Modeling In-Network Aggregation in VANETs

Stefan Dietzel, Frank Kargl, and Geert Heijenk, University of Twente
Florian Schaub, Ulm University

Abstract

The multitude of applications envisioned for vehicular ad-hoc networks (VANETs) requires efficient communication and dissemination mechanisms to prevent network congestion. In-network data aggregation promises to reduce bandwidth requirements and enable scalability in large vehicular networks. However, most existing aggregation schemes are tailored to specific applications and types of data. Proper comparative evaluation of different aggregation schemes is difficult. Yet, comparability is essential to properly measure accuracy, performance, and efficiency. We outline a modeling approach for VANET aggregation schemes to achieve objective comparability. Our modeling approach consists of three models, which provide different perspectives on an aggregation scheme. The generalized architecture model facilitates categorization of aggregation schemes. The aggregation information flow model supports analysis of where information is aggregated by a scheme. The aggregation state graph models how knowledge about the road network and its environment is represented by a scheme. Further, it facilitates error estimation with respect to the ground truth. We apply each modeling approach to existing aggregation schemes from the literature and highlight strengths, as well as weaknesses, that can be used as starting point for designing a more generic aggregation scheme.

I. Motivation

Vehicular ad-hoc networks (VANETs) enable vehicles to communicate with each other on the road. Many applications have been envisioned for this kind of mobile ad-hoc communication. Intersection collision warning, lane merge assistance, smart traffic management, and emergency vehicle warnings are just a few examples of potential applications that promise to enhance road safety and efficiency, support drivers, or provide infotainment

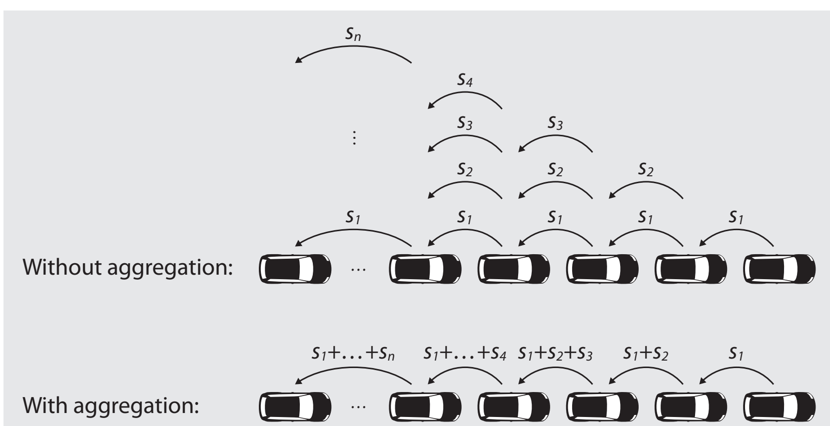

Fig. 1: Schematic bandwidth comparison between normal message dissemination and aggregation for an application where each vehicle disseminates its current speed s_i.

IEEE Communications Magazine, vol. 49, no. 11, November 2011
Author pre-print version. Definitive version at: http://dx.doi.org/10.1109/MCOM.2011.6069721

services. Currently, many research projects and field operational trials prepare the deployment of VANET technology in Europe (e.g., DRIVE C2X, simTD, PRESERVE, SCORE@F), the USA (e.g., VSC, VSC-A), and Japan (e.g., SKY).

A major challenge in the deployment of VANETs is the efficient usage of available bandwidth considering the large number of envisioned applications and the even larger number of potential network nodes. Especially multi hop dissemination of information, which is required by some applications, creates a considerable scalability problem. Thus, the development of efficient forwarding and dissemination protocols has been a major research focus [1]. In-network data aggregation can potentially provide scalability for multi hop communication and enable the co-existence of multiple different applications by reducing per-application bandwidth requirements. Instead of many vehicles sending single messages, which are all forwarded individually, multiple similar messages can be combined into one aggregated message that represents the accumulated content.

Consider for example a traffic information system as shown in Figure 1, which is used to detect traffic jams. A vehicle inside a traffic jam detects that it is not moving and its neighbors are also not moving or only moving at slow speed. Without aggregation, the vehicle would send a message reporting the condition as a geo-broadcast towards vehicles approaching the traffic jam. Other vehicles in the traffic jam would also start generating such messages. In order to be useful for approaching vehicles, i.e., for car navigation, the traffic information needs to be disseminated over multiple hops. Now, instead of forwarding many similar messages, which would congest the wireless medium, vehicles can aggregate their own view with warnings received from other vehicles and only disseminate the aggregate message. Such an aggregated message then contains summarized information about the traffic jam (e.g., 6 km long, at location X, 312 vehicles involved, average speed 3.2 km/h) once it is received by approaching vehicles.

The example shows the main benefits of in-network aggregation. Bandwidth requirements can be reduced, and less resources are required at receiving vehicles, because fewer messages need to be processed. Reduced processing and communication requirements also imply reduced energy requirements for on-board units.

These benefits are well recognized in the VANET research community. But current aggregation schemes and their aggregation functions are often tailored to specific scenarios and information types. As an advantage, these schemes are presumably optimized for their scenarios and applications. The downside is that it is inherently difficult to compare the performance and accuracy of different aggregation schemes. Furthermore, most schemes cannot support multiple applications simultaneously, thus limiting the overall beneficial impact of aggregation. While the need for standardization is recognized by international standardization bodies, like ETSI [2], no specific suitable candidates for generic aggregation schemes are mentioned.

A. Existing VANET Aggregation Schemes

Early approaches for vehicular networks used fixed road segments [3] for aggregation. However, this simple approach shows limitations in terms of scalability [4] and flexibility [5].

More recent proposals argue that hierarchical aggregation is a key element to achieve scalable systems [4], [6], [7]. Sometimes hierarchy is implicitly present, because information further away is stored with reduced accuracy [6], [8]. Other schemes propose a more flexible partition of the road, claiming that this improves both data quality and communication efficiency [5], [8]–[10]. Beyond scalability aspects, some authors focus on aggregation-friendly forwarding mechanisms to make it more likely that aggregatable information items meet [11]. Other reoccurring aspects in related work are improved data fusion mechanisms, duplicate elimination [7], or improving real world approximation [8]. Some authors also worked on efficient data compression to make lossy aggregation supposedly unnecessary [12].

Most of the above-mentioned schemes specifically target the dissemination of current traffic state. Some focus on aggregating parking space availability. In both cases, application-domain-specific assumptions are commonly made, for instance, optimizing communicated data structures and data fusion functions for the specific use case.

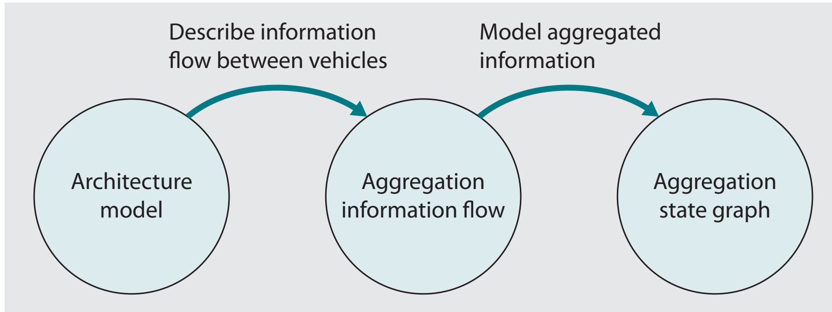

Fig. 2: Aggregation modeling workflow.

Even if schemes support different types of application data, none of them disseminates combined data from different application domains. More importantly, existing schemes not only focus on one specific application domain, but they also focus on one specific aspect of aggregation. For instance, some schemes propose flexible mechanisms for road network partitioning, but rely on a duplicate sensitive average for data fusion. Other schemes employ a robust duplicate insensitive data fusion mechanism, but rely on fixed road segmentation as the underlying aggregation structure. Thus, while numerous promising aggregation schemes exist, it is hard to select candidates for possible standardization or further development.

We argue that a generic modeling approach for data aggregation in VANETs is required to address these issues and unlock the full potential of in-network aggregation in vehicular networks. In this paper we extend the work of [13], which focuses on using modeling tools to achieve a generic aggregation scheme. In particular, this paper uses specific examples to better illustrate how the presented models can be applied to existing aggregation schemes. Each modeling tool highlights a different aspect or functional component of the presented schemes. Beyond introducing models for comparison, we outline how the models can be used to instantiate generic schemes.

B. A New Modeling Approach

Our modeling approach consists of three orthogonal modeling components (see Figure 2): an *architecture model*, an *information flow model*, and an *aggregation state graph*. The architecture model describes an aggregation scheme in a structured way, differentiating the main functional components. The information flow model serves as a tool to understand where an aggregation scheme combines information items and also to exemplify requirements for aggregation schemes. The aggregation state graph models the information that is communicated and serves to compare the accuracy of different real world representations.

In the following sections, we will describe our modeling approach and the requirements for each component. We will use a simple aggregation scheme similar to [3] as a *baseline scheme* to show how each tool can be applied in practice. The baseline scheme uses fixed-size road segments, for which the average speed is calculated and disseminated. No hierarchic aggregation is applied. In the discussion, we will compare the baseline with newer, more advanced implementations.

II. Architecture Model

The architecture model describes functional components of an aggregation scheme. Figure 3 shows the overall system: a vehicle receives information from remote vehicles or local sensor observations. The data is then processed inside the vehicle and periodically disseminated to direct neighbor vehicles. Neighboring

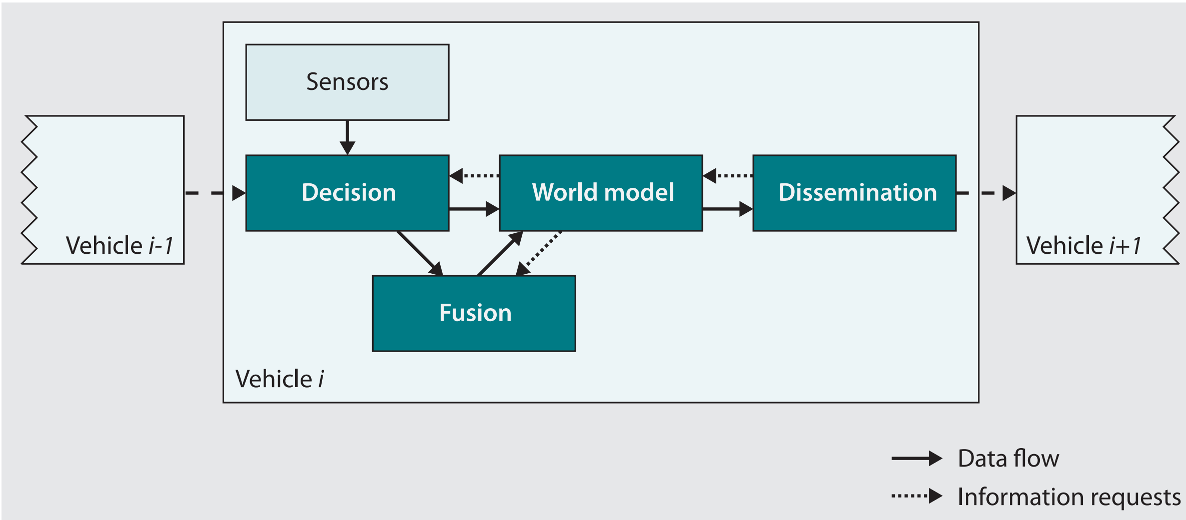

Fig. 3: Architecture model of the baseline scheme, showing the functional components required by all aggregation schemes.

vehicles apply the same scheme, therefore achieving an implicit multi hop dissemination. Inside a vehicle, an aggregation scheme can be sufficiently described by four tasks and respective components: (1) *decide* whether data items can be aggregated, (2) *fuse* several data items together, (3) maintain a *world model* composed of received and self-obtained information, and (4) periodically *disseminate* a subset of the world model to other vehicles. Note that Figure 3 shows one specific instantiation of the architecture model, which aligns with the baseline scheme. Other arrangements of the functional components are possible. For instance, some aggregation schemes argue that local storage is cheap, and therefore decision and fusion are only done before dissemination and not before adding new information to the local world model. Similarly, specific functional components might be very simple in some schemes. But independent of these variations, the set of required functional components stays the same.

A. Decision

The *decision component* is responsible for deciding whether two information items are similar enough to be aggregated. To reach the decision, all information contained in the presented items, as well as all information in the world model, can be used. The more items the decision component selects for aggregation, the more precise information is lost.

Therefore, the most important requirement for the decision component is *flexibility.* To achieve an efficient yet realistic approximation of real world phenomena, the decision component needs to be able to combine information on long road stretches in case of large phenomena, such as traffic jams. On the other hand, small incidents, like a single car having an accident, need to be represented and must not be lost because of too coarse aggregation [9]. Furthermore, the decision component needs to be able to reduce data granularity with growing distance to the event origin in order to scale well in large networks [4].

The baseline scheme uses a fixed decision structure. Each road is segmented into equally long slices. Two information items are aggregated if they originate from the same road segment. However, such a decision is not flexible enough following the above definition. Newer aggregation schemes include different influence factors in the aggregation decision. For example, fuzzy logic rules provide a means to flexibly express aggregation decision rules [5].

B. Fusion

Once two information items have been selected for aggregation, the *fusion component* performs the actual data fusion. Fusion can either be a lossless or a lossy process. However, lossless fusion can be insufficient to achieve the bandwidth reduction required for large-area data dissemination [4]. Therefore, many existing aggregation schemes use lossy data fusion. To build a good lossy aggregation scheme, application domain knowledge is necessary. Such knowledge helps to decide which data items are important for application decisions, and which data items can be removed in the aggregation process without loosing too much required information. Important properties of a fusion function are:

– *Hierarchical applicability.* Often, support for hierarchical aggregation is necessary, for instance, to reduce data granularity with increasing distance. Without it, an aggregation scheme cannot scale to large dissemination areas, as argued in [4]. Therefore, fusion functions need to be hierarchically applicable.

– *Duplicate insensitivity.* Many events will be sensed by more than one car (e.g., free parking spots). Depending on the application domain, it is necessary to filter out duplicates in order to avoid false information.

– *Data quality tracking.* Ideally, a fusion function provides a means to keep track of data quality when aggregating. For instance, when data is averaged the standard deviation can be kept as a data quality metric.

In the baseline scheme, speed reports from different vehicles are simply averaged. There is no protection against a single vehicle adding the same speed report multiple times, thereby biasing the result. One possibility to achieve duplicate insensitivity is to trade off exact counting. By employing a variant of FM-Sketches [7], duplicates can be filtered while maintaining hierarchical applicability of the data fusion function. Similarly, the average alone does not indicate the heterogeneity of the underlying data. In [5], the standard deviation is added to the average to keep track of the aggregation error, even throughout multiple applications of the fusion function.

C. World model

A vehicle's *world model* collects all information available to that vehicle. It changes over time due to the reception of previously unknown information from other vehicles.

The key requirement for the world model is to support *efficient range queries* for subsets of the contained data. The decision component needs to query the world model for potentially aggregatable information whenever new information is received. The dissemination component also needs to query subsets of the information in the world model to determine the information to be disseminated to neighboring vehicles.

The baseline scheme uses a simple lookup table to match road and segment ids to their current traffic state. Because the road and segment ids can be ordered, searching is efficient in the table. However, this lookup mechanism does not scale to more flexible aggregation structures, neither does it scale to aggregation of two-dimensional area information, such as in city scenarios.

Abraham et al. [14] provide a survey of suitable index structures for more complex aggregation schemes without fixed size segments. For instance, quad trees, as used in [6], provide an efficient querying mechanism for certain regions of a two-dimensional road network.

D. Dissemination

After possible aggregation, the *dissemination component's* job is to relay information to neighboring vehicles, resulting in indirect multi hop dissemination. However, it is not necessary to flood new information directly. A vehicle can periodically broadcast a subset of its world model to neighboring vehicles, which in turn will

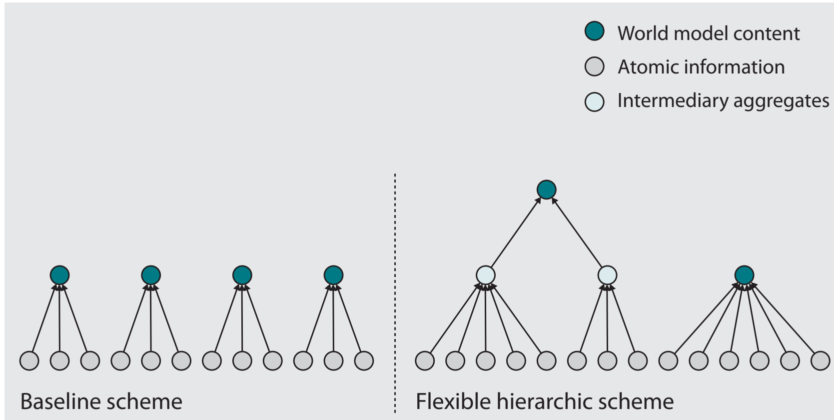

Fig. 4: Information flow models for different aggregation schemes.

continue to disseminate the information to vehicles further away. Several steps are necessary to select a suitable subset of information for further dissemination:

– *Bandwidth profile selection.* A bandwidth profile specifies the average amount of data per time period that can be used for information dissemination.

– *Data selection.* Next, suitable data needs to be selected for dissemination. A generic selection can include the most recent information from a reasonably large surrounding area. However, the data selection rules need to be configurable. For instance, a traffic state application might assign higher dissemination priority to information about traffic jams than to information about free-flowing traffic.

– *Periodicity.* In addition to periodic beaconing, more elaborate dissemination algorithms can be used. For instance, carry-and-forward can be used by vehicles driving in the opposite direction of a traffic jam to inform upcoming vehicles about the congestion.

The baseline scheme uses periodic beaconing with fixed packet sizes to disseminate subsets of the world model. If the world model content representation is larger than the packet size, priority is given to information closer to the current vehicle.

Extending this simple prioritization, [5] applies the idea of relevance-based information dissemination proposed by Kosch et al. [15]. That is, a number of relevance functions can be defined that prioritize the information inside the world model according to different criteria. Several of these rating functions can be used in parallel to apply different metrics for different applications. Each of these functions is then assigned a fraction of the total available bandwidth until the limit of the bandwidth profile is reached. This allows a flexible allotment of bandwidth to different applications with different requirements.

III. Aggregation Information Flow

The architecture model describes the information flow between the functional components of an aggregation scheme. The next step in understanding an aggregation scheme is to look at where information is aggregated. The goal of the information flow model is to represent aggregated information and its origins from the viewpoint of one particular target vehicle. For each target vehicle, the representation is different, but has the same characteristics.

The information flow can be represented with a directed graph structure, as shown in Figure 4. The topmost nodes of the graph represent the aggregated information present in the target vehicle's world model. The lower nodes represent the underlying aggregation steps of each aggregate. The lowest nodes, that is, the graph's leaf nodes, represent atomic sensor information at a specific point in space and time. The graph's directed edges represent aggregation of atomic or aggregated information items.

The information flow representation changes over time. It will converge when the aggregation scheme has disseminated enough information to closely depict the real world situation. Ultimately, we are interested in the graph after an infinite number of protocol steps. That is, given infinite time to disseminate all information available, what is the aggregation structure that a scheme builds? We can use this converged view to analyze a number of characteristics:

– *Support for hierarchical aggregation.* If the depth of the resulting graph is 1, an aggregation scheme does not support hierarchical aggregation.

– *Information dissemination range.* The number of leaf nodes in the graph represents the source area of the known information. Also, if information is aggregated more with increasing distance to the target vehicle, the graph structure will represent this by showing higher depths with increasing distance to the target vehicle.

– *Flexibility of the aggregated view.* If all aggregate nodes are connected to the same number of atomic nodes, the scheme is likely to use a fixed threshold for aggregation decisions. Similarly, the more irregular the structure of the information flow model is, the more flexible are the underlying aggregation decisions.

Thus, the information flow model gives an overview of how a specific aggregation scheme works. The graph's structure allows to explore the aggregation decision rules and dissemination scheme.

The baseline scheme, shown on the left in Figure 4, uses fixed size road segments for aggregation decisions. Therefore, the number of atomic information items leading to an aggregate is equal for each aggregate node. Moreover, no hierarchical aggregation is used as shown by the shallow graph structure with a maximum depth of 1. In contrast, a more flexible scheme, shown on the right in Figure 4, employs hierarchical aggregation to achieve a better bandwidth reduction with increasing distance to the target vehicle. Also, road segments in the flexible scheme are not fixed. Instead, aggregation decisions are made based on data similarity, such as similar speed information. The shown example is based on [5]. Similarly, [6] employs a quad-tree structure to achieve hierarchic aggregation, but uses fixed segmentation based on city neighborhoods, which is less flexible.

IV. Aggregation State Graph

Having an overview of the information flow, the next step is to model the communicated information itself. Existing aggregation schemes for vehicular networks use different, incompatible data representations. These are either suited for one particular use case, such as traffic state dissemination, or suited for a class of data, such as averaged information (e.g., traffic state, weather, travel time) or approximately counted information (e.g., parking spots). While such tailored representations are very bandwidth efficient, they make it difficult to compare two given aggregation schemes. This is reflected by the fact that existing aggregation schemes use different and orthogonal metrics for evaluation. Some use time delay induced by aggregation, some visibility, some the effect on travel times. Moreover, most existing schemes compare against non-aggregating schemes as a baseline. Rarely, two aggregation schemes are compared with each other using the same metric. In order to achieve comparability, we propose to use a generic graph model to represent the state of the road network as seen by an aggregation scheme, as well as to represent the ground truth of the actual real world traffic state.

The graph structure itself consists of road network junctions and bends, making up the graph's nodes, and the connecting roads, represented as edges of the graph. An alternative to using a graph structure would be to represent location simply as a continuous two-dimensional plane. However, all information we are concerned with originates from vehicles, which drive on a defined road network. Therefore, we argue that a graph structure is better suited to represent location information. The graph structure is annotated with time, as well as any further information that needs to be disseminated for applications, such as speed or temperature. The following are the main requirements for the road network state graph:

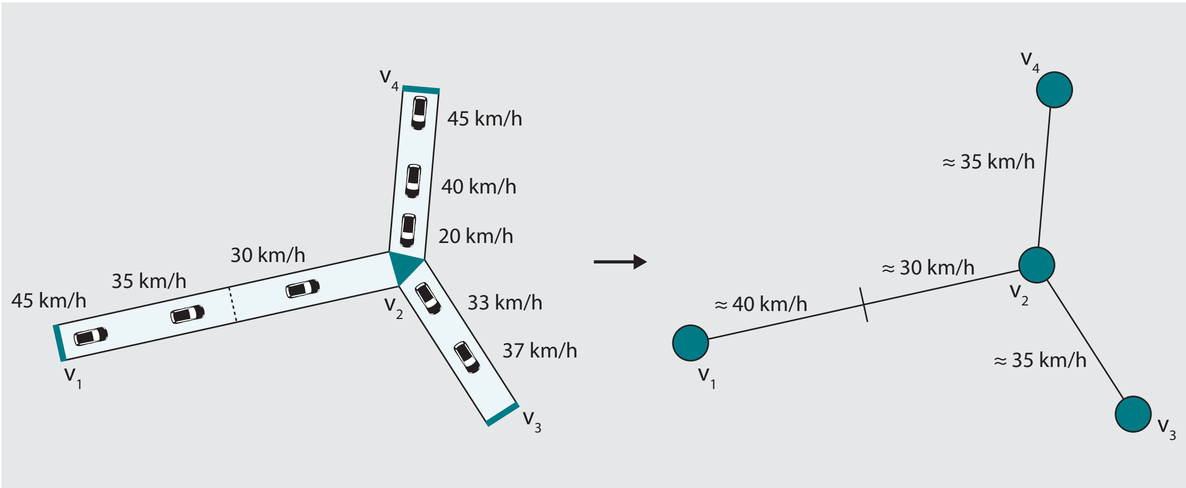

Fig. 5: An example road network and its state graph representation in the baseline scheme.

– *Fuzzy information.* In oder to deal with aggregated information, the state graph needs to cope with fuzziness of information. For example, speed could only be available as an average with a given standard deviation. Or the number of parking spots could be available as a number plus a certainty in percent, due to imperfect recognition algorithms.

– *Multiple application domains.* Given the number of different envisioned applications for inter-vehicle networks, it is safe to assume that several different applications requiring a broad set of information types need to be supported at the same time. Therefore, a state graph for in-network aggregation needs to handle more than one type of application data.

– *Comparability.* Given two graphs representing the aggregated view of two schemes, it should be possible to employ comparison metrics. One particularly interesting metric is the information quality loss due to aggregation. This can be represented as the mean error of the aggregated view compared to the real world data without aggregation.

We will now use a simple state graph representation to determine the aggregation error of the baseline scheme for a small example road network. We will then compare the baseline scheme to an all-seeing observer that knows the complete real world data from the road network. We use a graph $G = (V, E, P, S_t)$ with

- $V = \{v_1, \ldots, v_n\}$ the vertices (i.e., intersections and bends) of the road network,
- $E = \{e_1, \ldots, e_m\}$ the edges (i.e., streets connecting the vertices),
- $P = (p_1, \ldots, p_n)$ the positions of the vertices as GPS coordinates, and
- $S_t = (s_{e_1}, \ldots, s_{e_n})$ a set of functions describing the average speed on each edge e_i at time t.

Modeling the baseline scheme representation is straight-forward. Figure 5 shows the example road network we will use, as well as the aggregated state graph. Four vertices are connected by three road segments. The road segments $\{v_2, v_3\}$ and $\{v_2, v_4\}$ are both shorter than the fixed segment size. Therefore, the aggregated speed functions are both constant, that is, $s_{\{v_2,v_3\}} = 35$ and $s_{\{v_2,v_4\}} = 35$. The longer road $\{v_1, v_2\}$ is represented as two fixed segments. The average speeds here are given by a piecewise-defined function of the relative position x on the edge, where $x = 0$ is at the position of v_1, $x = 0.5$ is in the middle of the edge, and $x = 1$ is at the

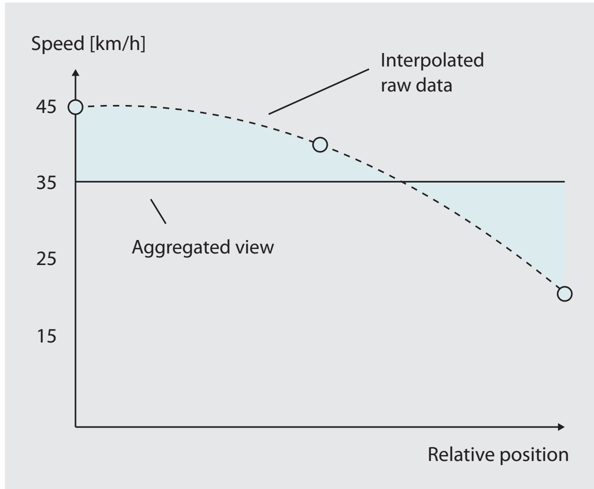

Fig. 6: Comparison between interpolated raw data and aggregated data for road segment $\{v_2, v_4\}$.

position of v_2:

$$s_{\{v_1,v_2\}}(x) = \begin{cases} 40 & (x < 0.5) \\ 30 & (x \geq 0.5) \end{cases}$$

Modeling the ground truth, that is, the road network state as seen by an all-knowing observer, is a more difficult task. Ultimately, the ground truth is the raw sensor data from each vehicle. This results in a large set of atomic information items for different points of the road network, which needs to be processed and interpreted to be comparable with aggregation schemes. However, this needs to be done in a way that is not biased towards a specific aggregation scheme. For our example, we use polynomial interpolation to gain a speed function from raw data samples. Figure 6 shows a polynomial that interpolates the raw data of road segment $\{v_2, v_4\}$. In addition, the aggregated view for the same segment is shown. The highlighted area in Figure 6 shows the aggregation error of the baseline aggregation scheme. The changes in traffic situation from free-flowing traffic around v_2 towards slow traffic at v_4 are not well represented due to the fixed segment length. By calculating the average difference between the aggregated speed function and the interpolated raw data function for the whole graph, the overall aggregation error can be obtained. The average error for the whole graph can serve as a first metric to compare two aggregation schemes. For a more detailed comparison, the aggregation error for specific scenarios can be compared, e.g., inner city junctions or highway segments.

V. Outlook

In this paper, we have presented a modeling approach comprised of three orthogonal modeling tools: an *architecture model* to understand important functional components of aggregation schemes, an *information flow model* to gain insight into the information dissemination and fusion mechanisms used, and a *state graph* to distill numeric metrics to compare existing schemes. For each modeling component we have outlined the desired functionality and important requirements. Throughout the paper, we used a baseline aggregation scheme to show how to apply our models in practice. Where possible, we also discussed more advanced aggregation schemes in comparison to the baseline approach.

The next evolutionary step is to use the modeling tools as a foundation for a truly generic aggregation scheme. Currently, several use cases for approximate multi hop data dissemination are envisioned. Traffic state information, weather data, dangerous road conditions, and parking space information are the most frequently mentioned examples. If in a future VANET deployment each of these applications uses a separate aggregation scheme, then benefits of aggregation will be diminished soon. Some examples for the generic models' instantiation as a generic scheme can be found in [13], but the design of a complete generic scheme is still an ongoing challenge.

In a first step, a generic architecture could assume similar data quality requirements for each information type. A generic aggregation scheme could then run as a network service, similar to cooperative awareness messages (CAM) [2] in the single hop case. Multiple applications could subscribe to such a generic aggregation scheme and add payload data. Each data type would be aggregated following common data quality requirements. For the step from generic modeling to a generic architecture, especially the architecture model and state graph can provide an important starting point. The architecture model components can be instantiated with concrete implementations to describe the generic aggregation scheme. Likewise, an efficient representation of the state graph can be used as a data structure throughout the system. Current research has shown that a generic aggregation scheme should not use pre-defined road segments for aggregation decisions [4], [5] and should ensure data quality, for instance, by employing duplicate elimination [7]. However, it has not been shown yet how these requirements can be combined in a generic aggregation mechanism.

In a second step, a generic architecture should be extended to support applications with different quality requirements at the same time and without introducing redundant messages. To achieve this, a sophisticated data representation is necessary. For instance, progressive encoding techniques could be employed to disseminate messages with the least required quality for all domains. In addition, more fine-grained information could be disseminated by encoding only the differences to the baseline messages.

While it has not been demonstrated yet that such a generic aggregation scheme can really be constructed, we consider it necessary to generalize and unify existing approaches. Besides efficiency and scalability, security and especially integrity aspects need to be taken into account. If a suitable generic aggregation scheme for different kinds of application data can be standardized, in-network aggregation will be able to show its true potential to help building scalable vehicular communication systems.

References

[1] E. Schoch, F. Kargl, M. Weber, and T. Leinmuller, "Communication patterns in vanets," *Communications Magazine, IEEE*, vol. 46, no. 11, pp. 119–125, November 2008.

[2] ETSI, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications," 2010.

[3] L. Wischhof, A. Ebner, H. Rohling, M. Lott, and R. Halfmann, "SOTIS – a self-organizing traffic information system," in *Proceedings of the 57th IEEE Semiannual Vehicular Technology Conference, 2003.* IEEE, 2003, pp. 2442–2446.

[4] B. Scheuermann, C. Lochert, J. Rybicki, and M. Mauve, "A fundamental scalability criterion for data aggregation in VANETs," in *Proceedings of the 15th annual international conference on Mobile computing and networking*. New York, New York, USA: ACM Press, 2009, p. 285.

[5] S. Dietzel, B. Bako, E. Schoch, and F. Kargl, "A fuzzy logic based approach for structure-free aggregation in vehicular ad-hoc networks," in *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking*. New York, New York, USA: ACM Press, 2009, pp. 79–88.

[6] M. Caliskan, D. Graupner, and M. Mauve, "Decentralized discovery of free parking places," in *VANET '06: Proceedings of the 3rd ACM International Workshop on Vehicular Ad hoc Networks*. New York, New York, USA: ACM Press, 2006, pp. 30–39.

[7] C. Lochert, B. Scheuermann, and M. Mauve, "Probabilistic aggregation for data dissemination in VANETs," in *Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*. New York, New York, USA: ACM Press, 2007, pp. 1–8.

[8] M. van Eenennaam and G. Heijenk, "Providing Over-the-horizon Awareness to Driver Support Systems," in *Proceedings of 4th IEEE Workshop on Vehicle to Vehicle Communications*, 2008.

[9] S. Dietzel, E. Schoch, B. Bako, and F. Kargl, "A Structure-free Aggregation Framework for Vehicular Ad Hoc Networks," in *Proceedings of the 6th International Workshop on Intelligent Transportation*, Hamburg, Germany, 2009, pp. 61–66.

[10] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode, "TrafficView: traffic data dissemination using car-to-car communication," *ACM SIGMOBILE Mob. Comp. and Comm. Review*, vol. 8, no. 3, p. 6, Juli 2004.

[11] B. Yu, J. Gong, and C. Xu, "Catch-up: a data aggregation scheme for vanets," in *Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking*. ACM, 2008, pp. 49–57.

[12] K. Ibrahim and M. C. Weigle, "CASCADE: Cluster-Based Accurate Syntactic Compression of Aggregated Data in VANETs," in *2008 IEEE Globecom Workshops*. IEEE, November 2008.

[13] S. Dietzel, F. Kargl, G. Heijenk, and F. Schaub, "On the potential of generic modeling for VANET data aggregation protocols," in *Proceedings of the 2nd IEEE Vehicular Networking Conference.* IEEE, 2010, pp. 78–85.

[14] T. Abraham and J. F. Roddick, "Survey of spatio-temporal databases," *GeoInformatica*, vol. 3, pp. 61–99, 1999.

[15] T. Kosch, C. J. Adler, S. Eichler, C. Schroth, and M. Strassberger, "The scalability problem of vehicular ad hoc networks and how to solve it," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 22–28, October 2006.