accidents (only for information on cars, such as speed or location). Similarly, the re-aggregation is not considered and the failure of car registrations by malicious cars (to calculate an aggregate) cannot be detected.

Works mentioned previously generally consider data summarization as a method of compressing information to save bandwidth. Data compression and data aggregation are distinguished in [21]. Only the aggregation process of data considers data semantics. Different aggregation algorithms are proposed. The “ratio based” algorithm considers a division of the road in front of the vehicle into a set of segments. At each of these segments is then associated an aggregation ratio. The “cost-based” algorithm aims for minimizing the cost (based on the error introduced during merging, the number of vehicles affected by aggregation, etc.) of records aggregation.

In [17], the authors present a study of the hierarchical aggregation of data. The motivation for this approach is that a vehicle requires detailed information on its neighbors, but the information related to more distant ones can be less detailed. An algorithm based on the use of Flajolet-Martin sketches is proposed to store approximate information. For example, it is possible to merge two aggregates (even if there are some overlap between them) while avoiding the appearance of duplication. The hierarchy of aggregation is predefined in the map grouping areas according to their natural relationships (e.g., by district or roads). In the context of spatio-temporal applications, the sketches are also used in [28] a way to avoid the problem of duplicates counting for queries with count or sum aggregates. The interest of the approach in [17] is that it supports any type of event. However, the important memory consumption may be very penalizing.

In [5], the authors present the protocol LBAG (location based aggregation). In this protocol, data aggregation relies on a hierarchy of static locations instead of considering a tree of nodes that would be particularly difficult to maintain because of the high mobility of vehicles. A communication protocol Geocast (based on a routing function of the position and diffusion control) is used to issue a message in a target area.

In [6], the authors describe a framework to efficiently summarize several streams joined by a relationship between one another. Summaries are built, which give information both on each stream individually, as well as on their relationship for any given time horizon. To realize this summary, three techniques where used in the summary structure: the first one is the micro cluster [1] that makes use of the cluster feature vector (CFV) aggregate [26]. The second one is the idea of dividing treatment between an online part producing snapshots of the system state, and an offline part analyzing these snapshots [1]. Finally the third technique relies on the use of Bloom Filters [3].

Finally, in [2], the authors propose a formal description of the aggregation process and derives from this description a data structure for the representation of aggregate data. The aggregation data structure (ADaS) provides a semantically
Contribution and perspectives

In this article, we have presented an aggregation structure of spatio-temporal events observed in the context of vehicular ad hoc networks. This structure is based on a two levels spatio-temporal model that allows to manipulate the same physical repository for all vehicles, and can support summaries exchanges easily and without loss of information. Our aggregation structure is realistic in size if the number of temporal dimensions remains controlled. The complexity of access to the structure is also efficient (logarithmic or linear). Currently, our aggregation structure is being incorporated in the VESPA simulator to aggregate events (e.g., released parking spaces) and an experimental evaluation of our solution is underway. For this, we use the simulator developed for the VESPA system, allowing the sharing of information between vehicles. This simulator, originally developed to evaluate the gain of inter-vehicles communication protocols, has been extended to simulate parking lots where vehicles are searching for an available space, while others leave their place and notify nearby vehicles etc. Throw this evaluation, we will validate the size of the aggregation structure and study its stability. We also want to study the contribution of these aggregation data to design a decision support system to help drivers, using multiple sources of information (e.g., observed events, summaries, external sources like traffic information, etc.). These systems have to manage information with different level of accuracy. In this context, works on probabilistic databases are an interesting direction to define a query evaluator able to handle all these information.

Acknowledgements. The present research work has been supported by Institut Telecom with a Futur and Rupture grant, the International Campus on Safety and Intermodality in Transportation (CISIT), the European Community, the Regional Delegation for Research and Technology, the Ministry of Higher Education and Research and the National Center for Scientific Research. The authors gratefully acknowledge the support of these institutions.

References

SUMMARIZING SENSORS DATA IN VEHICULAR AD HOC NETWORKS


