Data Aggregation in VANETs
A Generalized Framework for Channel Load Adaptive Schemes

Josef Jiru
Fraunhofer ESK
Munich, Germany
Email: josef.jiru@esk.fraunhofer.de

Lars Bremer
Fraunhofer ESK
Munich, Germany
Email: lars.bremer@esk.fraunhofer.de

Kalman Graffi
University of Düsseldorf
Düsseldorf, Germany
Email: graffi@cs.uni-duesseldorf.de

***Abstract*—One of the main communication challenges in vehicle-to-x communication is scalability. With increasing number of communication nodes the wireless channel must not get congested especially if a large amount of sensor data has to be forwarded over multiple nodes to a data processing application. This challenge can be solved by reducing the data load through data aggregation. This work introduces a framework for data aggregation as a decentralized congestion control mechanism on the application layer. This framework can be used to flexibly design aggregation schemes that adaptively adjust the generated data load depending on the overall channel load. Three basic aggregation schemes with different complexity and resulting data precision were developed within this framework and they are discussed in this paper. Performance evaluations show that the aggregation schemes are able to adapt to given channel load thresholds within seconds and deliver optimal data quality even in traffic jam situations.**

Keywords-Data Aggregation, Adaptive Systems, V2X Communication, Cross-layer Optimization, VANETs

I. Introduction

The next step towards improving road safety and traffic efficiency is direct communication between vehicles and their environment, e.g. based on the ETSI ITS-G5 standard [1]. All vehicles transmit Cooperative Awareness Messages (CAM) [2] up to 10 times per second including sensor data like temporal id, current position and velocity. Other vehicles and infrastructure units (Roadside Units, RSU) receive CAMs in their communication range. RSUs can gather sensor information over a road segment and forward this information multihop over several RSUs to a processing application on a control RSU. This application evaluates traffic situations in real time and warns vehicles upon dangerous situation in real time. The use case is illustrated in Figure 1.

One of the main challenges is scalability of the communication on a shared wireless channel. With increasing number of communication nodes the wireless channel must not get congested. Safety messages must be received in time while other services like forwarding sensor data to an application should adapt to the channel load. Especially in traffic jam situations RSUs cannot transmit all collected sensor information to the processing application and need to reduce the data to forward.

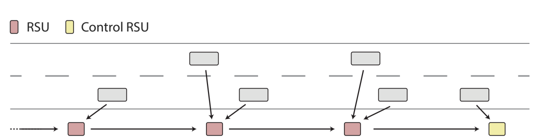


Figure 1. Vehicular Data is Forwarded to Traffic Application

In this paper we present a generalized framework for channel load adaptive aggregation schemes. This framework allows the design of highly customizable data aggregation schemes which meet the following criteria: 1. The schemes are able to adapt their aggregation levels based on the RSU's individual channel busy ratios (CBRs); 2. They are able to meet a minimal targeted data precision depending on specific application requirements; 3. It is possible to allow for a flexible configuration of aggregation levels depending on specific application requirements; 4. The data aggregation can be optimized for several metrics at once.

Considering a typical VANET application scenario, we show how the proposed framework can be used to develop three exemplary aggregation schemes. Using network simulations, we are able to show that the aggregation schemes developed within our framework allow for a flexible trade-off between the processing resources, the caused channel load and the resulting data precision. This paper starts with previous work and design goals in Section II, followed by an overview of the aggregation framework in Section III. In Section IV the three aggregation schemes are described. The schemes are evaluated in a given traffic scenario in Section V. A summary concludes the paper in Section VI.

II. Previous Work and Design Goals

Data aggregation describes the process of combining data records of different sources like the vehicles in a Vehicular Ad Hoc Network (VANET). RSUs might receive and aggregate vehicular data before it is forwarded to a data sink. This aggregation process can be generally divided into three phases [3]: In the *decision phase*, the aggregation scheme decides which data items should be fused in the following *fusion phase* with a given fusion function. In the *dissemination phase*, data is forwarded towards a data sink.

In previous work different data structures, decision and fusion strategies have been proposed. In TAG [4] a simple table is used to store the data. Tables, however, do not support selective fusion that combines only some metrics of two data records. While TAG proposes to use the absolute values stored in a table, CASCADE [5] suggests to store only relative values to a fix point. During the decision process, data records are identified for fusion. One strategy is to define data groups and fuse all data within this group. Such group can be defined by splitting the road into segments as proposed in SOTIS [6] and CASCADE, or more freely as proposed in TAG. The decision component of TrafficView [7] uses a cost function to identify the two items with least fusion costs. This function takes the distance of the vehicles and the number of vehicles represented by a data record into account. However, TrafficView's cost function falls short in considering other metrics than the distance of the vehicles and the number of vehicles represented by a data record. Most data aggregation schemes only focus on averaging data in the fusion process. However, TAG describes a query language that allows to specify different fusion functions. The query can further specify data selected for fusion and ignore data not meeting certain criteria. In any case, fusing data by one certain metric may have disadvantages in proposed use case. Individual extrema values might get lost by fusing over all elements of a group. Fusion may cause a safety threat, e.g. a slow car, might not be identifiable after fusion.

Most aggregation schemes optimize one specific component of the aggregation process. There is no reliable system with data structure and components that support flexible configuration of requirements on data precision with a flexible number of non-predefined metrics and the possibility to consider extrema values that always adapt to highly dynamic environment like VANETs.

The main goal of the proposed aggregation framework is an adaptive data aggregation that triggers data fusion using different levels when the wireless channel is getting congested, providing the best possible quality of service at the same time. The aggregation framework was designed to allow easy configuration of aggregation schemes that meet the following requirements: 1. *Reliable Delivery*: Sensor data should be reliably delivered to the control RSU minimizing packet loss; 2. *Delay Sensitivity*: The maximum end-to-end delay of data should be kept under a threshold; 3. *High Precision*: The aggregated data that arrive at the control RSU should have minimal deviation from the original data; 4. *Scalability*: The required quality of service should be provided even when scaling up the number of involved RSUs, vehicles and the length of the road segment; 5. *Flexibility*: The aggregation scheme should support different types of sensor values, seamless adding of new values and allow transmitting of extrema values.

III. Data Aggregation Framework

The proposed aggregation framework provides a foundation to design adaptive aggregation schemes. It is based on a modular architecture with five main modules. An overview of the framework is provided in Figure 2. Each phase of the aggregation process is represented by a single module following the generic architecture model for aggregation schemes proposed by Dietzel et al. [3]. Additionally, two modules are defined. One represents the data structure used in the aggregation process and the other implements the adaptive control of the aggregation process. The implementation of each module defines the properties of an aggregation scheme. Following, each of the five modules of the aggregation framework are introduced in more detail.

When a RSU receives vehicular information the data is stored in a *data structure*. The data structure stores different data types, combines data of multiple sources into one structure

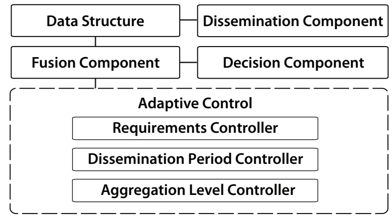
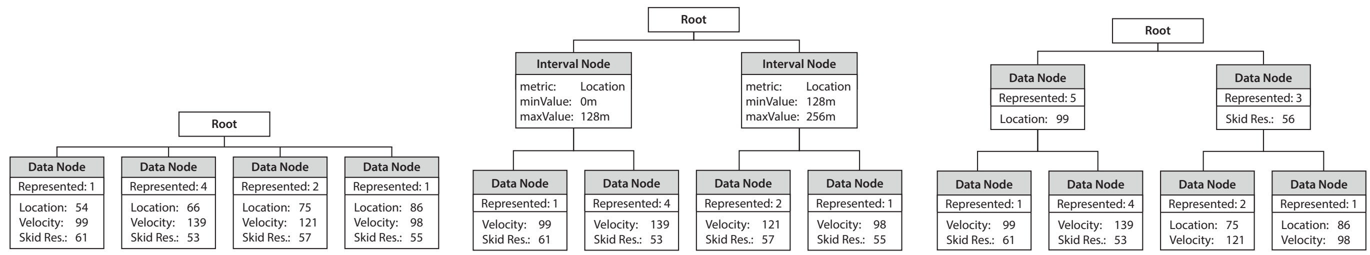


Figure 2. Aggregation Framework Modules

and supports size reduction by data fusion. Data structures with different tree representations are supported. The *decision component* chooses the most similar data records for fusion to achieve high data precision. It is based on the individual standard score of each metric. The *fusion component* provides a valid aggregation tree to the dissemination component, collects instructions how to aggregate data from the aggregation level control and allows the decision component to determine what data to fuse if necessary. It keeps the aggregation tree valid at all times. The *data dissemination* component defines when and how data is disseminated by a RSU to the next RSU in the direction of the data sink. Data is disseminated with an adaptive frequency by the node farthest from the control RSU. Whenever another node receives aggregated data it adds its own data of the requested interval and forwards the combined data immediately. The *adaptive control* with three controllers is responsible for the reliable delivery and the end-to-end delay. It monitors the CBR and controls it with the adaptive aggregation schemes maintaining a target CBR and minimizing the packet loss. The requirements controller defines the required metrics, fusion parameters and an initial dissemination frequency. This controller is triggered during initialization of the aggregation process by the application on control RSU. At runtime, the dissemination period controller observes the delay of incoming information at the control RSU and adjusts the requested dissemination frequency when the delay exceeds the targeted delay. The aggregation level controller is a decentralized component and is executed on each RSU. It observes the CBR and adjusts the aggregation level if CBR exceeds the targeted ratio.

IV. Aggregation Schemes

One key component common to all aggregation schemes is their data structure. Trees as flexible data structures allow to combine specific metrics by extending the tree with additional level for this metrics. Adding new data records or other tree structures to an existing tree is performed by adding the children of the root nodes of both trees to the root of the new tree. This allows an efficient implementation both in-node and in-network aggregation. The level of aggregation is also easy to adjust using the tree as data structure. In general, the fewer nodes the tree contains, the less bandwidth it needs during transmission. Two useful node types for an aggregation tree have been identified: The *data nodes* only contain vehicular data. The *interval nodes* on the other hand are particularly useful if the application requires a defined resolution of data quality. Its children have values that fit into the interval specified by the interval node. Data nodes placed in their fitting interval may only be fused with other nodes from the same interval. This limits the maximal error introduced by the fusion process. Three aggregation schemes designed using the framework are introduced next.



(a) Flat Aggregation Scheme (b) Binary Interval Aggregation Scheme (c) Free Hierarchical Aggregation Scheme

Figure 3. Exemplary Tree Layouts

1) Flat Aggregation Scheme: The flat aggregation scheme uses an adaptive control with an aggregation tree of depth 1. All vehicular data is stored in data nodes that are direct children of the root node as illustrated in Figure 3(a). The size of this tree can easily be restrained by limiting the number of data nodes. Whenever a tree exceeds the configured limit of data nodes, the validity check of the fusion component fails and two or more data nodes are fused until the tree is valid again. The advantage of this scheme is its simplicity and fast processing. However, only entire data records can be fused.

2) Binary Interval Aggregation Scheme: The binary interval aggregation scheme uses an interval nodes of length 2^x to represent metrics that require a minimum quality. The aggregation tree uses a certain number of interval levels in the aggregation tree. An example tree with one interval level is depicted in Figure 3(b). This tree layout allows multiple ways to reduce the tree size. First, the number of data node children can be limited on the last interval level. Second, the interval nodes can be resized. When keeping the children limit constant and increasing the size of the interval, the tree shrinks and data nodes are fused. For in-network aggregation, it is desirable that intervals of different nodes can be easily merged. For that reason, the size of the intervals is restricted to values of 2^x. The major advantage of this aggregation tree layout is the bounded imprecision by design due to the interval nodes.

3) Free Hierarchical Aggregation Scheme: The aggregation tree of the free hierarchical aggregation scheme consists solely of data nodes. The only constraint used for the tree is that each intermediate node, which is a node that is not the root nor a leaf, has only one metric. Thus, the number of metrics left in a leaf node reduces with increasing depth of the tree. The fusion component is free to decide which metric to fuse. The example tree in Figure 3(c) shows that the location metric of the first two leaf nodes has been fused into a new parent node. The two last leaves, however, fused the skid resistance instead. The maximal number of children allowed per node is reduced with increasing aggregation level. Each intermediate node and the root has a variable restriction of the maximal allowed child count. The major advantage of the free hierarchical scheme is its ability to dynamically determine which metrics to fuse and which metrics should be kept individually.

V. Evaluation

A. Simulation Setup

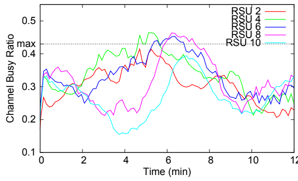
The network simulator ns-3.18 was used for evaluation. It was enhanced by ITS modules enabling simulation of ETSI ITS-G5A [1] and GeoNetworking protocols [8] as well as positioning and mobility modules. The wireless channel assumes Nakagami highway propagation model with 6 MBit/s data rate and 10 MHz bandwidth using the control channel 180 at 5,9 GHz for all communication. Transmission power is 15 dBm for simplification. The targeted maximal CBR is 0.43, following the channel states from DCC White Paper [9]. The CAM transmission frequency follows CAM generation rules [2]. CAM payload size is set to 250 Byte.

A simple traffic scenario is used in the evaluation - a 10 km highway with two lanes in each direction and 700 vehicles randomly distributed on these four lanes. 10 RSUs are placed in a distance of 400 m to each other. The mobility models assume vehicles velocity between 20 - 40 m/s in free traffic flow in both directions. RSUs receive CAMs from vehicles in both direction, but extract only relevant CAMs for further process. After 2 minutes a sudden single directional traffic jam in the middle of the equipped road segment forces the velocity to drop to 3 - 8 m/s. For the next 4 minutes vehicles queue in one direction. Afterwards, traffic jam dissolves slowly for the next 4 minutes, restoring the original velocity distribution. Up to 600 vehicular data records per second per RSU containing 9 different metrics each were received by RSUs and transmitted multihop over 10 RSUs with in-node and in-network aggregation to a processing traffic application.

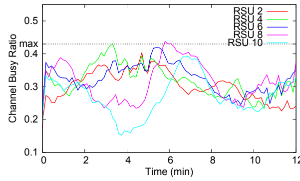
A sample configuration of 10 aggregation levels was used in this simulation. While in the flat and free hierarchical scheme the aggregation levels only depend on the number of children nodes, the binary interval scheme contains additionally two interval layers in the tree restricting the imprecision on these metrics. The first layer defines a temporal interval for timestamp metric, in this simulation the timestamp in CAM data, and is set to 2048 ms. The second interval uses the position metric and is set to 1024 m.

B. Results

Since the main objective of the adaptive data aggregation schemes is to reduce the load on the wireless channel we compare the CBR and aggregation level changes of each RSU followed by the trade-off in resulting precision errors. The CBR of the wireless channel is measured as the ratio of the time a wireless device of a RSU is busy to the total time. Figure 4(a) shows the CBR of a system that only aggregates vehicular data into the tree-based data structure but does not fuse any data. Thus, the channel load can not be reduced and the targeted CBR threshold of 0.43 is exceeded. CBR of the binary interval aggregation schemes is illustrated in Figure 4(b). In the traffic free flow (0-2 min) the CBR is almost stable around 0.35 for all RSUs. As the traffic jam starts the CBR rises for affected middle and later also lower RSUs experiencing dense traffic and decreases for higher RSUs having very low traffic. As the traffic jam moves on in the direction of higher RSUs at min 4-7 every RSU is effected and their CBRs rise having a peak at min 5-7. All three aggregation



(a) No Data Aggregation



(b) Binary Interval Aggregation

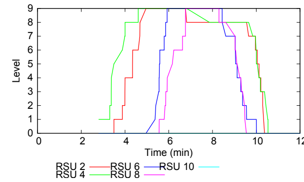
Figure 4. Channel Busy Ratio without and with Data Aggregation

schemes quickly adapt to the channel load and change their aggregation level accordingly to meet the restrictions not to exceed the maximal threshold. While the traffic jam dissolves, CBR lowers for all RSUs back to the initial value.

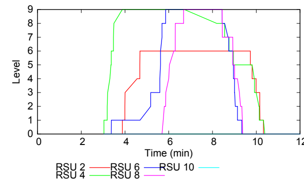
The up- and downgrading of aggregation levels regulates the channel load of aggregated data. Figure 5 shows the aggregation levels for the three adaptive aggregation schemes. The traffic jam starts at min 2 and as the traffic becomes more dense the aggregation levels increase starting about min 3 in the area of RSU 4-5. While the traffic jam extends in both directions, the aggregation levels rise first in the direction to lower RSUs to minimize the additional data load for the rest of the road segment. Later, the traffic moves slowly in the direction of RSUs 7-9 and the aggregation levels rise there too. Beginning with min 7 the traffic jam starts to dissolve and the aggregation levels slowly decrease back to zero.

Data fusion introduces data imprecision that each aggregation scheme aims at keeping low. The resulting error introduced by each scheme is compared exemplary in two metrics: position and velocity. Each figure states the number of data records received with a certain error, the average difference from true value (Mean Absolute Error - MAE) and the Root Mean Square Error - RMSE. The aggregation scheme using no data fusion is used as reference.

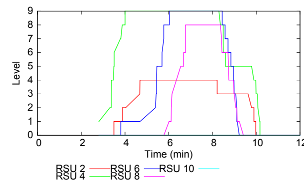
The precision regarding the position metric is illustrated in Figure 6. The flat aggregation scheme with its lowest complexity introduces the highest MAE of 138 meters (RMSE 392 m). The free hierarchical aggregation scheme with its highest complexity lowers the MAE down to 114 meters (RMSE 264 m). The binary interval scheme has the lowest MAE with 60 meters (RMSE 122 m) because the interval metric limits the maximum error by design. In contrast to the position metric, the velocity metric is more precise in the free hierarchical (MAE: 0.44 m/s; RMSE: 1.45 m/s) than in the binary interval aggregation scheme (MAE: 0.45 m/s; RMSE: 1.34 m/s) lacking an interval layer for this metric. The flat schemes performs best (MAE: 0.3 m/s; RMSE: 0.95 m/s).



(a) Flat Aggregation

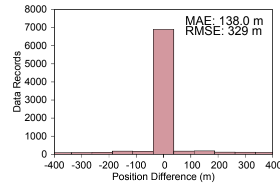


(b) Binary Interval Aggregation

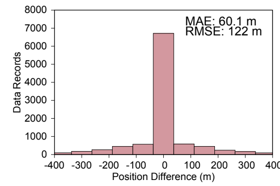


(c) Free Hierarchical Aggregation

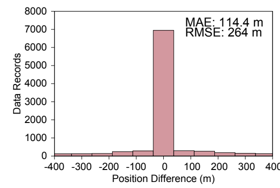
Figure 5. Aggregation Levels



(a) Flat Aggregation



(b) Binary Interval Aggregation



(c) Free Hierarchical Aggregation

Figure 6. Position Precision in Traffic Jam (4 - 7 min)

VI. Conclusions

In this paper we presented a framework to design adaptive aggregation schemes with flexible data structures, easy parametrization of the decision, fusion and dissemination components, and requirements on end-to-end data delivery within highly dynamic VANET environments. Three exemplary aggregation schemes with a different trade-off between complexity and resulting data precision were presented and evaluated in a simulation under extreme conditions. The aggregation framework is implemented relying on current standards and standard drafts for vehicular communication. It restricts the additional data load to given requirements on CBR and the aggregation schemes adapt their aggregation levels in real time even in congested traffic situations. Changing the CBR restrictions, adding new metrics or configuring the requirements on data precision and delay can be realized by changing only this specific parameter, other parameters will adjust automatically to deliver optimal data under given conditions.

References

[1] "ETSI EN 302 663 V1.2.1 - Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band," 2013.

[2] "Draft ETSI EN 302 637-2 V1.3.0 - Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service," 2013.

[3] S. Dietzel, F. Kargl, G. Heijenk, and F. Schaub, "Modeling In-Network Aggregation in VANETs," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 142–148, 2011.

[4] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "TAG: A Tiny Aggregation Service for Ad-Hoc Sensor Networks," *ACM SIGOPS Operating Systems Review*, vol. 36, no. SI, pp. 131–146, 2002.

[5] K. Ibrahim and M. Weigle, "CASCADE: Cluster-Based Accurate Syntactic Compression of Aggregated Data in VANETs," in *Proceedings of the IEEE Workshop on Global Communications (GLOBECOM '08)*, 2008, pp. 1–10.

[6] L. Wischhof, A. Ebner, and H. Rohling, "Information Dissemination in Self-Organizing Intervehicle Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 90–101, 2005.

[7] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode, "TrafficView: Traffic Data Dissemination using Car-to-Car Communication," *Mobile Computing and Communications Review*, vol. 8, no. 3, pp. 6–19, 2004.

[8] "Draft ETSI EN 302 636-4-1 V1.2.0 - Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 1: Media-Independent Functionality," 2013.

[9] "C2C-CC Distributed Congestion Control (DCC) White Paper for Day One Applications v1.0," 2013.