Discovering Groups of Signals in In-Vehicle Network Traces for Redundancy Detection and Functional Grouping *

Artur Mrowca[1,2], Barbara Moser[1], and Stephan Günnemann[2]

[1] Bayerische Motoren Werke AG, Knorrstr. 147, 80788 Munich, Germany
{artur.mrowca, barbara.moser}@bmw.de
[2] Technical University of Munich, Boltzmannstr. 3, 85748 Garching, Germany
guennemann@in.tum.de

Abstract. Modern vehicles exchange signals across multiple ECUs in order to run various functionalities. With increasing functional complexity the amount of distinct signals grew too large to be analyzed manually. During development of a car only subsets of such signals are relevant per analysis and functional group. Moreover, historical growth led to redundancies in signal specifications which need to be discovered. Both tasks can be solved through the discovery of groups. While the analysis of in-vehicle signals is increasingly studied, the grouping of relevant signals as a basis for those tasks was examined less. We therefore present and extensively evaluate a processing and clustering approach for semi-automated grouping of in-vehicle signals based on traces recorded from fleets of cars.

Keywords: in-vehicle · clustering · signal · redundancy detection

1 Introduction and Related Work

Modern vehicles communicate across multiple Electronic Control Units (ECUs) in order to run various functionalities. Those are implemented by multiple domains and are incrementally optimized throughout the development process of a car. With the growing demand for security, safety and entertainment more functionality is added and with this the complexity in in-vehicle networks increased. E.g. modern premium vehicles contain over 100 million lines of source-code onboard and, per function, up to 15 ECUs communicate with more than 2 million messages transmitted per minute. This communication between ECUs is defined via signals that are sent in defined messages. Signals resemble a dimension of information transmitted over time (e.g. GPS position).
In order to optimize and diagnose behavior in such in-vehicle systems during development, traces are recorded from in-vehicle networks of test vehicles and analyzed off-board (Fig. 1). One dimension of optimization is the refinement of specifications of the communication behavior of signals, as historical system

* This work was supported by the BMW Group.

growth led to redundancy in signal specifications. I.e. potentially identical and thus, redundant information is transmitted multiple times via distinct signals. This leads to more message collisions on in-vehicle networks, which result in loss of information or jam signals that can cause ECUs to fail. Thus, it is imperative to reduce the number of signals to improve safety and stability of in-vehicle systems. Further, the number and complexity of signals aggravates subsequent data analyses per domain, as it becomes increasingly difficult to identify signals

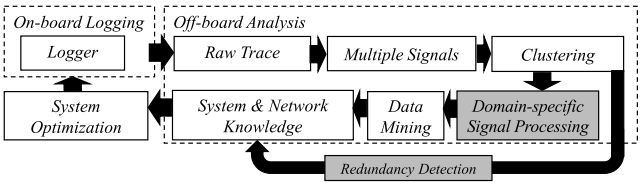


Fig. 1: Data is logged on-board and analyzed off-board. There functional groups and redundancy are detected. The scope of our contribution is marked grey here.

that belong together and need to be analyzed jointly. E.g. analyzing the correct functioning of the wiper may only include signals, such as the rain sensor or the wiper position, while other signals are needless to consider.
Finding such groupings is cumbersome and in general not possible to be done manually. Consequently, detection of redundant and interrelated signals needs to be performed automatically, e.g. by using unsupervised learning algorithms.
We therefore present and evaluate a Data Mining approach that allows for systematic clustering of interrelated signals using recorded in-vehicle network traces. Clustering in-vehicle signals is challenging due to several reasons. First, the signals to cluster are heterogeneous, i.e. data can be categorical, numerical, ordinal or binary. Massive amounts of traces are processed, with more than 10 million signals transmitted per minute. Also, different to classical scenarios the ratio of "number of target clusters" to "number of input samples" is high. Lastly, data is recorded as raw bytes which requires prior interpretation to achieve a data format that allows for subsequent clustering.
Related Work: A comparable preprocessing approach was introduced both in [2] and in [1]. However, in [2] features are extracted from multiple signals in order to classify them as normal or abnormal. Also, in [1] the focus is on finding causal relations between individual features of signals and fault types. Both, approaches group segments of signals, whereas we aim to group whole signals. Also, no heterogeneous, but rather numerical signals only are considered there. Grouping of signals was performed in [12], where supervised learning approaches were used to classify signals as internal (state of vehicle) and external (state of environment). However, for massive numbers of signals a supervised approach requires high labeling costs. To overcome this we investigate the possibility of an unsupervised scenario, where no labels are given. Also, we examine more than

two target classes. Many Data Mining approaches where applied to in-vehicle signals, most of which are focused on diagnostics. In [8] diagnostic neural networks are trained for fault classification and in [4] induction motor drive faults are detected using recurrent dynamic neural networks. More recently diagnosis in in-vehicle signals was done for anomaly detection, e.g. by using condition indicators [14]. CAN signals were used for predictive maintenence [11]. In [13] vehicle signals are used to predict compressor faults, and in [16] to model the remaining useful life time of batteries in trucks.
Moreover, in-vehicle signals were used in applications, such as detection of scenarios [18] or driver workload monitoring [15].
Most existing Data Mining tasks are based on a subset of relevant signals. While Data Mining on in-vehicle signals is well studied, less attention was paid to the detection of those relevant signals. But, the growing numbers of existing signals makes it essential to find groups of signals before such techniques can be applied or to optimize their performance. This is, as investigation of irrelevant signals causes misclassification in diagnosis and increases computational complexity. We are the first ones to in-depth investigate the systematic grouping of in-vehicle signals for the purpose of functional grouping and redundancy detection.
Contributions: First, we present a concept that detects groupings of automotive signals. This is done by reducing data to relevant features which allows for local inspection of signals towards redundancy detection and domain-specific grouping. Second, we evaluated the approach using 10 real world data sets of different characteristics. With this the dependence of window size and selected features on the clustering performance is evaluated. Third, the performance of 9 clustering approaches for the task of in-vehicle signal segmentation was inspected formally and experimentally. We show that using our approach systematic analyses of in-vehicle signals is enabled. Lastly, we discuss the influence of cluster parameterization on the granularity of the grouping. Fine granularity results in better performance towards redundancy detection. Coarser parameterization is better for detection of interrelated signals, that affect common functions.
Concept Overview: Our approach for grouping interrelated signals is shown in Fig. 2. During *Preprocessing* raw traces are interpreted and prepared with the strategy proposed in [7]. Next, *Feature Engineering* is used to extract and reduce feature vectors per signal. Lastly, *Clustering* is used to find groupings.

2 Preparing Traces

In this section first, automotive traces are introduced. Next, grouping of signals from such traces requires data extraction, preprocessing and feature engineering steps which are presented in the second part of this section.

2.1 Automotive Traces

Cars implement several functionalities. Those require ECUs, sensors and actuators to exchange information across its internal vehicle network (e.g. CAN bus).

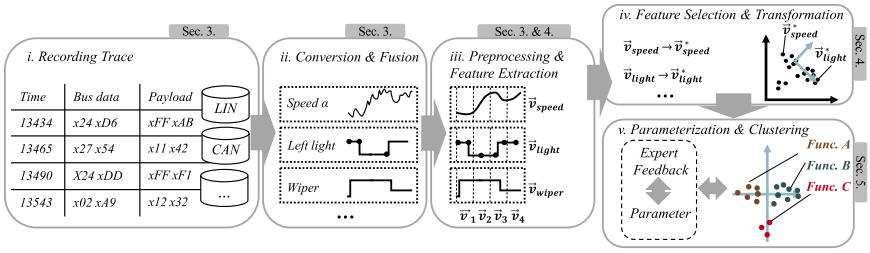


Fig. 2: Overview of the Data Mining approach with the respective sections detailing the process stated.

This information can be decoded to yield multivariate time-series of signals. I.e. each dimension represents a defined information in time. This is called a signal and shown in Fig. 2 *ii.* Signals can be categorical (e.g. car mode: driving/parking), binary (e.g. engine: on/off), ordinal (e.g. heating: level 3) or numerical (e.g. speed). The incremental growth of vehicular systems results in an increasing number of signals. E.g. one vehicle may contain up to ten thousand CAN signals. Manual grouping of interrelated signals has become intractable, which requires automated unsupervised algorithms as presented in this work.

2.2 Data Extraction and Cleaning

Data Extraction: Data is ingested as time stamped byte sequence and is converted to signals as shown in Fig. 2 *ii.* It contain billions of entries per vehicle. Those are stored in large-scale distributed database systems, such as Apache Hive and processed with distributed engines such as Apache Spark [3].
Data Cleaning: First, raw signals contain invalid entries which may result from invalid network states, such as jammed signals or defect ECUs. Those invalid entries are identified using network specifications and are dropped. Second, depending on its data types signals require different features to be extracted. We thus categorize signals as numerical, if more then a threshold number of values in that signal is numeric and as categorical else. Third, missing data is replaced via interpolation if data is numeric and via repetition of the last valid signal value else. Fourth, numerical data is smoothed to reduce noise resulting from disturbances in the network, e.g. using Exponential Moving Average filtering. Lastly, for better comparability numerical signals need to be within a common value range. Several approaches can be used for this. However, we found Interquartile Range normalization to achieve best comparability as the resulting signals allow to compare the shape of the signals rather than their absolute values.

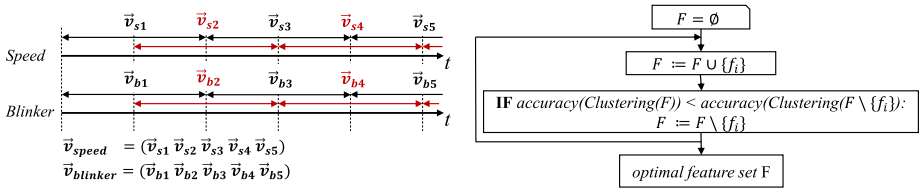
3 Feature Engineering

Here we describe feature extraction per signal, identification and transformation of features and formal evaluation of clustering algorithms for our scenario.

3.1 Feature Extraction

In the given scenario we assume that interrelated signals occur within common time ranges and change their value in similar time intervals. To capture such temporal-causal dependence in a feature vector, we propose the following approach.

Distance metrics: Comparing signals requires a distance metric. Common metrics are Euclidean, Dynamic-Time-Warping (DTW) [10] or Short Time Series (STS) distance [6]. Due to computational complexity increasing with the length of the time-series both DTW and STS are not suited here. Thus, we use Euclidean distance.



(a) Feature extraction from overlapping windows.

(b) Feature selection using forward backward search.

Fig. 3: Feature extraction and forward backward feature selection approach.

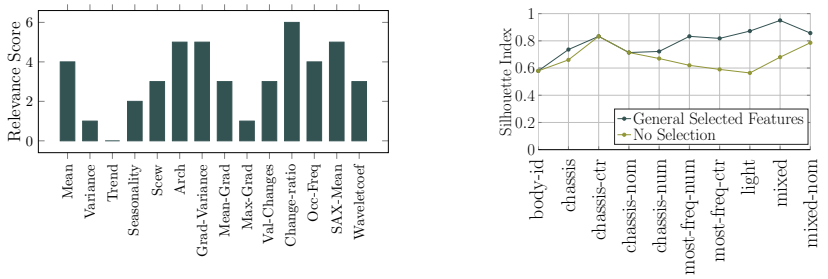
Extraction Approach: As shown in Fig. 3a, all signals are sliced according to overlapping windows. Per window w_i and per signal s_i a sub-feature vector $v_i = f(w_i, s_i)$ is extracted using $f : (w, s) \mapsto v$ and the signals data type $type(s_i)$. Signals s_i can be numerical $type(s_i) = num$ or categorical $type(s_i) = cat$. Depending on this data type, different characteristics (i.e. features) are extracted to represent the value behavior of the corresponding signal s_i. Thus, for each s_i the following features are extracted according to its $type(s_i)$. If $type(s_i) = num$, it is required to capture the shape of the signal per windows. This is done per window with features *mean*, *variance*, *skew*, *arch*, the *magnitude of the average slope*, *variance of the slope*, *maximal slope*, the *mean of the SAX symbol occurrences* and the *wavelet coefficients*. Those numerical features were chosen as they were successfully applied in similar tasks in [9] and [17].

If $type(s_i) = cat$ only information about occurrences and their value is available and thus, need to be extracted as appropriate features. In order to make numerical signals ($type(s_i) = num$) comparable to categorical once, numerical signals are symbolized in value using Symbolic Aggregate Approximation (PAA) [5]. Next, per signal identical repetitions are removed, which allows the numerical signals to be additionally treated as categorical. On those categorical and discretized numerical signals extracted features are the *number of times a value changed*, the *number of times a value occurred* and the *change ratio per window*, which is the number of value changes divided by the number of samples per

window. Thus, the change ratio is the weight of a window, i.e. the amount of change that occurred in it.
Choosing those features allows to compare nominal and numeric signals, while comparison among numerical signals is done on a more fine grained level using its numerical features. To now represent a signal s_i with identifier m, sliced in n windows, as a feature vector v_m, all subvectors v_{mi} are stacked as $v_m = (v_{m1}v_{m2}...v_{mi}...v_{mn})$. This representation captures temporal interrelation, as same dimensions represent same windows and value behavior is represented by each value in a dimension.

3.2 Wrapper-based Feature Selection

A classical approach to determine most important features is to use forward-backward search in a wrapper-based evaluation. I.e. the quality of a subset of features is evaluated on a validation data set using the clustering target (e.g. redundancy or function grouping) it is optimized for. As clustering is unsupervised, in this step a ground truth was defined for the training set, which is done manually by experts. As a optimization target the precision is used, i.e. ratio of signals that were correctly clustered (according to the expert).
With this, per data set a feature subset with maximal precision is found. To avoid overfitting the search is run on several data sets with various characteristics (e.g. ratio of numeric to categorical signals), yielding an optimal feature subset per data set. Next, all features are ranked by counting subsets that contain this feature, which ranks more general features that are valid for more data sets higher. The top ranked features are used for further processing (e.g. top 50 %) and are extracted per window. This process is depicted in Fig. 3b.



(a) Relevance score determined as number of optimal feature subsets in which a feature occurred.

(b) Clustering performance in terms of Silhouette index before and after the generalized feature selection is applied.

Fig. 4: Results of the experiments for the feature selection used.

3.3 Feature Transformation

The resulting feature vector is of high dimension, as for n windows and f features, the vector has $n \cdot f$ dimensions. However, the curse of dimensionality states the problem that with higher dimensions of feature vectors, feature vectors appear further away in terms of distance measures. Also, higher dimensions are computationally more expensive during clustering. That is why the number of dimensions needs to be reduced. For this we apply a two step approach.
First, per dimension the variance is computed to determine its amount of information. Low variance indicates less information per feature and window e.g. if a signal did not occur in a window. Therefore dimensions with variance smaller than a threshold are dropped. Second, dimension is reduced to a information maximizing space with a Prinicipal Component Analysis (PCA). As PCA is a linear transformation, inherent properties of each signal vector are conserved (e.g. Euclidean Distance). Also, only the most informative dimensions are used. The transformed feature vector is used for clustering.

4 Clustering

The nature of in-vehicle network signals renders not all clustering approaches equally applicable for the given use-case of redundancy and correlation detection. Therefore, we investigate the suitability of multiple approaches next.

4.1 Properties and Approaches

Desirable properties: Signals are heterogeneous, of huge size and yield very less signals per target clusters. Furthermore, even after reduction, data is of high dimension due to highly complex characteristics per signal. Consequently, a desirable property for redundancy detection is the possibility to parameterize the approach towards clusters of certain levels of granularity. E.g. at a higher level of granularity per wheel the four sensors of the rotational frequency should be grouped, while at a lower level both left wheel (front & back) and both right wheel sensors should be assigned to two separate groups. Moreover, the computational complexity needs to be still kept low as massive data is processed. Lastly, it is important to visualize the data to be able to verify the results of the clustering and to inspect the level of correlation between elements of clusters.
Clustering Approaches: Centroid-based clustering, such as k-Means and k-Medoids, represent classes relatively to a class centroid and iteratively find such optimal centers per class. In Expectation Maximization centroids can be Gaussian probability distributions, which are optimally fit over given data. In hierarchical clustering approaches groups are sequentially decomposed, which can be done bottom-up or top-down. Density based approaches (e.g. Density-Based Spatial Clustering of Applications with Noise (DBSCAN)) group data by assigning a radius around a data point and grouping overlapping neighborhoods. Raster-based approaches include WaveCluster, where a raster is defined

and points in the same raster are grouped. In Affinity propagation data points are passing messages which indicate the affinity to its neighbors to determine clusters. In Self-organizing maps (SOM) the space is divided in hexagons which approach each other iteratively to form groupings.

4.2 Suitability Analysis

Clustering approaches are not equally well suited to fulfill the properties stated in subsection 4.1 and are a trade-off between those. We compared the most important approaches formally in terms of their applicability to in-vehicle signal clustering with results shown in Table 1 and discussed here. An experimental evaluation of those approaches is given in sections 5 and 6.
Centroid-based: Granularity is settable as target clusters k. k-Means is in general suited for high dimensional data as prototypes are found as mean of all clusters and a separation is forced through k. But, only spherical clusters are possible which is contrary to signal feature vectors which can be grouped in any shape. k-Medoids and EM are less suited. In k-Medoids samples are part of the data set which shifts the centroid on a data point and thus, imbalances the center.
Hierarchical: Such approaches are independent of shape, as successive splitting or joining is performed based on neighborhoods. But, top-down clustering tends to split the biggest cluster more often. This results in many clusters of similar size which is not the target grouping in our scenario where cluster sizes may vary. Granularity can be parameterized on according splitting and joining rules.
Density-based: Those approaches allow for multiple granularity by setting the radius per data point, while they are independent of shape as neighboring elements are found using the radius. This radius can exist in any dimension leaving this approach to be well suited for clustering of signals.
Grid-based Those approaches allow for multiple granularity by setting the raster size and are independent of shape as the raster can be of any shape. Above that,

Table 1: Comparison of algorithms in clustering of in-vehicle signals. I.e. handling high-dim. data, detect clusters of any shape, allow multiple granularities of clusters, visual representation and computational complexity, with t iterations, maximal depth d, n examples and k classes.

Approach	High-dim. data	Complex-shapes	Multiple Granularities	Visualization	Complexity
k-Means	yes	no	yes	no	$\mathcal{O}(nkt)$
k-Medoids	no	no	yes	no	$\mathcal{O}(k(n-k)^2 * t)$
EM	no	no	yes	no	$\mathcal{O}(nk * t)$
DBSCAN	yes	yes	yes	no	$\mathcal{O}(n \log n)$
Agglomerative	yes	yes	yes	dendrogram	$\mathcal{O}(n^3)$
Top-Down	no	yes	yes	dendrogram	$\mathcal{O}(2^d * nkt)$
WaveCluster	indirect	yes	yes	no	$\mathcal{O}(n)$
Affinity Propagation	no	yes	no	no	$\mathcal{O}(n^2)$
SOM	indirect	yes	yes	map	$\mathcal{O}(n * t)$

high dimensional clustering is possible with the limitation that dimensions need to be restricted as e.g. in WaveCluster similar Wavelet coefficients will be too far away to be assigned in one clusters (due to curse of dimensionality). With the reduction to a sufficient number of dimensions and its low computational complexity those approaches are well suited.
Affinity Propagation: Here prototypes are data points themselves, leading to similar imbalance as in k-Medoids. However, common grouping is not dependent on cluster shape as the totality of points is considered for clustering.
Self-Organizing Maps: Due to small numbers of signals each hexagon is sparsely populated by data points making cluster detection difficult.
According to this formal evaluation we expect WaveCluster and DBSCAN to be most suited for the identification of groupings among in-vehicle signals.

5 Evaluation

Preprocessing and Clustering of in-vehicle signals requires appropriate parameterization in terms of windowing, feature engineering and selection of clustering algorithms. This is studied in this section.

5.1 Experimental Setup

Environment: Due to large raw traces, preprocessing and feature extraction were implemented on a cluster with 70 servers in Apache Spark. With the reduced data set the remaining steps (selection, transformation, clustering) were performed locally on a 64-Bit Windows 7 PC with an Intel® Core™i5-4300U processor and 8 GB of RAM using RapidMiner Studio, Python's Data Mining stack and R.
Datasets: The statistics of our data sets are shown in Table 2. To cover most characteristics of automotive in-vehicle network traces we evaluated the approach on 10 test data sets that are different in terms of signal types (e.g. chassis-nom vs. chassis-num), data points per type, signal number, association to one (e.g. chassis) or multiple (e.g. mixed) functions and resemble different excerpts of a journey. The target of our evaluation is the grouping of signals in terms of their assignment to similar functions. All approaches were parameterized per data set such that the true number of clusters is achieved and the best possible grouping (according to the expert) within this clustering is reached.
Evaluation Criteria: The approach is evaluated in terms of clustering quality. *Accuracy:* Accuracy is the number of samples $n_{correct}$ correctly clustered in relation to the total number of samples in the data set $n_{dataset}$ given as $acc = \frac{n_{correct}}{n_{dataset}}$. Here, the assignment of reference cluster labels to each signal as a ground truth is done manually by experts. *Silhouette index $s(i)$:* Measures a clustering assignment per data point i in terms of degree of affinity to its assigned cluster relatively to all other clusters. I.e. $a(i)$ as distance of i to all element within its cluster, $b(i)$ as average distance to all data points in all other clusters. It is optimal for $s(i) = 1$ and defined as $s(i) = \frac{b(i)-a(i)}{max\{a(i),b(i)\}}$.

Table 2: Statistics of the datasets: total number and proportions of numerical and nominal signals, data points per set, recorded part of journey. Here, small subsets are used for evaluation, while in practice thousands of signals are considered.

Set	Signals (tot[num/nom])	Datapoints (tot[num/nom])	Part of journey
body-id	38 [1/37]	2251 [89/2162]	complete
chassis	53 [18/35]	9999 [9896/103]	start
chassis-nom	35 [0/35]	9896 [0/9896]	start
chassis-num	18 [18/0]	103 [103/0]	start
chassis-ctr	12 [11/1]	10000 [9999/1]	mid
most-freq-num	24 [24/0]	12508 [12508/0]	start
most-freq-ctr	22 [19/3]	11773 [11765/8]	mid
light	39 [6/33]	10055 [2941/7114]	start
mixed	25 [12/13]	69402 [69339/63]	start
mixed-nom	13 [0/13]	9509 [0/9509]	start

5.2 Window Size

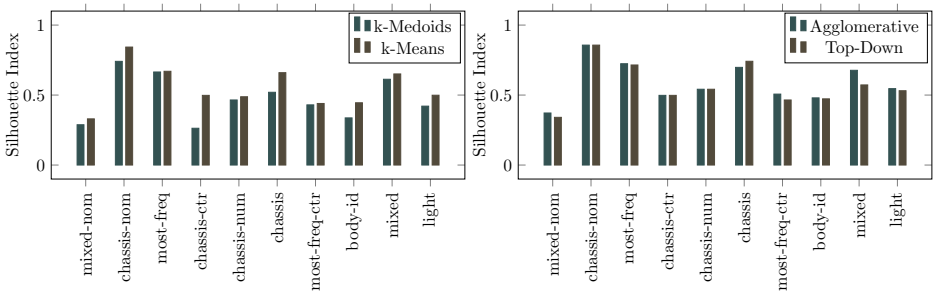
Setup: After preprocessing we split each signal in windows with 50 % overlap, extract all features, transform them and perform clustering. Per data set the window size is increased successively from 0.1 seconds to 5000 seconds and the performance is measured in terms of accuracy. From this we identify the window size with highest accuracy as optimal. We evaluate k-means for clustering here, while other approaches yielded similar results. The results are shown in table 3.
Results: If the window is too small patterns relevant for features are overseen, while for big windows feature details are simplified away. Also, as can be seen in table 3 less frequently changing signals, e.g. with a higher number of nominal signals, require bigger windows , e.g. in body-id, light and mixed-nom, as those signals do change less often. If more frequently changing numerical signals need to be clustered smaller windows appear to be optimal which is the case in chassis, chassis-ctr, most-freq-ctr and mixed.

Table 3: Experimentally determined optimal windows per data set in seconds.

body-id	chassis	chassis-ctr	most-freq-ctr	light	mixed	mixed-nom
128.8	3.5	79.8	1.8	533.6	1.5	2147.7

5.3 Feature Selection

Setup: Feature selection is performed as described in sec. 3.2 where feature subsets are successively searched and evaluated once per data set using clustering performance as optimization target. To find features that generalize over all data sets, we measured the number of times a feature was included in the

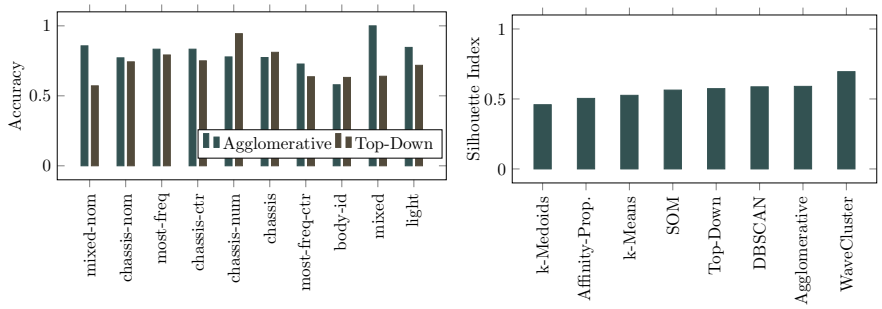


(a) Comparison of centroid-based algorithms in terms of Silhouette index.

(b) Comparison of hierarchical algorithms in terms of Silhouette index.

Fig. 5: Results of experiments for centroid and hierarchical approaches.

optimal feature subset. k-Means was again used for clustering. The results are shown in Fig. 4a. The performance gain of our generalized feature selection was measured before and after the ranking selection, with results shown in Fig. 4b. **Results:** It can be seen that for the numerical characteristics best features are the mean, skew, arch, as well as the variance and magnitude of the gradient. This shows that the fine granularity of numerical signal characteristics requires to capture noise, value and shape characteristics. For nominal characteristics all nominal features were suited. This shows that the frequency and type of a nominal/discretized numerical signal can be captured. Further, this resembles our assumption that in-vehicle signals are correlated, when they occur and change their value together. As Fig. 4b depicts a performance gain of up to 20 % (e.g. at light data set) is achieved with this approach. Notably, all data sets show an improvement after the generalized selection.



(a) Comparison of hierarchical algorithms in terms of Accuracy.

(b) Average Silhouette index over all data sets.

Fig. 6: Results of experiments for hierarchical approaches and all data sets.

5.4 Comparison of Clustering Algorithms

As stated in sec. 4 the characteristics of in-vehicle signals require clustering algorithms that can handle high-dimensionality, different granularities and have low computational complexity.

Setup: To examine the suitability of different algorithms for grouping of in-vehicle signals, we evaluated k-Means, k-Medoids, DBSCAN, Agglomerative, WaveCluster and SOM clustering approaches on all data sets in terms of clustering quality. This is done by using the optimal feature subset as selected by our feature selection approach, parameterization with expert feedback and by consequent application of the clustering approaches.

We first compared Agglomerative against Top-Down clustering and k-Means against k-Medoids, to evaluated the characteristics of those sub types in terms of applicability to in-vehicle signals. This is followed by a general experimental comparison of all approaches.

Results - Sub types: As illustrated in Fig. 5a, among centroid-based approaches k-means performs better than k-Medoids. This is, as taking the mean

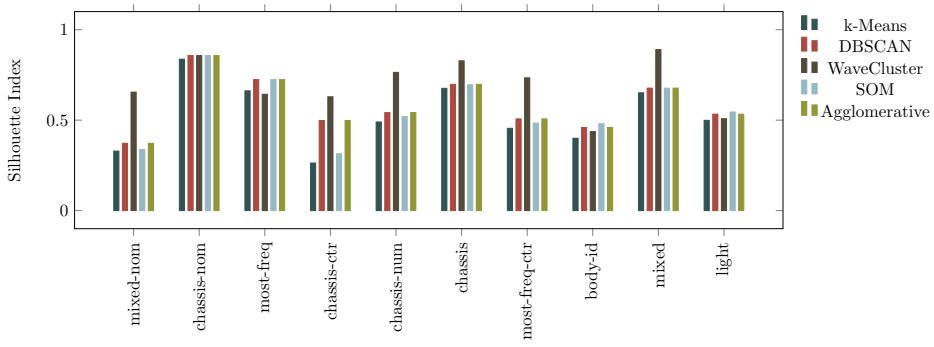


Fig. 7: Silhouette index per data set and clustering algorithm.

among signals for clustering avoids a shifting bias as stated in sec. 4.

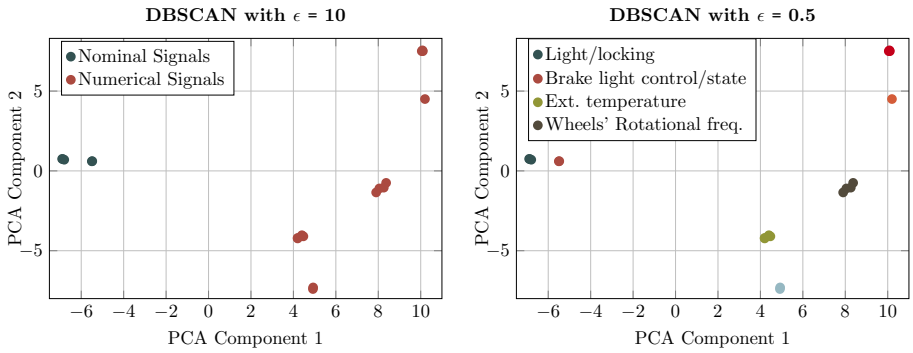
Results - Hierarchical: Among hierarchical approaches Agglomerative clustering results in better accuracy in 90 % and in better Silhouette index in 70 % of all cases which is shown in Fig. 5b and Fig. 6a. The best centroid-based and hierarchical approaches are evaluated with further clustering approaches giving results shown in Fig. 6b and Fig. 7.

Results - Overall: As depicted in Fig. 6b, DBSCAN, Agglomerative clustering and WaveCluster works best if a data set contains mixed characteristics (i.e. different signal types, proportions of nominal to numerical, etc.) combined. Also, in those cases centeroid-based approaches perform worse. This confirms our expectations and formal analysis of the approaches in sec. 4. Further, as depicted in Fig. 7, WaveCluster performs best on 80 % of all data sets and shows solid

results in the remaining 20 %. Thus, this approach seems best suited for our scenario. This is because extraction of Wavelet coefficients enables to well capture both fine and coarse grained properties of signals equally. Also, as described in sec. 4 WaveCluster can well represent the shape and the data's high dimension. Similarly, DBSCAN and Agglomerative Clustering are well suited to capture those properties. However, the latter approach is biased in that it tends to find clusters of nearly similar sizes which is not given in all test sets.
As deduced in sec. 4 SOM and k-means perform slightly worse, as dimensions are reduced in SOMs and k-means cannot capture varying cluster shapes.
Conclusion: All clustering approaches have solid results in terms of cluster quality. This shows that the proposed processing and clustering approach is well suited for groupings of in-vehicle network signals. WaveCluster and DBSCAN perform best due to their ability to capture most of the heterogeneous characteristics included in such signals. As described an optimal window size depends on the structure of the processed data and thus, needs to be determined. Further, features subsets as discussed in sec. 5.3 allow for good generalization when clustering in-vehicle signals.

6 Case Study

In this section we exemplary show how our approach can be used to detect signal redundancies and group signals of common functions.
Setup: For this case study a realistic data set was used. After the preprocessing of sec. 2 this data set contains 419 signals and (after reduction) 20 026 065 data points recorded from one vehicle over eight days. This processing is again implemented on the Hadoop system described in 5.1, while the resulting reduced data is processed locally. *Preparation:* An optimal window size of 17.7 seconds was found with 7 477 windows of 50 % overlap. Per window the features found in sec. 5.3 were used resulting in more than 10 000 dimensions per signal. Reduction to less dimensions is done by filtering for dimensions with a variance bigger 0.3 and a successive PCA, resulting in 80 dimensions per signal which can be used for local clustering. For clustering we used DBSCAN.
Analogy Detection by cluster inspection: Depending on the parameterization of the clustering, granularity of the target can be set. I.e. if redundancies need to be detected a more fine grained target parameterization is required, while the opposite holds for grouping according to functions. This is illustrated in Fig. 8b where a coarse grouping separates signals with different data types and finer clustering extracts signals of similar functions. Finding an appropriate granularity is done through expert feedback. The extracted clusters can be inspected and successively parameterized towards a good target clustering. Experts can then asses the grouping results, e.g. decide whether a grouping signifies a redundancy.
Results: With our approach redundancies and related signals were found in the analyzed data set. E.g. we found redundancy among speed signals and signals representing the time. The further were the speed signal for the speedometer, the state of the speed in horizontal direction and the speed of the car's mass center.



(a) Result of signal clustering with DBSCAN and ϵ = 10

(b) Result of clustering with DBSCAN and ϵ = 0.5.

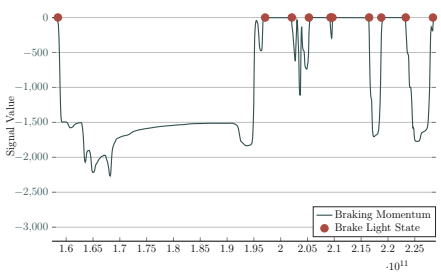
Fig. 8: Clustering with DBSCAN at different granularities by varying ϵ. The legend shows the signals which were grouped. E.g. the locking signal turns on the light when the car is closed.

Those are all identical as they measure the vehicle speed and thus, can be reduced to one signal in future architectures of the vehicle. An example for detected groups of similar functions are signals related to the braking function which were grouped (see Fig. 9b, red cluster). It shows that the brake light state, state of the driver braking, braking momentum on the wheels and the target braking momentum resulting from the driver pressing the pedal are grouped. In particular as Fig. 9a shows, with our approach nominal signals were grouped together with related numerical signals. Further examples of discovered functional groups are signals for automated parking (e.g. parking space, driver intervention), battery state (e.g. battery capacity, state of charge) or constant signals (e.g. air pressure, state of the belt buckle). Thus, the proposed approach is well suited to find signals of common functionality, which in turn enables successive domain-specific analyses of relevant signals and Data Mining applications on related signals.

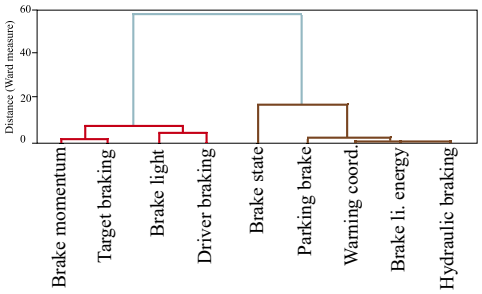
7 Conclusion

A Data Mining approach for systematical detection of groupings of in-vehicle signals is presented. In particular numerical and nominal signals are made comparable for clustering and massive data is reduced early to a lower dimensional space. We evaluated the optimal window size, a general feature subset and the suitability of different clustering algorithms for clustering of in-vehicle signals. A case-study showed that redundancies and signal groupings can successfully be found with this approach.

Lessons learnt: We found the optimal window size depends on the character of the data set and on the features. We showed that grouping heterogeneous signals



(a) The numerical signal "Braking Momentum" and the nominal signal "Brake Light State" are grouped with our approach.



(b) Dendrogram illustrating hierarchical clustering at various granularities. I.e. branches resemble possible groupings. E.g. one possible granularity is shown in red and brown.

Fig. 9: Excerpt of the results of the case-study for signal clustering.

is possible by assigning nominal features to numerical signals. To handle massive traces, extraction, selection and transformation steps are performed on a cluster, while locally a tractable lower dimensional vector is analyzed with expert feedback. We formally and experimentally demonstrated, that automated clustering of in-vehicle networks is possible. There both fine and coarse grained structure needs to be captured, which is best possible with WaveCluster. The introduced approach allows for future automation of tasks like anomaly detection or situation detection. System optimization is enabled by detection of redundancy and understanding which signals are sent jointly. Future work includes reduced Feature Engineering or further automation of the process. Also, a knowledge base could be designed to capture groupings, that are verified by experts and clustering optimized by exploiting this base. Further parameter evaluation needs to be performed to get a deeper insight on the mapping between parameters and redundancy or functional grouping. With the presented clustering approach we set the basis for future in-vehicle signal analysis of modern vehicles.

References

1. Crossman, J., Guo, H., Murphey, Y., Cardillo, J.: Automotive signal fault diagnostics. i. signal fault analysis, signal segmentation, feature extraction and quasi-optimal feature selection. IEEE Transactions on Vehicular Technology **52**(4), 1063–1075 (jul 2003). https://doi.org/10.1109/tvt.2002.807635
2. Guo, H., Crossman, J., Murphey, Y., Coleman, M.: Automotive signal diagnostics using wavelets and machine learning. IEEE Transactions on Vehicular Technology **49**(5), 1650–1662 (2000). https://doi.org/10.1109/25.892549
3. Hadoop, A.: Hadoop (2009)
4. Kim, K., Parlos, A.: Induction motor fault diagnosis based on neuropredictors and wavelet signal processing. IEEE/ASME Transactions on Mechatronics **7**(2), 201–219 (jun 2002). https://doi.org/10.1109/tmech.2002.1011258

5. Lin, J., Keogh, E., Lonardi, S., Chiu, B.: A symbolic representation of time series, with implications for streaming algorithms. In: Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery - DMKD 03. ACM Press (2003). https://doi.org/10.1145/882082.882086
6. Möller-Levet, C.S., Klawonn, F., Cho, K.H., Wolkenhauer, O.: Fuzzy clustering of short time-series and unevenly distributed sampling points. In: Advances in Intelligent Data Analysis V, pp. 330–340. Springer Berlin Heidelberg (2003). https://doi.org/10.1007/978-3-540-45231-7_31
7. Mrowca, A., Pramsohler, T., Steinhorst, S., Baumgarten, U.: Automated interpretation and reduction of in-vehicle network traces at a large scale. In: Proceedings of the 55th Annual Design Automation Conference on - DAC 18. ACM Press (2018). https://doi.org/10.1145/3195970.3196000
8. Murphey, Y.L., Masrur, M., Chen, Z., Zhang, B.: Model-based fault diagnosis in electric drives using machine learning. IEEE/ASME Transactions on Mechatronics **11**(3), 290–303 (jun 2006). https://doi.org/10.1109/tmech.2006.875568
9. Nanopoulos, A., Alcock, R., Manolopoulos, Y.: Feature-based classification of time-series data. International Journal of Computer Research **10**(3), 49–61 (2001)
10. Niennattrakul, V., Ratanamahatana, C.A.: On clustering multimedia time series data using k-means and dynamic time warping. In: 2007 International Conference on Multimedia and Ubiquitous Engineering (MUE 07). IEEE (2007). https://doi.org/10.1109/mue.2007.165
11. Nowaczyk, S., Prytz, R., Byttner, S.: Ideas for fault detection using relation discovery. In: The 27th annual workshop of the Swedish Artificial Intelligence Society (SAIS); 14-15 May 2012; Örebro; Sweden. pp. 1–6. No. 071, Linköping University Electronic Press (2012)
12. Prytz, R., Nowaczyk, S., Byttner, S.: Towards relation discovery for diagnostics. In: Proceedings of the First International Workshop on Data Mining for Service and Maintenance - KDD4Service 11. ACM Press (2011). https://doi.org/10.1145/2018673.2018678
13. Prytz, R., Nowaczyk, S., Rgnvaldsson, T., Byttner, S.: Predicting the need for vehicle compressor repairs using maintenance records and logged vehicle data. Engineering Applications of Artificial Intelligence **41**, 139–150 (may 2015). https://doi.org/10.1016/j.engappai.2015.02.009
14. Raptis, I.A., Sconyers, C., Martin, R., Mah, R., Oza, N., Mavris, D., Vachtsevanos, G.J.: A particle filtering-based framework for real-time fault diagnosis of autonomous vehicles. In: Annual Conference of the Prognostics and Health Management Society (2013)
15. Taylor, P., Griffths, N., Bhalerao, A., Popham, T., Zhou, X., Dunoyer, A.: Redundant feature selection for telemetry data. In: Lecture Notes in Computer Science, pp. 53–65. Springer Berlin Heidelberg (2014). https://doi.org/10.1007/978-3-642-55192-5_5
16. Voronov, S., Jung, D., Frisk, E.: Heavy-duty truck battery failure prognostics using random survival forests. IFAC-PapersOnLine **49**(11), 562–569 (2016). https://doi.org/10.1016/j.ifacol.2016.08.082
17. Wang, X., Smith, K.A., Hyndman, R.J.: Dimension reduction for clustering time series using global characteristics. In: Lecture Notes in Computer Science, pp. 792–795. Springer Berlin Heidelberg (2005). https://doi.org/10.1007/11428862_108
18. Zheng, H., Zhang, H., Meng, H., Wang, X.: Qualitative modeling of vehicle behavior for scenario parsing. In: 2006 IEEE Intelligent Transportation Systems Conference. IEEE (2006). https://doi.org/10.1109/itsc.2006.1706813