On Systematically Exploring the State Space for Events with SFCs

Bachelor of Science Thesis in Software Engineering and Management

Younis Akel
Gianmarco Iachella

Department of Computer Science and Engineering
UNIVERSITY OF GOTHENBURG
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2023

In the process of developing autonomous driving systems (AD systems), ensuring safety remains a constant and continuous priority. Scenario-based testing is a popular approach to guarantee the safety of AD systems, which however meets challenges in terms of diversity and efficiency of generation, as well as evaluating existing test case datasets in terms of their coverage of the different variations of a particular maneuver. This research aims to support approaches for diverse and efficient scenario-based testing generation through the application of space-filling curves in determining the state space of specific vehicle maneuvers. The research goal is addressed through the use of design science, in which we produce an artifact consisting of a theoretical approach that acts as an initial foundation for an algorithm that efficiently generates diverse variations of specific maneuvers. An important discovery made in this research is understanding how SFC encoding impacts the way permutations are generated, and the implications of this in terms of what is required to ensure plausible maneuvers are permuted. This research mainly provides an initial theoretical approach to generate permutations of a limited subset of lane change maneuvers based on defined constraints.

Supervisor: Christian Berger
Examiner: Miroslaw Staron

University of Gothenburg
Chalmers University of Technology
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Department of Computer Science and Engineering
UNIVERSITY OF GOTHENBURG
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2023

On Systematically Exploring the State Space for Events with SFCs

Younis Akel
Gianmarco Iachella

Abstract—**In the process of developing autonomous driving systems (AD systems), ensuring safety remains a constant and continuous priority. Scenario-based testing is a popular approach to guarantee the safety of AD systems, which however meets challenges in terms of diversity and efficiency of generation, as well as evaluating existing test case datasets in terms of their coverage of the different variations of a particular maneuver. This research aims to support approaches for diverse and efficient scenario-based testing generation through the application of space-filling curves in determining the state space of specific vehicle maneuvers. The research goal is addressed through the use of design science, in which we produce an artifact consisting of a theoretical approach that acts as an initial foundation for an algorithm that efficiently generates diverse variations of specific maneuvers. An important discovery made in this research is understanding how SFC encoding impacts the way permutations are generated, and the implications of this in terms of what is required to ensure plausible maneuvers are permuted. This research mainly provides an initial theoretical approach to generate permutations of a limited subset of lane change maneuvers based on defined constraints.**

Index Terms—**Autonomous driving systems, scenario-based testing, space-filling curve (SFC), characteristic stripe patterns (CSP), permutations, state-space, areas of interest (AOI)**

I. Introduction

A. Background

In the development of autonomous driving systems (AD systems), safety is an ongoing priority and consideration [1], [2], [3]. Robust testing allows for early identification of issues that could pose risks in real-world scenarios [4], which contributes towards ensuring safety. A commonly used method for testing AD systems is scenario-based testing [3]. Scenario-based testing can be summarized as using scenarios from end-user perspectives to test software. For example, making a hard turn on a frozen street is a safety-critical scenario worth testing. For each of these scenarios, system-specific tests can then be derived [4].

However, challenges related to testing are still present in the development of AD systems. These challenges vary from being data-related (i.e., not having enough realistic data on extreme edge cases), to being related to testing how AD systems handle cases of failure during a real-world operation [5]. Furthermore, identifying when a test case set has covered all, or at least all important scenarios, is also ongoing in the research related to scenario-based testing [6]. Due to these ongoing challenges, a gap was identified in the testing of AD systems.

A common approach in developing AD systems is Data-Driven Development, which can be summarized as the storage and processing of large amounts of multi-dimensional data for the development of software. However, the multi-dimensional form in which sensor data is stored proves to be computationally costly in terms of space and time complexity. The cost in terms of space is related to the variety of sensors and large amounts of multi-dimensional data. Consequently, querying and analyzing such data for specific traffic situations or maneuvers represent a challenge in regards to time complexity. A recent approach to this challenge is using the concept of space-filling curves (SFCs) in terms of converting multi-dimensional data into a single dimension [7], [8], whilst retaining spatial information, resulting in a very efficient way for data access and processing. Along with improving efficiency of querying multidimensional data, depending on how the single-dimension SFC representation of multidimensional data is visualized, the usage of SFCs may enable an improvement and different approach in research related to testing AD systems - this will be further discussed in Section I-B.

B. Problem Domain & Motivation

Due to the safety-critical aspect in AD systems, it is vital to ensure robust testing - implying that a variety of data is not only needed on multiple events, but also on many variations of the same event. For example, the event of a roundabout consists of many variations, such as the speed of the turn and diameter - all of which pose different safety risks. An identified gap and ongoing challenge in the research of AD scenario-based testing, is efficiently generating a diverse test case dataset consisting of as many safety-critical scenarios as possible [6]. The authors also denote that current methods of scenario generation search for the best scenario that conforms to requirements, causing a decline in diversity leading to a risk of over-fitting and testing a system against very similar and limited scenarios [6].

According to Ding et al. adversarial training using generated scenarios is an efficient way to train AD systems via the adversarial training framework [6]. However, an issue with this approach is the imbalance of data diversity found in testing datasets, leading to risks of developing an AD system too specific to this limited dataset [6]. This shows a gap not only in how such scenarios can be generated, but also denotes a gap in utilizing such scenario generation to verify existing datasets in terms of how diverse they are.

By exploring the state-space of the single dimensional representation of sensor data on SFCs, the possibility of identifying specific events through patterns arises. With the

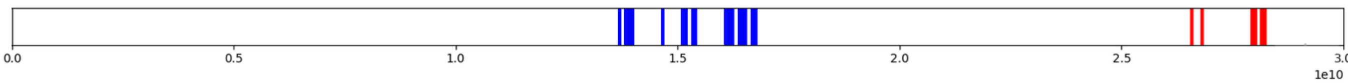

Fig. 1. Screenshot of a CSP on the single dimensional SFC representation of multidimensional data [9]

conversion of multi-dimensional car maneuver data into a single dimension via SFCs, pattern clusters emerge - more specifically, Characteristic Stripe Patterns (CSPs) are formed. An illustration of a CSP can be seen in Figure 1. CSPs from the SFC are correlated to car maneuver data. Identifying properties of CSPs such as their structure and spread would enable the permutation of events, such that data consisting of many variations of the same event can be generated systematically. This systematic generation would enable for generating a diverse set of variations of particular scenarios ensuring that safety-critical variations of a particular scenario are included, as well as to provide scenarios to evaluate and compare whether existing test case datasets are representative to all possible maneuver variations, and save time in terms of being able to artificially generate events.

C. Research Goal & Research Questions

The purpose of this study is to provide an initial theoretical foundation to systematically generate all the possible and relevant permutations of a particular event through the analysis of its CSPs' single-dimensional representation. This would act as a base on which further research can be conducted to eventually implement an algorithm that can vary multiple parameters within an event in an efficient manner. The aforementioned algorithm would support generating more diverse test case scenarios, as well as output reference datasets to which existing test datasets can be compared and evaluated against.

From the overall goal, as well as identified gap in the research, the following Research questions have been defined:

- RQ1: What are similarities and differences in characteristic stripe patterns (CSPs) on SFCs for similar events?
- RQ2: What properties of CSPs are necessary and sufficient to estimate dimensions of the state space for certain events and scenarios?
- RQ3: How can all possible event descriptions of a maneuver be systematically permuted in an efficient way?

RQ1 will initiate an analysis on CSPs of SFCs for similar events. This analysis will enable the identification of what aspects of CSPs could indicate specific events.

RQ2 focuses on identifying which properties are necessary and sufficient in determining an estimation for the state space for certain events and scenarios. For example, if we want to generate artificial data that represents lane change events, RQ2 would assist in providing which specific characteristics we may need to maintain such that our generated events are considered within its respective state space.

RQ3 focuses on a theoretical approach to produce plausible lane change permutations. The theoretical approach contributes to being able to generate a wider variety of test cases for particular events, as well as evaluating existing test cases. RQ3 acts as an initial step to achieving an algorithm that can generate all permutations of a particular event in an efficient manner, and therefore has constraints in terms of what it permutes (described further in Section IV). The success of the research is measured based on its performance in terms of time complexity, its completeness in terms of the types of lane change variations it can cover, and how extendable it is for future research.

We also initially had a fourth Research question with its purpose being to act as a metric to evaluate existing test case datasets to the datasets that would be produced by RQ3. Initially, we had intended to have a functional algorithm to answer RQ3, but due to a discovery made and discussed in Section VI-D1 paired with time constraints, we made a scope reduction in making a theoretical approach for the RQ3 and deem RQ4 as out of scope. Although, this does not negate the importance of RQ4, and hence we discuss its purpose as follows: RQ4 consisted of measuring the representativeness of real world test case data sets to all possible events generated by RQ3. This measurement would target the issue of evaluating whether existing datasets are encompassing enough to all variations of a particular real-world scenario in terms of how well a dataset covers rarer scenarios, such as safety-critical scenarios [6].

Lastly, throughout this study we will be using the terms 'events' and 'scenarios' interchangeably to denote a specific maneuver such as a lane change or roundabout turn.

D. Contributions

This research contributes to the state-of-the-art knowledge in AD system scenario-based testing by providing:

- A theoretical approach to contribute to computationally efficient diverse test case generation, through the application of SFCs for encoding of sensor data related to specific vehicle maneuvers
- The above would further contribute to evaluating existing datasets, contributing to the improvement and robustness of existing test case datasets in the domain of scenario-based testing in AD systems.
- A prototype algorithm for permutating lane changes via Morton values [10]. This prototype is described in Section IV-C
- Contributions to a repository for analysis of CSPs tailored for vehicle maneuver data [11]

E. Scope

The theoretical approach artifact has constraints applied to it that are described in Section IV. The theoretical approach is an initial step to the ultimate goal of an algorithm that can generate plausible permutations of all variations of a particular maneuver in an efficient manner. The approach would allow for extensions to be made to it and to allow for constraints to be made more lenient whilst maintaining the same theoretical approach.

F. Structure of the Article

The structure of this article is as follows:

- Section I - Introduction
- Section II - Related Work
- Section III - Methodology
- Section IV - Artifact
- Section V - Results
- Section VI - Analysis & Discussion
- Section VII - Conclusion and Future Work

II. Related Work

A. Use cases of SFCs

SFCs are widely used in several domains, mainly due to their characteristic data storage and querying efficiency. This section lists uses of SFCs in various areas, with the objective of delving into examples and evidence that share intuitions with the subject field of our study.

One relevant example is for SFCs that are used to classify malware into categories to improve antivirus programs [12]. The authors use existing research on the image representation of malware (which is used for classification into categories) and improve it by applying the concept of SFCs. They are essentially building on existing work in terms of categorization of image fingerprints of malware, upgrading the generation step of said fingerprints by using SFCs. Here we can see a parallelism with our contribution (i.e., malware as a specific maneuver, malware categories as maneuver categories, and antivirus programs as AD systems), in that we do not focus on the characteristics of categorization of maneuvers, rather in providing a way to efficiently generate examples of maneuvers of specific categories, to improve the efficiency of scenario-based testing in AD systems.

Martinez-Rubi et al. describe the query optimization of data management systems with the implementation of SFCs [13]. The authors present different approaches to SFC usage, and report the improved scalability of queries within growing multi-dimensional datasets. In their contribution, they point out the need for a feature of "Level of Detail" in future work that is related to data management and querying. For example, they describe a possible percentage of points to be specified within a data region. With regards to this necessity, we identify a potential point of development, since it is part of our artifacts to produce and measure data by adjusting the accuracy of the output, hence providing an example implementation that features the aforementioned "Level of Detail".

Although the research mentioned above makes wide use of SFCs, there seems to be a lack of application of the concept of CSPs and, more specifically, the intuition of permutations of CSPs. This could potentially be due to the different nomenclature that can be used to refer to such concept. Nonetheless, our research aims to address this gap, particularly with RQ1, RQ2, and RQ3. With our work we aim to introduce, use, analyze and report the improvements that CSPs bring to AD systems, with the vision that such concept can be used in other domains.

B. Scenario-based testing

Although Scenario-based testing was introduced and described in Section I-A, this section aims to discuss the literature available related to the state of the art in the domain, and aspects in which we identify areas that our research may contribute towards.

Scenario-based testing includes creating scenarios of real-world situations from which tests can be derived. This form of testing allows for AD systems to be evaluated in their performance and safety management in situations with varying risk levels. An ongoing area in the research of scenario-based testing is on improving the diversity, quality, and efficiency in generating realistic scenarios to be used in scenario-based testing [6]. Furthermore, in the existing literature, an emphasis is made on generating datasets with highly-critical scenarios [14].

Cai et al. report the state-of-the-art and current challenges of scenario-based testing through a survey on existing literature [15]. Importantly, the authors mention the lack of contributions with regards to measuring when enough scenarios are covered, as they point out that there are infinite possible scenarios to be generated, and no current solution to this "dilemma". With our work, we aim to contribute in addressing such aspect by focusing on providing a thorough, yet feasible, initial foundational approach to the generation of *all* possible scenarios for a given maneuver. Moreover, future work that would implement an algorithm based on our theoretical approach could address the aforementioned "coverage measurement issue" by providing guidelines on how existing scenario-based test datasets can be evaluated.

With regards to the diversity of scenarios generated, Bernhard et al. propose guidelines for diversity optimization, by classifying specific maneuvers into levels of safety concern [16]. As a contribution to this identified issue, our theoretical approach would provide the foundations to an algorithm that includes permutations of all scenarios of a lane change maneuver, including safety-critical scenarios. Future research could build upon the theoretical approach, which is extensible to covering different maneuvers such as ones deemed safety-critical - this is discussed further in VI-D3.

III. Methodology

Design Science Research (DSR) will be used in this research. DSR includes solving problems via creating and designing artifacts, as well as evaluating them [17]. The guidelines discussed by Peffers et al. show one approach on how DSR can be structured with activities [18]. The methodology typically contains an iterative aspect, which makes use of the first-hand knowledge gained from designing, creating and evaluating the artifact [18]. DSR is the main overall methodology and will contain the literature review methodology for gathering knowledge about the research domain. A literature review can be summarized as the analysis and synthesis of existing literature to provide an understanding of a topic, as well as to provide a base in terms of demonstrating that the research being conducted contributes to the domain knowledge [19]. In particular, the authors describe a three-stage process of effective literature review, which will be used to guide the review [19]. As the main methodology and goal of this research is to create two artifacts, the literature review will be conducted with the main goal being to gather knowledge required to develop the artifacts.

DSR is a good fit to solving the research questions due to an artifact being the output of this research, in particular: RQ3 is the RQ in which an artifact will be the outcome, whilst RQ1 and RQ2 could be considered more as identifying patterns and aspects of the CSPs required for RQ3 - suggesting that RQ1 and RQ2 occur earlier in the DSR iteration. In relation to the research by Levy et al. [18], this was our implementation of the activities that were carried out each cycle:

activity 1: Problem identification and motivation - At the beginning of each iteration, we used this activity to discuss the problems to address, in the context of the current cycle and the state of the artifacts. Existing literature upheld our reasoning and helped us in motivating our conclusions. Reporting the iterations of this activity was relevant to justify the solutions proposed, as well as to understand the adjustments that we made as the artifacts evolved.

activity 2: Define the objectives for a solution - From the observations made during the previous activity III, it was possible to define the objectives for a solution. The objectives considered the scope of the project as well as their feasibility with the resources available.

activity 3: Design and development - The design and development aspect encompassed the creation of the artifact from RQ3. For the first cycle, more focus was placed on the design aspect, to ensure that our understanding of the artifact met an acceptable level. As we progressed to Cycle 2, we adjusted the balance between design and development to guarantee the production and improvement of the artifact in subsequent cycles. During Cycle 2, we had created an initial prototype - which was replaced with a theoretical approach in Cycle 4 due to a discovery made in terms of how Morton values work in Cycle 3 (the prototype is discussed further in Section IV-C)

Regarding the algorithm artifact:

- An analysis was conducted on the single dimensional maneuver data in order to choose which combination of sensors provided better CSPs identification. Then, the data was analyzed in terms of similarities and differences in CSPs derived. This step related to answering RQ1.
- The second step consisted of answering RQ2, through analyzing what properties of the patterns were required for estimating the dimensions of a state space for a specific event.
- Step 3 consisted of developing a theoretical approach based on the findings of the previous two steps, which targeted answering RQ3. This step was initiated through a literature review, in which existing knowledge was analyzed in relation to data found in the previous steps. Analysis from the literature review acted as a base on which the theoretical approach was built.

activity 4: Demonstration - The fourth activity consisted of demonstrating "the use of the artifact to solve one or more instances of the problem" [18]. For the 2nd cycle where the artifact comprised of a prototypical algorithm, this included a demonstration of its output and time complexity. As we progressed to develop a theoretical approach in cycle 4, the activity entailed only proof-of-concept demonstrations.

activity 5: Evaluation - To evaluate the artifact, the 2nd cycle's prototype was evaluated in terms of time complexity. In the following cycles, the theoretical approach was evaluated in terms the following parameters (further discussed in Section V-D3): performance, modularity, modifiability, completeness.

A. Data Collection

A variety of data was collected to determine which parameters to base the CSP generation on due to their impacts on pattern occurrences on the CSPs. A clearer CSP pattern will enable for easier identification for what can be considered a lane change maneuver on a CSP, which in turn allows for more accurate permutations to be made. These parameters consisted of:

- Type of SFC used (Morton or Hilbert)
- Sampling rate
- Number of dimensions of the data
- Types of sensors used

The data collected was based on a real-world dataset consisting of hard braking and lane change maneuvers. These maneuvers were collected from a Volvo XC90 owned by Chalmers, which underwent a variety of maneuvers on a test ground as well as city roads. The data consisted of sensor readings such as acceleration in three dimensions.

From the aforementioned lane change datasets, we initially had access to a subset consisting of hard-braking and lane change maneuvers from a csv named 'braking_and_lanechange.csv' found in the SFC_scripts Github repository [20]. From this initial data, nine lane change maneuvers were extracted, which we refer to as "Dataset A" in our research. The following sensors were provided for Dataset A:

- GPS: Longitude and Latitude (20Hz)
- X, Y, and Z accelerations (20Hz)
- Speed (20Hz)

Data from Dataset A had a sampling rate of 20Hz, which we down-sampled to both, 5Hz and 10Hz - the reasoning for this being to analyze and view the impacts of the sampling rate on the patterns found on the CSP. For each of the different sampling rate varied Dataset A's, CSPs were then generated based on both, Hilbert and Morton indexing.

After the aforementioned data collection had taken place, we received a new subset of the real-world dataset consisting of 14 ”noise” maneuvers such as a braking maneuver, and 16 lane change maneuvers of varying harshness levels when making the lane change. We refer to this dataset as ”Dataset B” in our research. The following sensors and sampling rates were provided for the Dataset B maneuvers:

- GPS: Longitude and Latitude (20Hz)
- Steering wheel angle (50Hz)
- X, Y and Z acceleration (100Hz)
- Angular Velocities: Roll rate and Yaw rate. (100Hz)

Each of the sensor readings above came in a separate CSV file, and so maneuvers for each sensor were derived via finding the timestamp range of the lane change maneuver from the GPS data.

We decided to down-sample the Acceleration sensor CSV for ten of the Dataset B lane changes to generate data with 100Hz, 50Hz sampling rates. To verify the sensor readings' impact on CSPs, four lane changes of the aforementioned ten were down-sampled to 50Hz and based on: Y-acceleration in combination with either the steering wheel angle or the yaw rate. CSPs were not generated on the Roll rate sensor due to its measurement not being related to the nature of the maneuver. The sensor combinations that CSPs were generated on for the analysis were:

- Steering wheel angle and Y-acceleration
- Y-acceleration and steering wheel angle
- Yaw rate and Y-acceleration
- Y-acceleration and Yaw rate

The reasoning for down-sampling four lane changes from Dataset B was to balance time constraints during the data collection. The selected four lane changes were of varying levels of harshness as well as lane-change side. After the different sensor reading impacts had been analyzed, we decided to down-sample the aforementioned four lane changes further to both 5Hz and 10Hz. Results on the decision of Sampling rate can be found in V-A.

To finalize our answer to RQ2 and also to analyze the state space itself for RQ3, all lane change and noise data from Dataset B were extracted and SFC-encoded, based on the following parameters:

- Type of SFC used: Morton
- Sampling rate: 10Hz
- Number of dimensions of the data: 2
- Types of sensors used: Y-Acceleration and Steering Wheel Angle

Furthermore, roundabout data was also extracted from a dataset known as the 'SnowFox Roundabout Dataset', from which we also extracted and SFC-encoded roundabouts based on the aforementioned parameters. To provide clarity on the data, noise data from Dataset B does not refer to sensor noise, but rather to other maneuvers that are not lane changes.

In order to solidify our understanding of Morton encoding, as well as to ensure the validity of permutations, we needed to clarify the reason why Morton outliers occurred in some SFC-over-time plots. This analysis would determine whether there are value ranges where Morton values cannot happen, potentially affecting the dimensions of the areas of interest (AOIs) which are annotated in Figure 2. Moreover, it would explain clearly the way Morton encoding works, and give us hints on the adjustments necessary to generate valid permutations. Morton value data was collected in three ways:

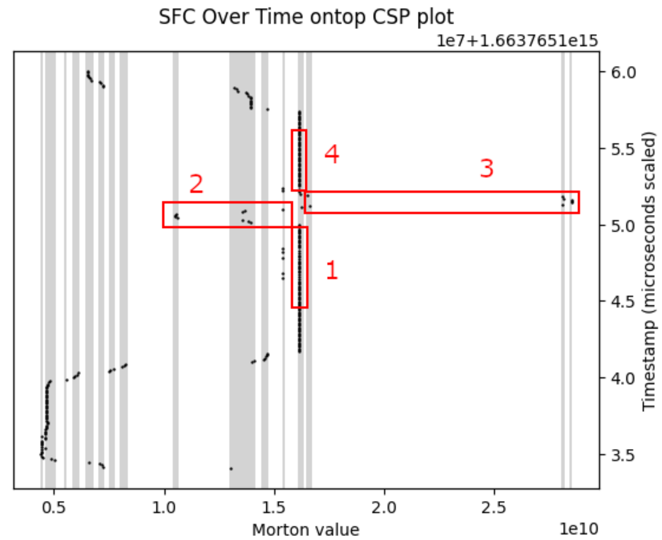

Fig. 2. SFC-over-time plot with the 4 different parts of a lane change maneuver annotated

1) Morton values manually generated: Morton values were generated using scripts, with varying Morton value range, and spread. Specifically, values were generated with linear and exponential growth, as well as following certain shapes, such as the horizontal bell shape that was recognized in our lane changes' areas of interest. This data can be observed in Figure 23.
2) Morton values taken from our lane changes data (Dataset B, LC15 and LC16): Morton values were taken from lane changes that contained outliers (since they were the focus of this analysis). The lane changes were chosen from Dataset B (lane change 15 and 16), as their SFC-over-time plot contained outliers. More specifically, the outliers on the right side were analyzed first (area number 3 in Figure 2), but the observations apply to both sides of the plot. The outliers were isolated in their respective SFC-over-time plots, and they were shifted using a variety of offsets. The impact that this would have on unpacked sensor data is further discussed in Section VI-D1.
3) Morton values taken from algorithm prototype generated lane change: Finally, Morton values were taken from one lane change generated using the algorithm prototype, to observe and further support the understanding of the Morton value sampling issue.

To improve the density and definition of the spike and contour plots, we collected sensor data from lane changes performed through a simulator (CARLA [21]). The simulated data was given to us by a group of peer researchers, who we thank and mention in Section VII. Firstly, we measured through a script the lane change with the median maximum amplitude in terms of Steering Wheel Angle. Secondly, we manipulated that lane change's Steering Wheel Angle sensor values to generate 5 files with increased steering amplitude (resembling harsher lane changes), and 5 files with decreased steering amplitude (resembling more gentle lane changes). The amplitude of the reference file was manipulated by applying a coefficient to the sensor values in 5 increasing steps of about 13% for increased amplitude, and 5 decreasing steps of about 10% for decreased amplitude. each other. Once the manipulated files were ready, they were given to our peers for performing the simulation, along with speed files which were required to run the simulation. The speed profiles used for the simulations were: 4m/s, 5m/s, 6m/s, 7m/s, 8m/s, 10m/s, 14m/s. We then received back the simulated lane change data files, which contained the relevant sensors for our research, along with location data which was useful for lane change validation. The simulated lane change data was then analyzed through a script, which checked the sideways movement of the vehicle through the location data of the simulations. With this approach, after defining the standard width of the lane (3.7m in Germany [22]), as well as the one of the vehicle (approximately 2 meters for a Volvo XC90), we were able to determine whether a simulated lane change could be considered valid or not. We defined a lane change to be valid when the car moved from one lane to another (i.e., two-lane lane changes or more were not considered).

After the analysis was completed, we were able to add five new lane changes to our pool of data. These lane changes were used to add more data to the density plots analysis, as discussed in Section III-B

B. Data Analysis

A variety of data analysis was conducted in relation to the lane change maneuvers from both, Dataset A and B mentioned in Section III-A. To analyze the lane change maneuver data in its multidimensional and CSP form, the following software, libraries and packages were used:

- Programming language: Python 3.9
- Data Processing libraries: pandas 1.5.3, numpy 1.24.2
- Data visualizations: matplotlib 3.7.1
- Computing Platform: Jupyter notebook 6.5.3
- Morton indexing library: morton-py 1.3
- Hilbert indexing library: hilbertcurve 2.0.5

The code used for analysis and data processing was based on a fork of the SFC_scripts repository by Lukas Birkermeyer [20].

In order to encode the sensor data into a Morton or Hilbert index, the data requires an offset and a multiplying factor. Both, the Hilbert and Morton libraries, require a non floating and positive integer. Hence, an offset of 10 was added to the data to ensure that values are positive and the values were multiplied by a factor of 10000 to remove the decimal and ensure that sufficient accuracy is maintained from the sensor reading. The adjusted sensor values were also casted to integers to ensure that they are non-floating. The aforementioned factor multiply and offset were already set in the SFC_scripts repository that we based our analysis on [20]. After applying the adjustments to the sensor data, the shape of the sensor readings against timestamp plot was still maintained such that the impacts of the maneuver were still accurately represented.

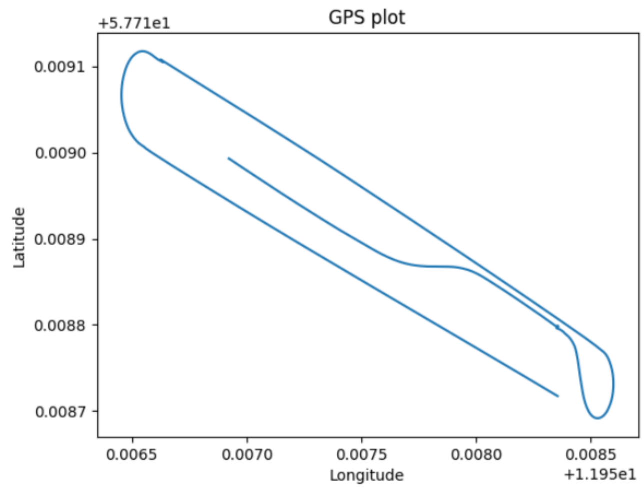

Fig. 3. Plot of latitude and longitude data showing the maneuver path of the vehicle

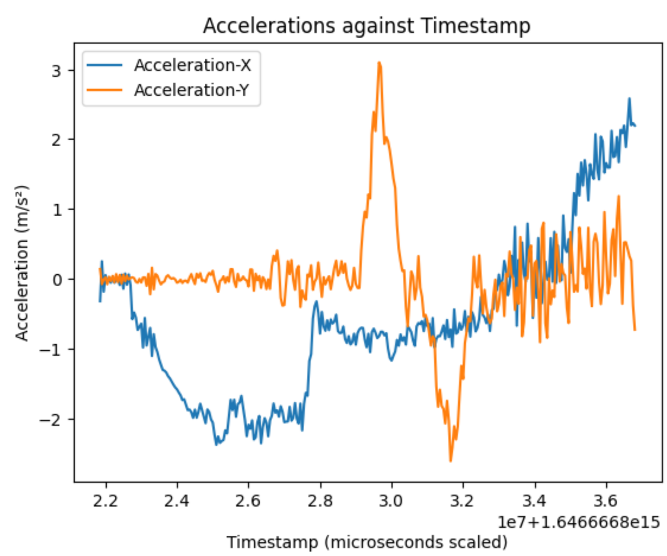

Fig. 4. This is an example image of X and Y acceleration against their corresponding timestamp

The first phase of the data analysis included plotting the longitude and latitude position of the vehicle - which is equivalent to GPS - to identify lane change timestamps from the lane change datasets (example of the plot seen in Figure 3).

Based on the timestamps extracted from the GPS plots, we could determine what parts of other sensor readings corresponded to lane changes when plotted against timestamps. This in turn helped us understand how sensors, such as acceleration sensors, were affected by lane changes. An example of the X-acceleration and Y-acceleration against timestamp plot can be seen in Figure 4. The analysis of sensor data, such as acceleration sensors, against timestamp plots provided insights into the relationship between between sensor readings and lane change patterns on the CSPs generated from corresponding sensors.

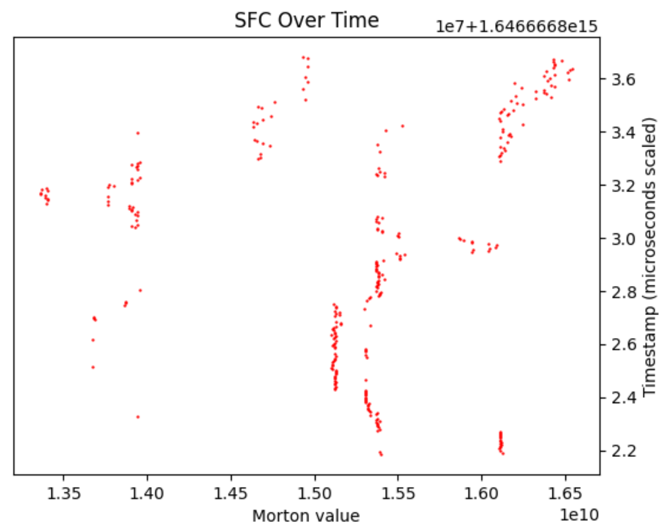

Fig. 5. Example of an SFC-over-time plot

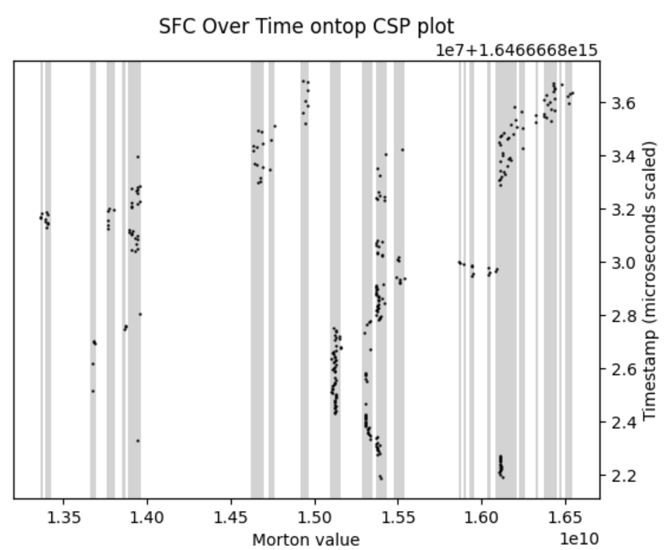

Fig. 6. Example of an SFC-over-time plot on top of a CSP

Throughout the research, CSPs were analyzed in more depth via SFC-over-time plots. An example of an SFC-over-time plot can be seen in Figure 5. An SFC-over-time plot shows the

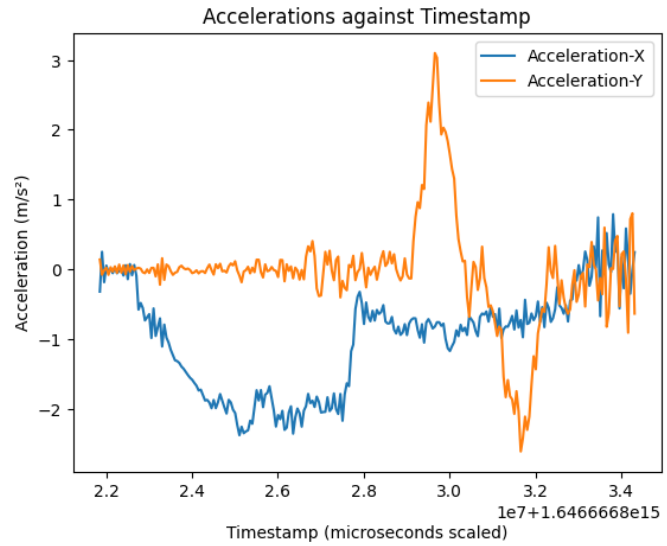

Fig. 7. Example of an X and Y acceleration against timestamp plot with maneuver noise removed

occurrence of Morton values with respect to time. Relating Figure 5 to CSPs, each point in the SFC-over-time plot represents an occurrence of a stripe in a CSP plot. To visualize the aforementioned relation, Figure 6 shows an example of an SFC-over-time plot on top of a CSP. The SFC-over-time plots enabled us to analyze what parts/stripes of the CSP correspond to different parts of the maneuver. For example, certain stripes are specifically related to the turning aspect of the lane change, whilst other stripes are related to the initial acceleration before going into a lane change. The link between the CSP and SFC-over-time plot is as follows: The SFC-over-time plot shows clusters of Morton values that correspond to a specific timestamp. When a point appears in the plot, a corresponding stripe can be seen in the CSP equivalent, and the more points there are per cluster, the thicker the stripe in a CSP will be. Furthermore, the SFC-over-time plot also provides insights into the order of generation of stripes in a CSP, enabling the identification of patterns through the distance between and ordering of clusters.

To analyze the impacts of the sensor choice and ordering, we had decided to analyze SFC-over-time plots to visualize how clear patterns were when varying the type of sensor and ordering of them.

After the aforementioned paragraph, all lane changes, noise and roundabout data were extracted as discussed in III-A. SFC-over-time on top of CSP plots, CSPs, and corresponding sensor against timestamp plots of lane changes were compared to those of noise and roundabout data. The analysis was focused on the position and range of stripe occurrence on CSP plots of different maneuvers compared to lane changes, and if the temporal occurrence of these stripes would be critical to estimate the dimensions of a state space for a maneuver, with the ultimate goal of the analysis being to finalize RQ2. Results of this analysis are presented in V-C.

With regards to the Morton value sampling issue, data analysis was conducted by unpacking and observing the collected Morton value data. Unpacking Morton values would give us the possibility to look at how the sensors values (i.e., Y-Acceleration, Steering Wheel Angle) were impacted by the Morton value sampling or manipulation that happened in the Data Collection stage. Once sensor values were unpacked, and sensors were plotted, the plots were compared to the real lane changes from our datasets, which we could use as a visual reference. If the newly generated plots did not visually follow the usual sensor behavior seen in lane changes from Dataset A and B (i.e., correlation of Steering Wheel Angle and Y-Acceleration), they were considered invalid.

The lane changes that were collected from Dataset B and simulator were also analyzed in terms of the density of Morton points in the SFC-over-time representation of the data via two plots: spike plots and contour on-top heatmap plots. An example of a spike and contour on-top of heatmap plot can be seen in 8. Prior to generating the density plots, the lane changes first had to be normalized based on the timestamp to ensure that different aspects of the maneuver [23], such as stabilizing into a lane after a lane change, are overlapping - this enables the density plots to highlight the more dense areas of points that are related to a similar part of the maneuver. This normalization was provided by peer researcher Renyuan Huang, who we mention in Section VII. An example of an overlapped SFC-over-time plot can be seen in 9.

The densities of the corresponding scatter points from the SFC-over-time plot were calculated via the Numpy library's "histogram2d" function, which provides a two dimensional array with a count of the amount of SFC-over-time points that fall within their respective bins. Regarding the binning of the density plots, the square root of the number of points on the SFC-over-time plots was taken, and then heuristically adjusted based on visual clarity of densities. The bin values are as follows:

- Spike plots: 31
- Contour plots on-top of heatmap: 50

Although both of the plots seen in 8 show the density, the spike plots provide insights into the overall pattern of the data, as well as distribution based on the density of the bins. The contours on-top of heat map plot provides a different perspective of the spike plots, allowing for Morton value ranges more related to lane change maneuvers to be more easier visualized, as the contours provide a 2D way to view the spikes, and the heatmap provides a visualization on how many Morton values from our SFC-over-time plot fall into a bin via the color of the bin - each bin is assigned a colour in the heatmap based on its density.

The reason for the density plots is to view if more concise areas of interest based on the density of Morton value occurrence on an SFC-over-time plot can be formed.

IV. Artifact

In this section we discuss the suggested theoretical approach for generating valid lane changes. Two main conceptual components are needed: Algorithm for Morton value sampling, and a plausibility checker (the need for the plausibility checker is discussed in the upcoming Section IV-B). Figure 10 contains a visualization of the suggested sequence of steps that would result in the most optimal use of the sampling algorithm and plausibility checker. This approach can be seen as a modular pipeline, where we want to emphasize the ease in the modifiability of its composition (i.e., addition or modification of constraints). Each of these conceptual components is explained in detail in their own subsections (Section IV-A and Section IV-B). Finally, the end of this section describes an initial Morton value sampling algorithm prototype that was developed in cycle 2 [10] (Section IV-C).

A. Morton value sampling Algorithm

The algorithm for Morton value sampling can be built with similar concepts as the prototype explained in Section IV-C. Specifically, the theoretical approach of the algorithm is conceptualized with the use of 4 areas of interest (representing the 4 different maneuver parts of a lane change) divided in columns and rows, representing Morton value range and time dimension respectively. An important improvement from the prototype is that more concise AOI Morton value ranges can be defined within the different maneuver parts shown in Figure 2 via the use of spike plots (explained in Section VI-D2). The aforementioned improvement would increase the likelihood of sampling correct Morton values (contributing to the issue discussed in Section VI-D1, as well as greatly reduce the amount of Morton values that can be sampled. Once the AOIs are determined, Morton values are sampled for each AOI separately, with the following conditions:

- AOIs referring to the vehicle in a straight driving configuration (i.e., before and after the lane switch, areas 1 and 4 in Figure 2) are characterized by a sequence of constant Morton values (which should resemble constant steering angle). It is important to note that this is a limitation in the amount of lane changes generated, as the steering wheel angle is not exactly constant during these sections of the maneuver. However, to not degrade performance of the algorithm, we considered the aforementioned choice as the best in terms of trade-off between amount of permutations and time efficiency of the sampling step.
- AOIs referring to the vehicle during the lane switch (i.e., initialization of lane switch and stabilization in the new lane, areas 2 and 3 respectively in Figure 2) are characterized by what can be seen as C-shaped Morton values in the SFC-over-time plots, as explained in Section IV-C. This choice was made to provide an initial and basic approach to systematic Morton value sampling, as we generally notice that the shape of occurrence of Morton values resembles the one of its related sensor values.
- The number of rows per AOI is equal to the time duration of the AOI multiplied by the sampling rate of the sensors. Consequently, the number of Morton values sampled per AOI is equal to its number of rows. Again, this is an

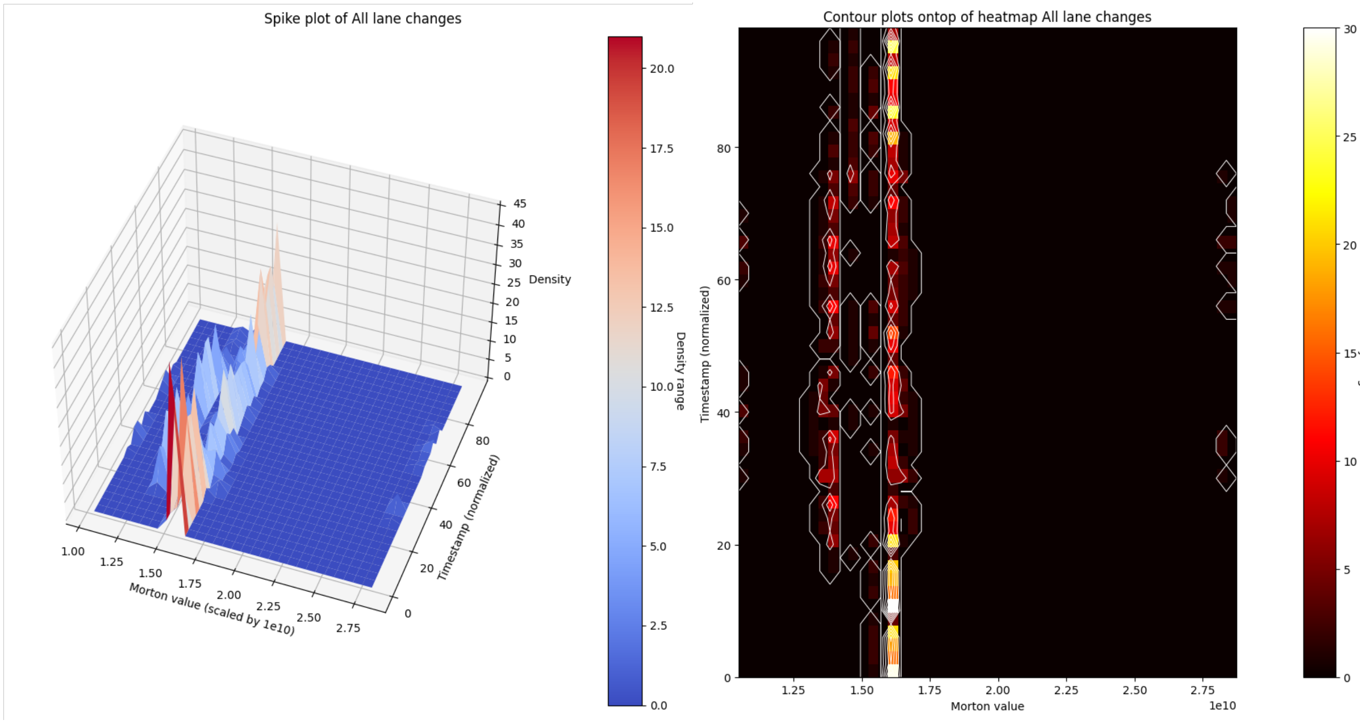

Fig. 8. Example of the two different density plot types

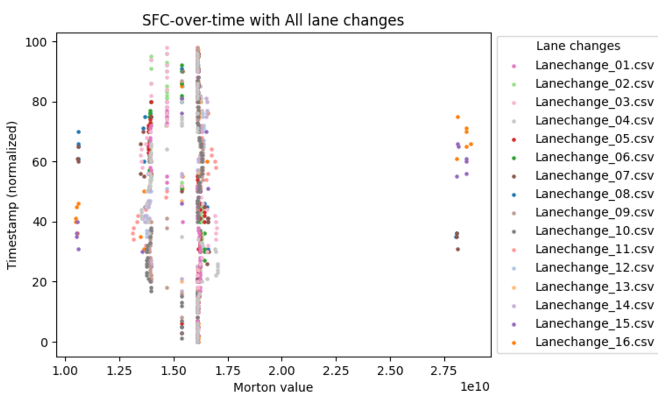

Fig. 9. SFC-over-time plot of all left and right lane changes overlapped after being normalized via the timestamp

initial approach that simplifies the determination of the amount of rows per AOI that future work can modify as needed, for example by increasing the number of rows per AOI while keeping the number of Morton values sampled constant.

After the sampling step is completed for all AOIs, the algorithm shall connect these AOIs generating unique sequences of Morton values that cover the entire duration of the manoeuvre. This can be done by performing the multiplication of all combinations from all AOIs (as explained in Section IV-C).

B. Plausibility Checker

Following the findings made on Morton encoding (discussed in Section VI-D) in terms of plausible permutations within areas of interest, we determined that there is a need for a conceptual component that can guarantee whether the sampled Morton values can generate valid lane changes. This component will be referred to as "plausibility checker". The plausibility checker introduces rules that analyze unpacked sensor data for violations in the defined constraints that a lane change should follow. The approach proposed focuses on steering wheel angle sensor checks only as an initial step for future research to build upon, but it is important to note that we deem it applicable to acceleration as well. Nonetheless, the concept is limited in that it still would not consider the correlation between the sensors, but only their individual evolution. We therefore present the following constraints which should be put in place for Steering Wheel Angle sensor data, regarding unpacked AOIs where Morton values are generated following a curved pattern (areas 2 and 3 in Figure 2):

- There should be exactly one peak per AOI's unpacked sensor data.
- Sensor values shall evolve smoothly and with no sudden increase or decrease.
- Sensor values must be all positive or all negative, according to which AOI they are being unpacked from.
- The two curved AOI sections which make up the curved part of steering wheel angle sensor values shall be symmetrical.

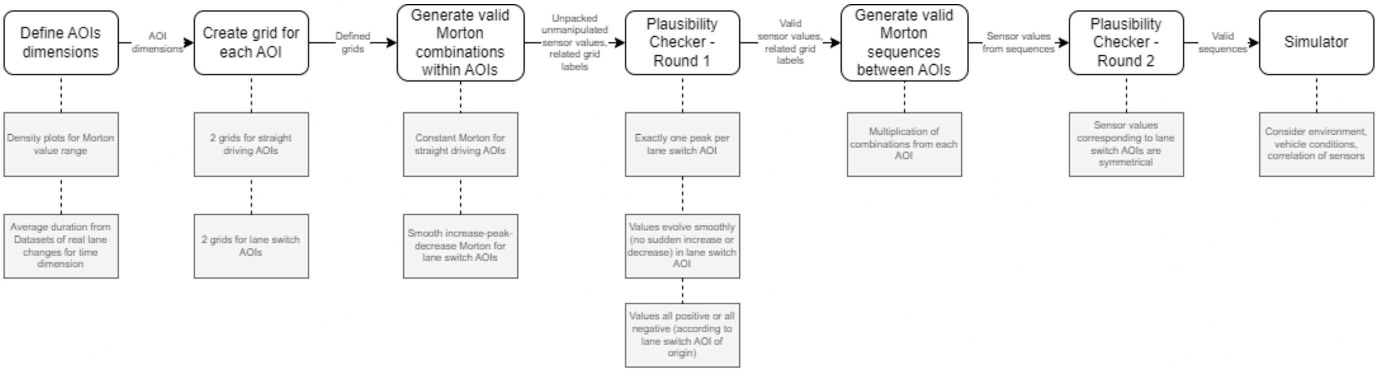

Fig. 10. Visualization of suggested sequence of steps for usage of sampling algorithm and plausibility checker

The first two constraints can be ensured by traversing pairs of adjacent sensor values, and checking for the coefficient of the straight segment that connects them. A change in the sign of the coefficient would mean that a peak has been reached (first constraint), and a defined threshold would ensure that the inclination of such segments evolves smoothly (second constraint). It is important to note that the smoothness of the inclination shall be checked separately before and after the peak value, to avoid invalidation of correct lane changes (since after a peak value it is expected to see a sudden change in inclination). The third constraint can be ensured by checking the sign of sensor values. Here it is important to de-manipulate the unpacked sensor values by reverting the application of “OFFSET” and “FACTOR_MULTIPLY”, to obtain the raw sensor values which should be either positive or negative. Finally, in order to ensure the fourth constraint, AOIs shall be checked against each other for symmetry. One example of how this can be done is by having sensor values stored in separate arrays and comparing the absolute value of each element at the same index. If the absolute value is the same (or within a certain threshold), the two AOIs can be considered symmetrical. If the unpacked sensor data from the sampled Morton values passes the aforementioned checks, the lane change can be considered valid. However, it is important to note that in order to validate the lane change with total certainty, domain properties need to be specified as well. These domain properties are: vehicle speed, vehicle type, lane width, environment of the maneuver. In order to ensure that the combination of sensor values is valid considering the aforementioned domain properties, we deem necessary the use of software that can simulate lane changes by providing sensor values as input, and consequentially a script that can check the output of the simulations, similar to how it was discussed at the end of Section III-B.

C. Algorithm prototype

This section discusses the implementation details of an initial prototype for the Morton value sampling algorithm. The initial algorithm was developed using Python 3.9 due to the language’s simplicity, but we acknowledge that using other languages could provide more efficient permutations. The first implementation is a brute-force algorithm, and was developed in this way to act as a foundation that can be built and improved upon. We would like to note that this was an algorithm we had created in Cycle 2, prior to making a discovery on how Morton values should be sampled as discussed in VI-D1. Hence, the boundary boxes from which Morton values are sampled in this prototype are too broad and result in many implausible lane change permutations. But, the utilization of a 2D array grid approach could still be utilized by future research for more concise areas of interest defined by density plots, as discussed at the start of SectionIV-A.

Areas of interest were first determined to lay out the boundaries in which Morton values appear in an SFC-over-time plot - these were based on the different maneuver parts related to a lane change, and can be seen annotated in 2. From this, a grid was developed using a 2-Dimensional Array data structure to imitate an SFC-over-time plot. The array contains 0’s and 1’s, where 0 represents an empty space, and 1 represents a ”(Morton, timestamp)” tuple. Within this 2D array, all possible combinations of 1’s positions in the grid are made.

As all permutations of possible 1’s are made in the grid, we coded rules that would define what a lane change should look like in terms of SFC-over-time clusters/point in order to mitigate the occurrence of permutations that are not related to a lane change maneuver pattern, which is denoted in Section V-D. Rules were made for each of the types of areas of interests denoted in Figure 2, and can be summarized as the following in relation to Fig 2:

- Area of interest 1 & 4: Generate 1’s by keeping the column value (Morton value) constant over the rows (Timestamp) for each combination.
- Area of interest 2 & 3: Keep permutations of the grid that consist of at least one entry point into the curved shape, one peak point, and one exit point. To clarify, a point in this case would be represented by a 1 in the 2D array grid.

After the valid permutations are determined for each area of interests, the corresponding Morton and timestamp values are extracted from the index of the 1’s in the 2D array in order to produce a set of points for each area of interest.

Finally, permutations are then generated via permuting the combinations of valid permutations from each area of interest.

To simplify the initial prototype, the following implementation limitations had been decided:

- Keep the grids and rows related to the sampling rate, such that the possible positions of 1's are reduced. If this was not in place, more columns and rows could be added such that there is a more accurate scale for Morton values, which in turn increases the possible permutations as we would want to keep the number of points constant within an area of interest to stay consistent with the sampling rate of the data.
- As this was an initial prototype, we had decided to keep the resolution of our grid axis at 2 decimal places in regards to both, the timestamp (scaled 1e6) and Morton values (scaled 1e10).

Regarding the permutations of the 1's for each individual Grid, a back tracking permutation algorithm was inspired by an article in the "Medium" web page [24]. Regarding the permutations of the different areas of interests, a nested for-loop approach was taken to permutate all possible combination of subsequent areas of interests.

V. Results

A. Parameters to base CSPs on

During the start of our research, we assumed that Morton or Hilbert may provide better CSP generation based on different values of: sampling rate, dimensions of the data, and types of sensors. For example, Morton may work better with a higher sampling rate than Hilbert. Going on this assumption, we decided to plot the same variations of the three aforementioned parameters for both Morton and Hilbert based CSPs.

As discussed in the Data Collection Section III-A, we had two subsets, called Dataset A and Dataset B respectively, derived from a maneuver dataset. Results that were found from both datasets will be discussed.

Regarding the sampling rate choice, as described in Section III-A, we initially plotted ten lane changes with 100Hz, 50Hz, and 20z sampling rates specifically for X and Y accelerations, and after, plotted four of these ten with 10Hz and 5Hz sampling rates. From these different sampling rates, we did not find any major differences in terms of patterns and stripes generated apart from clusters seeming more evident in SFC-over-time plots based on higher sampling rates. However, we decided to opt for 10Hz as it provided the best compromise between detail of the "S" shape pattern discussed in Section V-B and amount of data, allowing for easy visual identification of a lane change as well as a potential performance benefit for the algorithm artifact due to a reduced amount of data to permute.

Regarding the sensor choice, we had deemed the X-acceleration to provide most insights into the lane change maneuvers and hence decided to replace the Z-acceleration with the yaw rate and steering wheel angle sensors that came with Dataset B to see if they combined better with

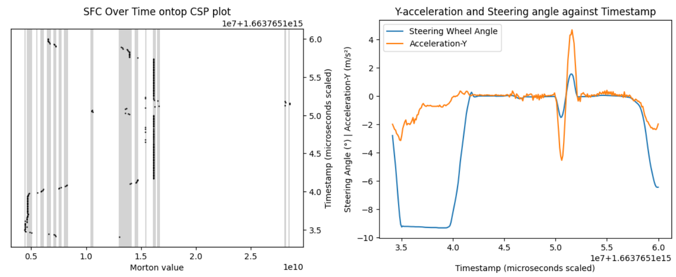

Fig. 11. SFC-over-time on top of CSP based on Morton encoding side by side with Y-acceleration and steering angle against timestamp plot

the X-acceleration in terms of pattern clarity in the CSPs. We determined that the combination of steering angle and Y-acceleration provided the clearest pattern for identifying lane changes, and had a strong correlation to the maneuver. As an example, figure 11 shows the acceleration and steering wheel angle against time stamp, and its corresponding SFC-overt-time plot. The S shape maneuver can be clearly seen in the time frame of the lane switch (timestamp values between: $5\mu s$ (scaled) - $5.3\mu s$ (scaled)). Furthermore, due to the close representation of the SFC-over-time plot to the maneuver, it is also a lot clearer when noise occurs, as can be seen in time stamps: $3\mu s$ (scaled) - $4.2\mu s$ (scaled). Furthermore, we also identify that a lane change around Morton value 1.6 (1e10) just before the lane switch occurs - this observation may contribute towards both RQ1 in terms of finding similarities between patterns and RQ2 in terms of what is necessary/sufficient in determining the dimensions of the lane change maneuver.

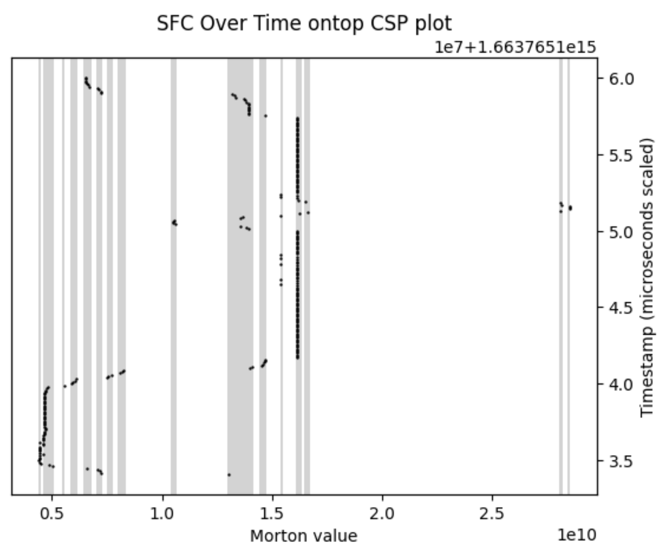

Fig. 12. SFC-over-time on top of CSP based on Morton encoding. The parameters are Morton encoded in the order: Y-acceleration and Steering angle.

Finally, we also found that the ordering in which sensor data is fed into the Morton encoding impacts the order at which stripes appear in CSPs. For example, Figure 12 and 13 show the steering wheel angle and Y-acceleration combination for a

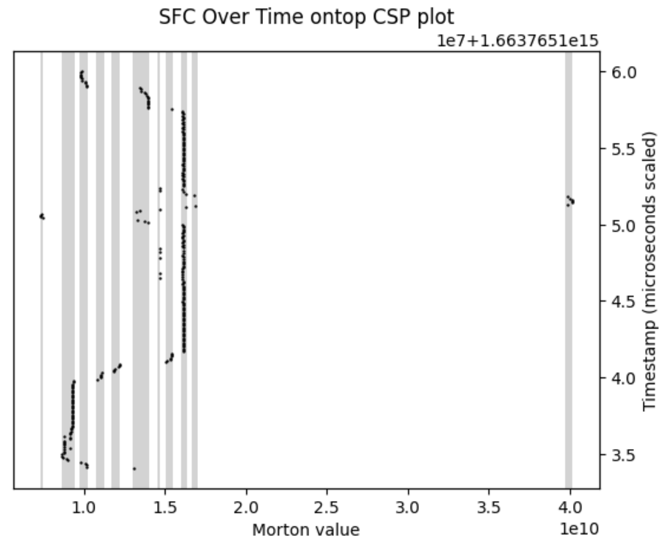

Fig. 13. SFC-over-time on top of CSP based on Morton encoding. The parameters are Morton encoded in the order: Steering angle and Y-acceleration.

lane change with varying which parameter is first fed into the Morton encoding. Although the same general pattern arises in both CSPs in terms of SFC-over-time plot clusters, it can be seen that the ordering of the annotated stripes changes between the graphs.

B. Research Question 1

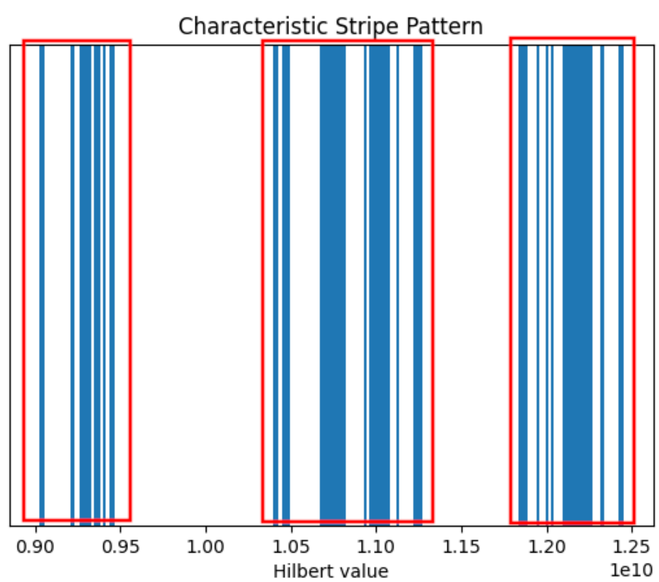

Fig. 14. Annotated CSP based on Hilbert encoding to emphasize three stripe groups

As the SFC-over-time plots act as refinements of CSPs by providing more insights into the temporal domain of the CSP, we had used them to answer RQ1. Based on the SFC-over-time plots described in Section III-B, we had decided that

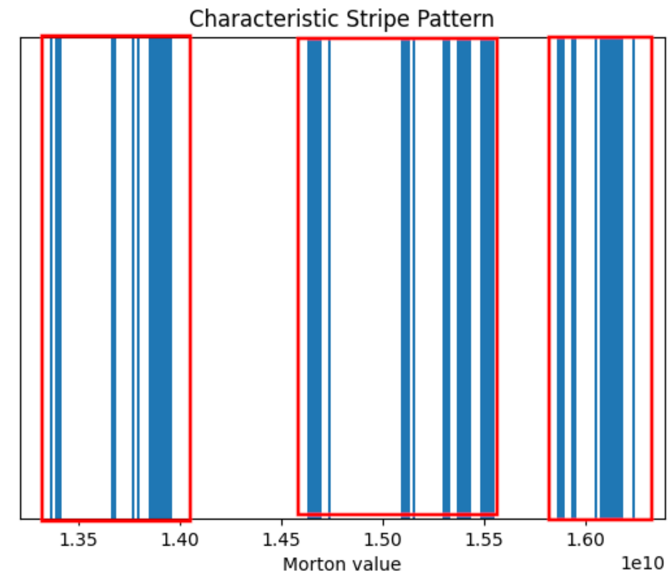

Fig. 15. Annotated CSP based on Morton encoding to emphasize three stripe groups

using Morton indexing to generate CSPs provided more clear and consistent patterns.

Compared to Hilbert, Morton had a consistent pattern for all nine lane changes. In particular, an ”S” shaped pattern can be seen within the time frame of the lane change, an example of this pattern can be seen in the highlighted area in Figure 19. The ”S” shape pattern closely follows the turning maneuver in terms of turning into a different lane, and then turning the opposite direction to stay inside of the newly switched lane.

Hilbert based SFC-over-time plot seemed more inconsistent to us compared to Morton. For example, figure 16 and Figure 17 show two different lane changes on a Hilbert-generated CSP, that both have a similarity in terms of the clusters within the highlighted area that fall inside of the time frame of the lane change. Figure 18 on the other hand shows a third lane change based on Hilbert indexing, but the clusters within the highlighted area that represent the lane change look different and do not follow a similar pattern.

Furthermore, we found that at Y-acceleration values of around 4m/s^2 and greater, cluster points from the ”S” shaped pattern that represent lane change turns seem to jump outwards, forming stripes at the start and end of the CSP. This can be seen in Figure 20, where cluster points that seem to be apart of the peaks of the curved parts of the ”S” shape within the annotated area jump outwards at x=4 and x=0.5

C. Research Question 2

Following the Dataset B analysis for answering RQ2 discussed in Section III-B, we determined that the temporal domain of CSPs is necessary to estimate the dimensions of the state space of an event. The temporal domain of the state space plays a critical role in differentiating maneuvers that may have stripe occurrences in same areas that could be related to a lane

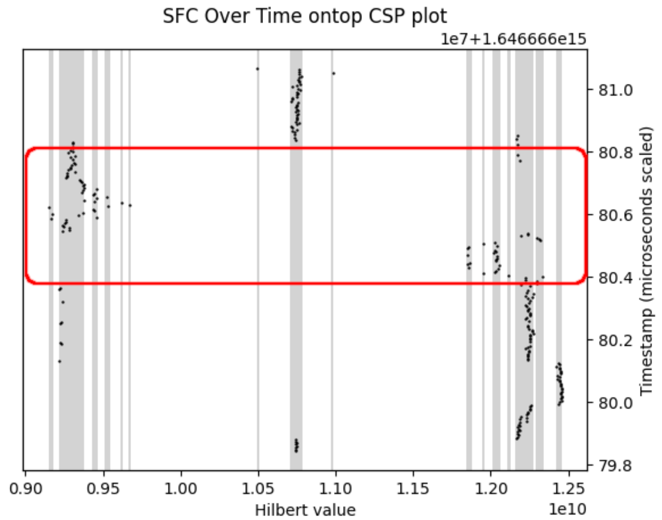

Fig. 16. Lane change 1 (Dataset A) with SFC-over-time on top of CSP based on Hilbert encoding

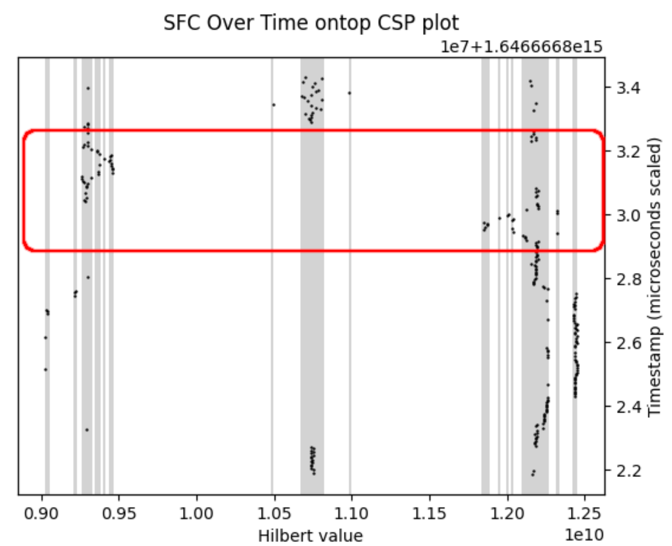

Fig. 17. Lane change 2 (Dataset A) with SFC-over-time on top of CSP based on Hilbert encoding

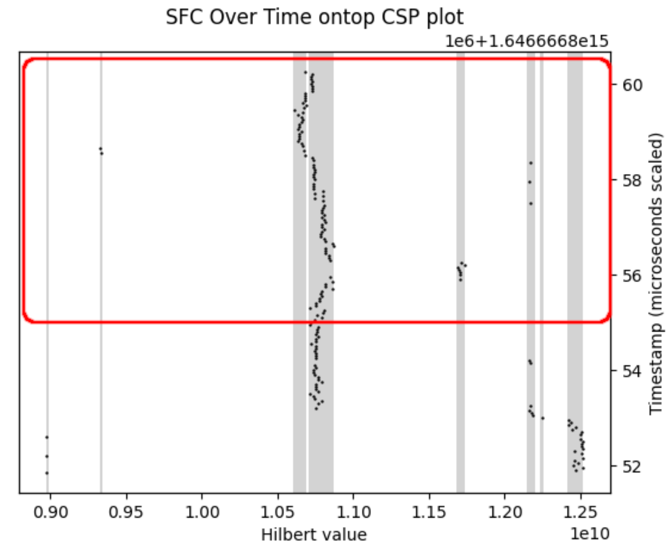

Fig. 18. Lane change 3 (Dataset A) with SFC-over-time on top of CSP based on Hilbert encoding

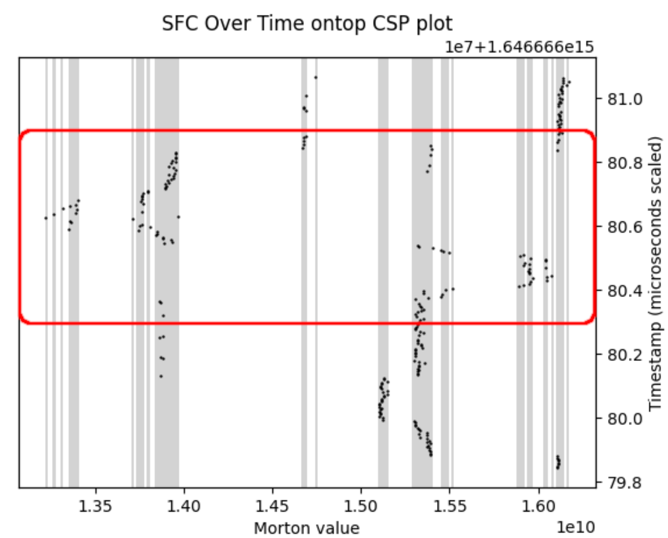

Fig. 19. SFC-over-time on top of CSP based on Morton encoding, with the lane change segment annotated. The plots are based on Lane change 1 (Dataset A)

change, and also allow for a confident detection/classification of the maneuver type inside of the state space.

Figure 21 shows CSPs for a lane change, roundabout, and noise maneuver in which a clustering of stripes can be seen around X = 1.5. Although a stripe is not seen in the lane change at X=4.0 when compared to the noise, this may be attributed to the lane change not having noise, in other words, if a lane change does have noise and a stripe occurred at roughly X=4.0, it may be difficult to tell apart the maneuvers with a high level of confidence. Viewing the noise maneuver and roundabout CSP, the stripes occur round X=1.5 and X=4.0 for both plots, further showing that it is not unlikely for different maneuvers to have stripes occurring in similar areas. All in all, we deem this could lead to a rise in false positives and false negatives in terms of determining the maneuver type based on stripe occurrence and limits of stripe appearance. To put the aforementioned results in a different perspective, if a CSP corresponds to the limits and Morton value occurrences of a lane change CSP, we cannot be certain that it is a lane change.

From the aforementioned results, we determine that a temporal domain of the CSP is necessary to determine the state space of a maneuver, and when combined with Morton value occurrence, is sufficient to estimate a state space. A more detailed discussion on this is made in Section VI.

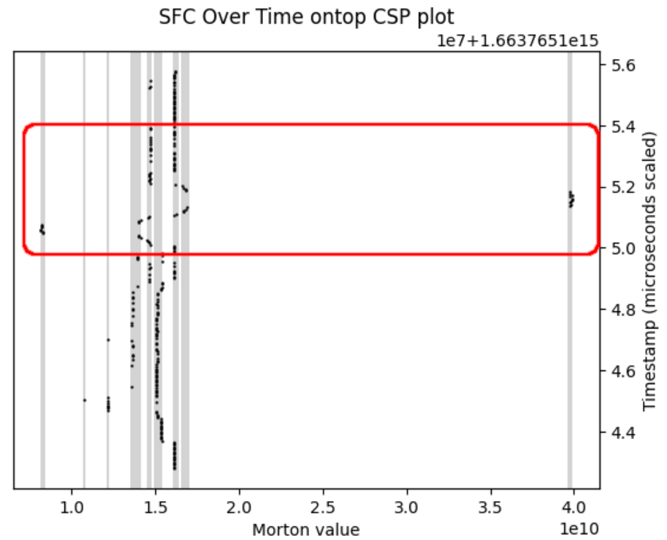

Fig. 20. SFC-over-time on top of CSP based on Morton encoding, with the lane change segment annotated. The plots are based on Lane change 8 (Dataset A), where the lane change segment includes two outliers at x: 4 and x: 0.5

D. Research Question 3

1) Morton value issue: Following the Morton value analysis aforementioned in Section III-B, we gathered the following information:

- How Morton encoding works: Morton encoding uses bit interleaving to "pack" multi-dimensional data into a single value (an example can be seen in Figure 22). Firstly, the data is converted to its binary equivalent. Secondly, the bits from each dimension are interleaved to produce the encoded values. This means that the resulting encoded Morton values will have the same amount of bits as the sum of bits of the multi-dimensional data. In our case, the sensor data to encode was Y-Acceleration and steering wheel angle. These values, with "OFFSET" and "FACTOR_MULTIPLY" applied, take up usually 17 bits of data each. This means that, most of the time, the resulting encoded Morton values will be the decimal representation of 34-bits binary numbers (i.e., 17+17). However, seldom, Y-Acceleration sensor values take up 16 bits (Y-Acceleration $<$ -3.4465 m/s^2) or 18 bits (Y-Acceleration $>$ 3.1071 m/s^2). Consequently, Morton values are affected (i.e., becoming either 33 or 35 bit numbers), and outliers are observed in the SFC-over-time plots (as shown in area number 3 of Figure 2).
- Morton values manually generated: Morton values cannot be generated linearly or exponentially within one area, and be expected to contain unpacked sensor data that follows such behaviors in a predictable way. An example result can be seen in Figure 23. The binary numerical system needs to be taken in consideration to find the optimal way of sampling Morton values such that encoded sensor data is affected in a predictable manner.
- Morton values taken from real lane changes: as Morton values were moved, the unpacked sensor data became increasingly distorted, in a way that was not expected. Furthermore, it is important to note that these distortions were observed even though the values were only manipulated to stay within the defined outlier ranges (i.e., referring to the range of Morton values from 28087995466 to 28735866149 within area number 3 in Figure 2), and that the offset values utilized for this manipulation had to be high enough to show an impact on the sensor values (since Morton value is in the scale of 1e10). The result of such a manipulation can be seen in Figure 24
- Morton values taken from algorithm prototype generated lane change: as expected following the previous results, the generated one from the algorithm prototype was not considered a valid lane change, due to the sensor data being too distorted to resemble the ones of the real lane change data, as can be seen in Figure 25

2) Density Analysis of SFC-over-time Plot Clusters: Regarding the density analysis on our SFC-over-time plots, figure 26 and 27 show spike plots and contour on-top heatmap plots respectively. The spike plot containing all lane changes in figure 26 shows annotated areas of Morton ranges in which Morton values related to a lane change occur. From this, we conclude that density plots provide "arbitrarily formed areas" in which Morton values related to lane change maneuvers are more likely to appear in.

Furthermore, when viewing the contour on-top heatmap plot equivalent, we can see a different perspective of the spike plots with insights on the bins and their impact on the contours. For example, in the contour on-top heatmap plot containing all lane changes in figure 27, at annotation 1's Morton range we can see a lower density, whilst at annotation 2, there is a high density concentration. The aforementioned result highlights the strength of the contour on-top of heatmap plots in being able to narrow down areas in the state-space of an SFC-over-time plot in which Morton values more related to lane changes are more likely to appear. Although, this result may not directly imply that the Morton range at annotation 1 in figure 27 should not be used to sample Morton values from for a lane change, as this area is still related to a lane change, and may not be as dense due to a lack of data. This shows that the heatmap is necessary to determine the areas of interest most concisely, as they will still highlight specific bins within less contour-dense areas that are a part of the maneuver.

Lastly regarding the density plots, simulator data was used to increase the density of the plots. There were increases in density at particular areas, some of which are, annotated on the left and middle contour on-top heatmap plot in figure 28. After simulated lane changes were added to the middle plot, there is a decrease in density in annotation 1, and a general increase in areas 2 and 3. Although, these density increases did not significantly increase the density space, which may be attributed to only 5 lane changes being added, as well as them being based on an existing lane change as mentioned in Section III-A.

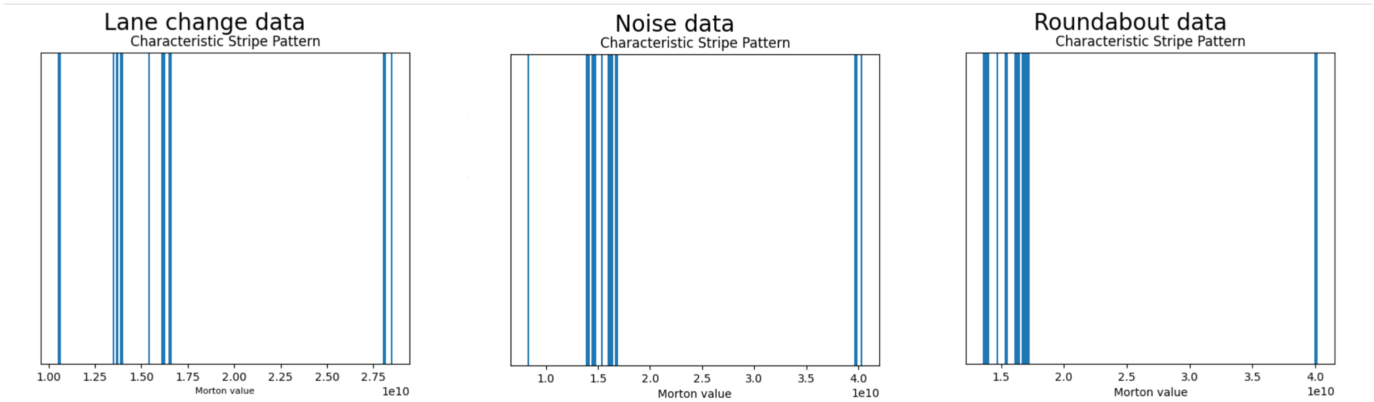

Fig. 21. CSP plot examples of of a lane change, roundabout, and noise maneuver.

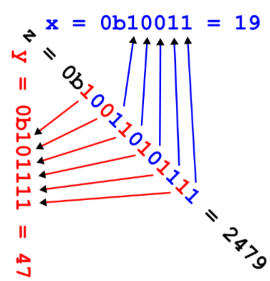

Fig. 22. Example of bit interleaving performed over values ”x” and ”y” to obtain ”z”

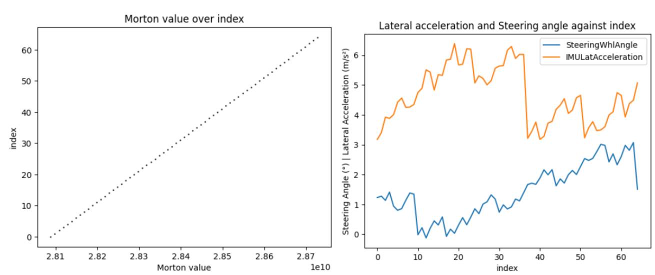

Fig. 23. Morton values generated linearly (left) and relative unpacked sensor data (right)

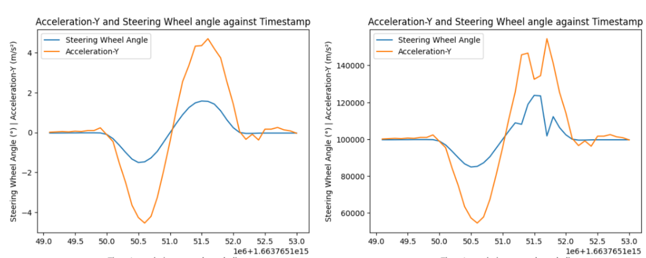

Fig. 24. Dataset B lane change 15 original (left) and manipulated with an addition of 50000000 to the Morton values of the outliers (right)

3) Artifact: With regards to results obtained from the theoretical approach, we performed an evaluation of the artifact on the following criteria: performance (measured in terms of time complexity), modularity, modifiability, completeness. The following paragraphs describe the measurements for each of these aspects in detail, while the results are discussed in relation to the research goal in the Analysis Section VI. The end of this subsection contains measurements regarding an implemented prototype for the algorithm explained in Section IV-C.

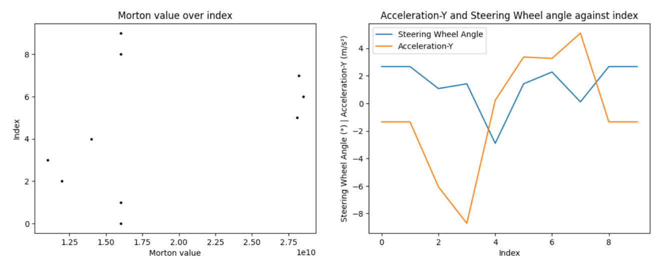

Fig. 25. Morton values generated using algorithm prototype (left) and relative unpacked sensor data (right)

- Performance: time complexity of the theoretical approach was measured by concatenating the initial results obtained from the prototype algorithm for Morton value sampling with an estimate of what the suggested implementation of the plausibility checker would cost in terms of time complexity over the input. A detailed description of the performance results can be seen in the following Table I, where:
  – ’p’ represents the number of columns per AOI. In our suggested approach, it is defined by the range of Morton values covered.
  – ’q’ represents the number of rows per AOI. In our suggested approach it corresponds to the number of sensor values per AOI. Since we are measuring worst case complexity, ’q’ is generalized and used to refer to the number of sensor values per whole sequence (i.e., when the AOIs are connected).
  – ’y’ represents the number of AOIs. In our suggested

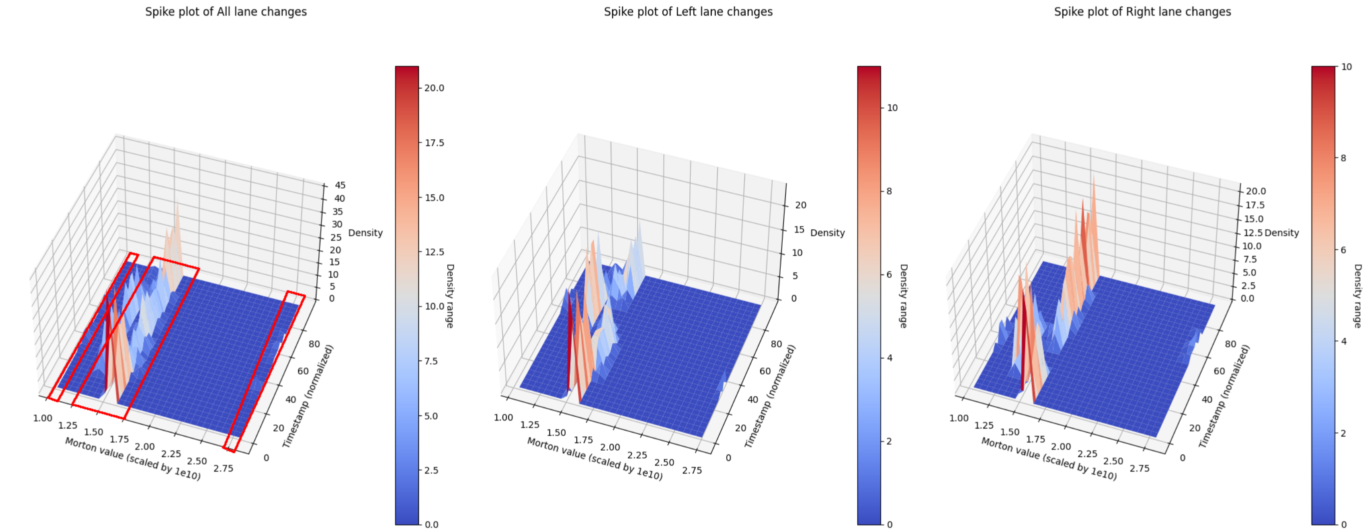

Fig. 26. Spike plots containing all lane changes (left), left lane changes (middle), right lane changes (right)

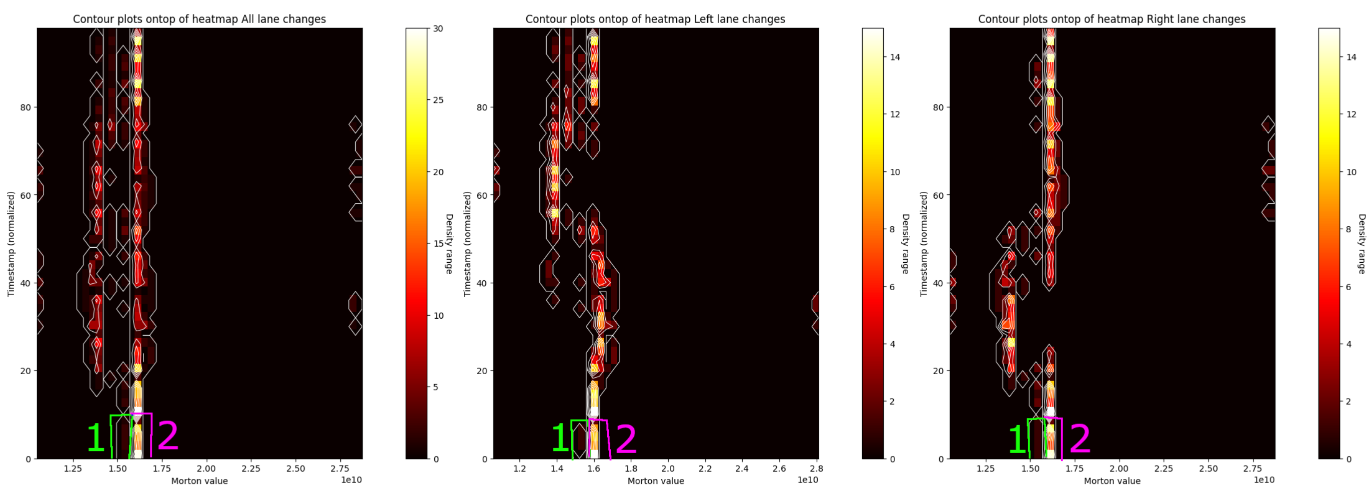

Fig. 27. Contour on-top heatmap plots containing all lane changes (left), left lane changes (middle) and right lane changes (right)

approach there are 4 AOIs.

– Overall complexity is the result of concatenating the complexity of each theoretical step, taking into account how they interact with each other.

- Modularity: in our context, modularity was considered as a measure for the cohesiveness and separation of concerns of the theoretical approach's components [25]. The modular pipeline design described in Section IV is characterized by the conceptualization of constraints and checks that can be easily added to the pipeline. Moreover, modularity of the approach is upheld by the division of the algorithm for Morton value sampling and plausibility checker in different blocks that can be activated in different steps, as explained in Section IV.
- Modifiability: for this quality, we utilized the definition from Bengtsson et al.: "The modifiability of a software system is the ease with which it can be modified to changes in the environment, requirements or functional specification" [26]. We contribute towards fulfilling this

TABLE I
PERFORMANCE OF THEORETICAL APPROACH

Theoretical step	Time complexity
Generate Morton for straight driving AOIs	$O(p * q)$
Generate Morton for lane switch AOIs	$O(p^q)$
Plausibility Checker round 1	$O(q)$
Generate Morton sequences between AOIs	$O((p^q)^y)$
Plausibility Checker round 2	O(q*y)
Simulator	$O(((p^q)^y) * (q * y))$
Overall complexity	$O((p^q)^y)$

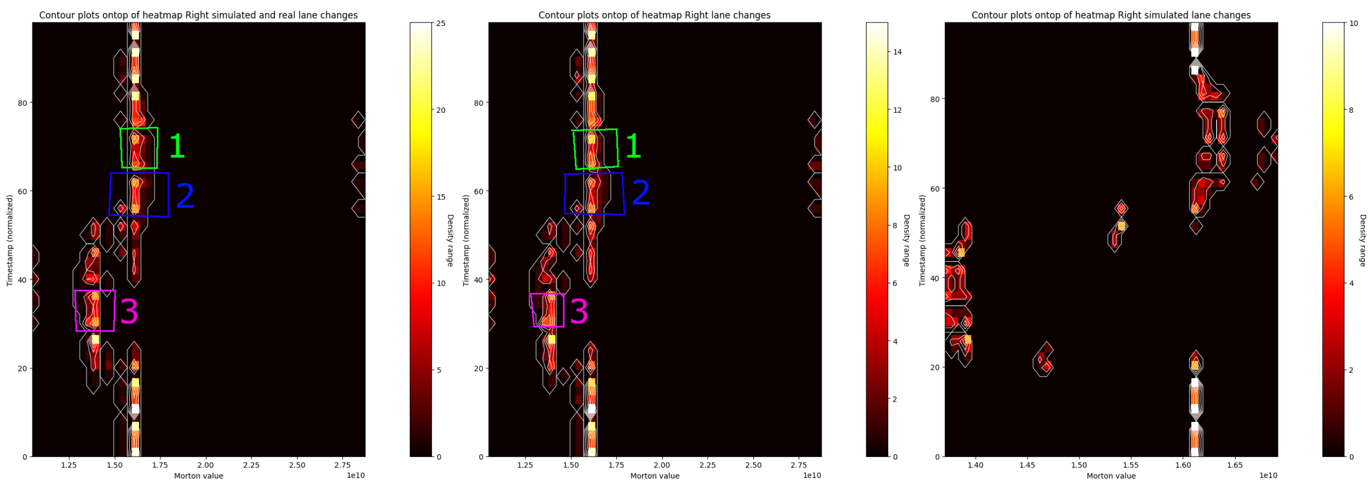

Fig. 28. Contour on-top heatmap plots containing right lane changes with real and simulator data (left), right lane changes (middle) and right simulator lane changes (right)

quality by ensuring that each of the checks are theoretically sound in terms of not only being limited to the constraints. For example, we envision the implementation of the plausibility checks with variable parameters that can be easily adjusted to accommodate changes in the required characteristics of the maneuvers to be generated (such as allowing for less smooth steering wheel angle evolution within a certain threshold).

- Completeness: the completeness of the approach refers to the extent of lane change maneuvers we would cover. With regards to the constraints defined for lane change maneuvers, we cover events performed in a smooth road environment with smooth evolution of steering wheel angle, and we focus on systematically generating lane changes that vary in “harshness” (i.e., amplitude of the steering wheel angle sensor values, and consequent Y-Acceleration sensor readings).

The prototype implementation of the Morton value sampling algorithm described in Section IV-C uses the values shown in Figure 29 for areas of interest dimensions. Within each area of interest, Morton value permutations are generated. The concatenation of each area’s unique permutations results in uniquely generated lane change maneuvers.

		Morton value range for each area of interest (using same scale as SFC-over-time: 1e10)			
		Straight before LC	Straight after LC	Left side during LC	Right side during LC
Lane change	LC 1	1.61	1.61	1.39-1.46	1.62-1.63
	LC 4	1.61	1.61	1.35-1.54	1.62-1.71
	LC 9	1.62	1.61	1.4-1.55	1.62-1.65
	LC 15	1.6	1.6	1.05-1.4	1.65-2.85
Overall min-max value range		1.6-1.62	1.6-1.61	1.05-1.55	1.62-2.85

		Duration of each area of interest (using same scale as SFC-over-time: 1e7+1.663765e15 μs)			
		Straight before LC	Straight after LC	Left side during LC	Right side during LC
Lane change	LC 1	0.15	0.15	0.3	0.3
	LC 4	0.17	0.17	0.26	0.26
	LC 9	0.2	0.2	0.3	0.3
	LC 15	0.18	0.18	0.24	0.24
Average duration		0.18	0.18	0.28	0.28

Fig. 29. Areas of interest measurements (Dataset B)

With regards to the evaluation results of the algorithm prototype discussed in Section IV-C, the time complexity has been determined as follows:

- Generation of permutations within one area of interest: O(columns^{rows}).
- Generation of permutations between areas of interest: O(n^{y}), where n is the amount of permutations of a box, and y is the amount of boxes (in our implementation y=4).

Based on the aforementioned rules and implementation, the Algorithm produced 15,180 permutations.

VI. Analysis & Discussion

A. Parameters to base CSPs on

The results of which parameters to base the CSP on are more general and a foundational step for the research. These results are required by each RQ based on the following:

- RQ1: CSPs must be defined in a way that provides clarity in answering RQ1 for identifying similarities and differences between CSPs of similar events.
- RQ2: CSPs must be defined in order to determine which properties are necessary and sufficient in estimating the dimensions of the state space.
- RQ3: CSPs must be defined in order to have an area in which we can permute in.

B. Research Question 1

The results in relation to RQ1 show that through an SFC-over-time plot, we can visualize similarities and differences between the CSPs for similar events. The ”S” shape similarity that was described in the results is consistent amongst all lane changes, the difference being the width of the ”S” shape, which seems to be impacted by harsher maneuvers.

Prior to the SFC-over-time plots, we suspected Hilbert encoding to outperform Morton encoding in regards to pattern consistency due to its superior locality preservation [27], and due to the CSP having a clearer distinction in terms of three pattern groups that can be see annotated in Figure 14, compared to the Morton generated CSP equivalent that can be seen in Figure 15.

In relation to the research goal, the results of the ”S” shape similarities and differences indicate that these similarities could be kept a constant, whilst the differences in terms of the outward jumping points at x=4 and x=0.5 in figure 20 would be what would be permuted/changed per permutation for the. In other words, RQ1 directly contributes to the RQ3 in terms of defining what must be maintained and what can be adjusted when making permutations in RQ3.

The outward jumping of points at x=4 and x=0.5 in figure 20 at higher Y-accelerations indicated initially a potential for the boundaries in terms of start and end of the CSP to correspond to how harsh of a turn is made for the lane change. This correlation provided us with an assumption that edge cases in terms of when the car loses maneuverability could be used to also define the edges/dimensions of the CSP which directly contributes to RQ2. Although, due to the issues mentioned in Section VI-D1, we found that these can not act as boundaries in which every Morton value corresponds to a lane change, hence the need to specify more concise areas using Density plots as discussed in Section VI-D2.

We want to emphasize that the results discussed in this section assist the overall permutation strategy of RQ3 and hence can contribute to literature regarding generation in scenario based testing as described in Section II-B. Lastly in relation to Section II-A, we believe the discussion in this section highlight that CSPs (with refinements based on the domain/goal) can be applied to situations where permutations must be made on them. An example of such refinements would be an SFC-over-time plot that also provides the temporal domain as in our results.

Regarding the ordering of sensors, we have not concluded which ordering of the steering wheel angle and acceleration-Y sensor is better, but acknowledge that the ordering of Y-acceleration and Steering wheel angle may reduce the length of the CSP, which may provide efficiency benefits for future research if an algorithm based on our theoretical approach is created due to reducing the state space in which permutations can be made.

Lastly, we want to mention that the results of RQ1 were derived from CSPs using the Acceleration-X and Acceleration-Y sensor. Later in the research the Acceleration-X sensor was replaced with the Steering Wheel Angle sensor. Even though different sensors were used, we found the ”S” shape pattern to still hold, and hence deem the results of RQ1 to remain valid. An example of the ”S” shape pattern with the Steering Wheel angle and Acceleration-y sensor can be seen within the combination of annotations 2 and 3 in figure 2

What are similarities and differences in characteristic stripe patterns (CSPs) on SFCs for similar events?

C. Research Question 2

To summarize our answer to RQ2, the temporal domain and occurrence of Morton values are both necessary, and when combined, are sufficient to estimate the dimensions of the state space for certain events and scenarios. The temporal domain and occurrence of Morton values can be visualization using an SFC-over-time plot.

As discussed in Section V, we found that the temporal domain of a CSP is necessary to estimate the state space of a maneuver, and sufficient when combined with the occurrence of Morton values. We want to emphasize that if the temporal domain is not used to estimate the dimensions of a state space, the variability of maneuvers (such as what occurs before and after a lane change), could impact stripe occurrences and allow for false positives, as well as false negatives in terms of maneuver identification. From this, we determine that the temporal occurrence is necessary to provide confident estimations on the dimensions of a state space.

An SFC-over-time plot provides a refined visualization of a CSP showing the temporal occurrence of Morton value of a CSP. As discussed in Section III-B, the SFC-over-time plots provide insights into the clustering of Morton values, allowing the identification of areas of interest in which Morton values occur related to the time which in turn define the dimensions of the state space.

In terms of the research goal, these results show that it is possible to permute on a single-dimensional representation of sensor data to create variations of a particular maneuver via the temporal occurrence of Morton values due to forming ”clusters” which indicate areas in which Morton values for maneuvers have a higher chance of occurring. We want to highlight that although the permutation algorithm would permute a 2D space in terms of both, Morton values and their corresponding time stamp, the main performance benefit is that many sensor values can be packed into one Morton value (ensuring that there is a consistent 2D dimension in which we permute, regardless of the number of vehicle sensors). One aspect to consider is that points generated within these areas of interest are not guaranteed to be a part of a lane change, and hence subsequent plausibility checks must be made, which is discussed further in Section VI-D1.

In context to the related work mention in Sections I-B and II, this result contributes towards the efficiency of generation of scenarios, as a scenario could be created based on the sequence of permuted Morton and Timestamp value pairs. Although, as discussed in Section VI-D1, the plausibility of such Morton sequences must still be confirmed - implying a performance hit due to further data processing being required. The results show the use case of CSPs in terms of being able to define maneuvers based on state space dimensions via the SFC-over-time representation of CSPs, and hence contributes as a possible use case to the lack of application of CSPs as mentioned in Section II-A. Furthermore, in relation to II-B, through being able to estimate the dimensions of a state-space, more diverse scenarios as well as all variations could theoretically be generated as we would know what is considered a lane change based on the dimensional limits, and hence enables the sampling of all variations of a lane change profile, such as more harsh lane changes and smoother lane changes - although, we want to note that the scope of our theoretical algorithmic approach is more constrained within this thesis, as discussed in Section IV.

D. Research Question 3

To summarize our answer to RQ3, with our theoretical approach we are able to systematically permute unique and valid lane change maneuvers with specific constraints on the chosen sensors (i.e., Steering Wheel Angle and Y-Acceleration) that describe such maneuvers. The approach is not as efficient as expected due to the way Morton value sampling has revealed itself to be more complex than anticipated, but nonetheless the use of SFCs greatly reduces the amount of data that needs to be processed from multi-dimensional sensors.

Considering the overall objective discussed in Section I-C, the thorough analysis of Morton-based SFCs acts as a solid foundation for future work to build on. Furthermore, the theoretical approach proposed has been envisioned to be easily adaptable and extensible for eventual changes in context and requirements, which should make it easily usable in future research.

With regards to contributing to the state-of-the-art knowledge and issues discussed in Sections I-B and II, our research provides useful insights and implications that are entailed with the use of SFCs for scenario-based test case generation. The proposed theoretical approach enables for generation of diverse and valid lane change maneuvers, with the benefit of using Morton-based SFCs for generation of events based on multi-dimensional sensor data.

The following subsections aim to address each of the results with respect to RQ3, the overall research goal and landscape of the related works.

1) Morton value issue: Reflecting on the results gathered from the Morton value issue, we observe that:

- Outliers have a particular reason for appearing in the SFC-over-time plots, and therefore the algorithm must generate permutations in more specific areas. These specific areas are better defined in Sections V-D2 and VI-D2.
- Although more specific areas of interest can be defined, as explained in Section VI-D2, Morton values cannot be randomly sampled within these areas. In our suggested theoretical approach (explained in Artifact Section IV), there needs to be an appropriate approach ensuring that the unpacked sensor values resulting from the sampled Morton values follow the constraints necessary to describe a lane change maneuver.

The improved understanding of Morton encoding helps us in the advancement of RQ3, since it enables us to focus on the validation of Morton sequences with a clearer understanding of the steps that need to be taken after permutations are generated. Furthermore, the results about Morton encoding analysis (as shown in Section V-D1 and the considerations entailed with SFC-over-time sampling can help related research in terms of providing a clear explanation for overcoming issues with such concepts, as discussed in Section II-A. Importantly, since our analysis was performed in the context of vehicle sensors data, we provide useful insights into the considerations (as well as potential limitations) that future work that utilizes the SFC approach for scenario-based testing for AD systems need to take into account.

2) Density Analysis of SFC-over-time Plot Clusters: Regarding the results of the Density plots, these directly build on RQ2's results by utilizing the SFC-over-time plots to identify more dense areas. This contributes directly to the Algorithm artifact of RQ3, as the areas in which Morton values should be permuted can be defined by the arbitrarily formed areas that are made visible from the Density plots. Morton values sampled from these areas of interest are more likely to be a part of the respective maneuver, in this case, lane change maneuvers - more on this is discussed in the coming paragraphs.

Building on the results of the density plots, we see that the spike plots provide an overview on where Morton values are more likely to occur, the contours make this more digestible via a 2 dimensional visualization and when paired with a heatmap, provide a way to visualize and identify Morton ranges that are more related to a lane change. Furthermore, we want to highlight that less dense areas, such as the outliers after X=2.75 in the spike plot containing all lane changes in 26, do not necessarily mean that area/Morton range should be ignored. Rather, we see this as an opportunity for future research to add more lane changes to these visualizations, resulting in more density within that area - in other words, contours which have more distance between one another (less dense areas) in the contour plot containing all lane changes in 27 could be seen as a base of a peak, and if more lane changes are added, it would highlight within that area an even more concise range to sample Morton values from. To summarize this, if an area in the state-space of figure 27 does not have much density, it does not necessarily mean that area contains noise - rather, more and diverse data (with maneuverability edge cases, regular lane changes etc) could expose this as if an area is made more dense, it would is more likely to not be noise.

Another aspect we feel is important to discuss is that the density in density plots are relative to one another. The highest density of points that can be seen in the plots in figure 27 are for straight driving parts of the lane change (Below Y=20). Based on this, it may be useful for future work to extract and overlap specific parts of a lane change, which may provide more accurate areas of interest for specific aspects of the maneuver, and if this is done for all the different aspects of a lane change (i.e. straight drive, lane change, stabilization, straight drive), very concise areas of interest/Morton values that are part of the lane change may be derived. This would again contribute to RQ3, as the more concise areas of interest that can be defined, the more we can reduce the space in which we permute in whilst also increasing confidence in Morton values actually being related to a lane change maneuver.

Although the area from which to sample Morton values is more concise via density plots, this still does not guarantee that Morton values within these areas of interest are part of a lane change individually or in sequence with other Morton values. In other words, a lane change maneuver will fall within these areas of interest, but it is not guaranteed that any Morton point individually or in sequence is valid if sampled from these areas. This is due to the way Morton values are fundamentally encoded, and more info is discussed in Section VI-D1. Due to this, plausibility checks will have to be made on the decoded sensor values that correspond to a permuted Morton sequence, as discussed in Section IV-B.

In terms of the research goal mentioned in Section I-C, these results show that it is possible to define areas of interest in an SFC-over-time plot via the density visualizations. This, in turn, allows for the algorithm to permute Morton-timestamp pairs, reducing the dimensions to permute on from many sensors to two parameters (Morton and timestamp). Furthermore, with more concise areas of interest that raise our confidence in plausible lane changes being generated, future research can utilize this method of defining areas of interests when implementing a maneuver permutation algorithm, to which real-world datasets can be verified against in terms of their coverage and diversity.

In context to the landscape related work mention in sections I-B and II, the results discussed in this subsection show that the SFC-over-time representation of CSPs can be utilized to define areas in which we are more likely to have Morton occurrences for a particular maneuver. This is the first step in creating an efficient algorithm for scenario based generation due to reducing the number of parameters we permute to 2, rather than the number of sensors of the vehicle. Regarding the issues faced in diversity of scenario based generation [6], this approach in identifying areas of interest is dependent on the amount of data used. Although we do find outliers at around X=2.75, there is potential for outliers to be further out for edge cases where the car may be about to lose maneuverability for example. Therefore, we suggest for future works to normalize SFC-over-time plots by time, overlap them and generate corresponding density plots based on a dataset with a large variety of safety-critical scenarios, edge cases where the car is about to lose maneuverability, and normal lane changes such that the impact on the areas of interest from the overall profiles of a lane change (such as edge case scenarios, safety-critical scenarios etc.) can be determined. In regards to the simulator data added to the density plots results, the visualizations of the left most plot in figure 28 do not indicate that the Morton ranges themselves varied when compared to the middle and right most plot that it is comprised of, and hence it can be deemed that a simulator can provide data to be used for determining areas of interest via being normalized and visualized with density plots. This synthetic generation of data via a simulator may assist in terms of generating more edge case scenarios and scenarios that are much rarer in reality [6], resulting in a more diverse range areas of interests from which to sample Morton values to allow for the permutation of these rare scenarios. Overall, the aforementioned discussion also contributes to the related work landscape described in II-B as well as overall research goal, as it contributes towards the ultimate goal of having an algorithm that produces all permutations, to which real-world datasets can be evaluated against to determine if 'enough' scenarios are covered.

3) Artifact: With regards to the proposed theoretical approach for lane change generation, this section analyses the results obtained with the measurements aforementioned in Section V-D3:

- Performance: time complexity of the suggested approach is heavily impacted by the Morton value sampling steps. Alternative ways of generating valid Morton value permutations that improve the complexity of the algorithm would be very useful for addressing larger amount of data (for example if increasing the resolution of columns, or the sampling rate of data). Furthermore, the use of the simulator represents a trade off for confidence in validity of permutations and cost of the overall plausibility checks of generated scenarios.
- Modularity: we considered this quality to be fundamental in our approach, as it enables extensibility of the system, for example by allowing the addition or removal of constraint to the plausibility checker with as little impact as possible on the other components. The modularity of the theoretical approach's suggested design is limited by the amount of information that goes from one component to the other: in our case, both the Morton values per AOI and which AOI they belong to must be passed as theoretical arguments throughout the pipeline.
- Modifiability: Since the theoretical approach is characterized by several constraints in terms of steering wheel angle data, we want to ensure that as requirements are likely to change the aforementioned constraints, these changes can be implemented without too many implications. Similarly as for modularity, limitations with this quality are related to the theoretical arguments that are passed throughout the pipeline, as they negatively influence the ease with which an implementation could be modified with changes in requirements.
- Completeness: We consider this as a key qualitative measure that future research can build on top of, with the hope of covering a broader spectrum of events. The constraints specified in the theoretical approach greatly limit the characteristics of maneuvers that can be covered, nonetheless a wide variety of maneuvers would still be generated within those constraints. Specifically, we would not be able to cover lane changes that are not performed smoothly in terms of changes in steering wheel angle, and we would also not encompass lane changes where the sensor noise and environmental factors are taken into account, but we cover variety in terms of "harshness" (i.e., amplitude) of steering wheel angle.

The contributions introduced through the theoretical approach are relevant for answering RQ3 in terms of addressing the necessity for systematic and efficient generation of vehicle maneuver data. The overall pipeline suggested in Section IV has relevant improvement points with regards to performance.

However, the use of SFCs reveals itself useful for reducing the number of dimensions to generate permutations on. The proposed approach has been designed to be adaptable and extensible in terms of maneuver-specific constraints, which enables the generation of more diverse test case datasets, issue discussed in the context of scenario-based test generation for AD systems [6]. Moreover, the constraints defined in the generation step (e.g., smoothness and symmetrical evolution of steering wheel angle) have safety-critical impacts in the generated events, which means that safety-critical scenarios can be generated through the control of specific parameters, contributing to the systematic classification of such events [16]. Finally, the performance evaluation presented in Table I can be used as an initial step for the definition of coverage of "all" possible events related to a specific maneuver [15].

E. Threats to Validity

1) Construct validity: Analyzing the performance of the theoretical approach based on the time complexity is not a complete estimation of how long it would take to run an implementation of it. For example, simulator time complexity grows linearly with the size of its input, but to measure its time performance, we would need to run tests on an actual simulator. This can present a threat to validity in terms of providing a somewhat incomplete view over the performance of the approach, which ultimately would have to be judged with actual measurements.
2) Internal validity: Due to the limited event data provided to base our research on, as well as the related sensors, our findings related to CSP on SFCs might be inaccurate. This issue is supported by the confirmation bias of looking for patterns that is implicitly present in the research, since the incentive is to find patterns with the use of SFCs to validate the results. As a consequence, we believe that more vehicle maneuver data and other sensors could provide different insights on the choice between Morton or Hilbert based SFCs.
3) External validity: Our research has been focused on a specific type of maneuver (i.e., lane changes), and the artifact described in Section IV is characterized by many constraints (mainly concerning environment conditions, sensor noise, and smoothness of the maneuver). Thus, the claims that are made (e.g., CSPs work similarly for other maneuvers, plausibility checker works for other sensors and situations), are based on an understanding of the concept from a specific perspective with constraints in mind, which might represent a threat to the generalization of our findings.

VII. Conclusion and Future Work

The conducted research lays the foundations for exploring generation of scenario-based tests using SFCs-based approach for permutation of multi-dimensional vehicle sensor data. The analysis of Morton-based SFCs suggests that a random sampling of Morton values inside AOIs of an SFC-over-time plot of a CSP is not enough to ensure the expected characteristics of the state space of generated events. However, the usage of density plots, which measure the frequency of Morton values occurring at a particular timestamp, provides a way to define areas in which Morton values are more likely to occur for particular events, such as lane changes - this reduction of the value range to sample Morton values from increases the chance of sampling valid Morton values.

Although, even with more concise areas, the plausibility of permuted lane changes still can not be ensured, and hence plausibility checks must be run on decoded sensor data to ensure confidence. A challenge still present is in the performance implications of such an approach that arise when generating plausible lane change maneuvers. Future research can improve upon this issue of performance in particular by focusing on how Morton values are permuted from the areas of interests defined by the density plots, as well as an alternative to requiring a simulator to verify if a permutation is valid in regards to domain properties.

Furthermore, future research can also expand upon the theoretical approach in making constraints more lenient, enabling the generation of a wider range of lane change events types, such as safety-critical scenarios of when the car is about to lose maneuverability.

Acknowledgments

In this section we would like to thank those who have helped us by contributing towards the success of our research project. First of all, we would like to thank our supervisor Christian Berger, for providing us with valuable support, insights, and feedback throughout the research. We are thankful to peer researcher Renyuan Huang for providing us with scripts for data normalization that were essential in the data analysis step. We are grateful towards peer researchers Christopher Axt and Alexandre Rancati-Palmer, for providing us with vehicle sensor data from the CARLA simulator [21], which was useful to further validate our analysis of SFCs for lane change events. We would like to thank peer researchers Zizhuang Cui and Jiacheng Li for sharing information regarding taxonomy of road lanes [22], which was used along with the simulator data to validate simulated lane changes.

References

[1] A. Broggi, M. Buzzoni, S. Debattisti, P. Grisleri, M. C. Laghi, P. Medici, and P. Versari, "Extensive tests of autonomous driving technologies," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1403–1415, 2013.

[2] L. Liu, S. Lu, R. Zhong, B. Wu, Y. Yao, Q. Zhang, and W. Shi, "Computing systems for autonomous driving: State of the art and challenges," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6469–6486, 2021.

[3] G. Lou, Y. Deng, X. Zheng, M. Zhang, and T. Zhang, "Testing of autonomous driving systems: where are we and where should we go?" in *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 2022, pp. 31–43.

[4] F. Hauer, T. Schmidt, B. Holzmüller, and A. Pretschner, "Did we test all scenarios for automated and autonomous driving systems?" in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 2950–2955.

[5] P. Koopman and M. Wagner, "Challenges in autonomous vehicle testing and validation," *SAE International Journal of Transportation Safety*, vol. 4, no. 1, pp. 15–24, 2016.

[6] W. Ding, C. Xu, M. Arief, H. Lin, B. Li, and D. Zhao, "A survey on safety-critical driving scenario generation-a methodological perspective," *CoRR, vol. abs/2202.02215*, 2022.

[7] J. K. Lawder and P. J. H. King, "Querying multi-dimensional data indexed using the hilbert space-filling curve," *SIGMOD Rec.*, vol. 30, no. 1, p. 19–24, mar 2001. [Online]. Available: https://doi.org/10.1145/373626.373678

[8] C. Böhm, S. Berchtold, H.-P. Kriegel, and U. Michel, "Multidimensional index structures in relational databases," *Journal of Intelligent Information Systems*, vol. 15, no. 1, pp. 51–70, 2000.

[9] L. Birkemeyer. (2022) geofence. Screenshot. [Online]. Available: https://github.com/fatfrog9/geofence/blob/main/results/Threshold_1D_25e10.png

[10] I. Akel. (2023) algo_prototype. Github repository. [Online]. Available: https://github.com/Marble879/Thesis-Permutation-Algo

[11] I. Lukas, Akel. (2023) Sfc_scripts_fork. Github repository. [Online]. Available: https://github.com/Marble879/SFC_scripts

[12] S. O'Shaughnessy, "Image-based malware classification: A space filling curve approach," in *2019 IEEE Symposium on Visualization for Cyber Security (VizSec)*, 2019, pp. 1–10.

[13] O. Martinez-Rubi, P. van Oosterom, R. Gonçalves, T. Tijssen, M. Ivanova, M. L. Kersten, and F. Alvanaki, "Benchmarking and improving point cloud data management in monetdb," *SIGSPATIAL Special*, vol. 6, no. 2, p. 11–18, mar 2015. [Online]. Available: https://doi.org/10.1145/2744700.2744702

[14] X. Zhang, J. Tao, K. Tan, M. Törngren, J. M. G. Sánchez, M. R. Ramli, X. Tao, M. Gyllenhammar, F. Wotawa, N. Mohan *et al.*, "Finding critical scenarios for automated driving systems: A systematic literature review," *arXiv preprint arXiv:2110.08664*, 2021.

[15] J. Cai, W. Deng, H. Guang, Y. Wang, J. Li, and J. Ding, "A survey on data-driven scenario generation for automated vehicle testing," *Machines*, vol. 10, no. 11, 2022. [Online]. Available: https://www.mdpi.com/2075-1702/10/11/1101

[16] J. Bernhard, M. Schutera, and E. Sax, "Optimizing test-set diversity: Trajectory clustering for scenario-based testing of automated driving systems," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 1371–1378.

[17] E. Knauss, "Constructive master's thesis work in industry: guidelines for applying design science research," in *2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*. IEEE, 2021, pp. 110–121.

[18] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A design science research methodology for information systems research," *Journal of management information systems*, vol. 24, no. 3, pp. 45–77, 2007.

[19] Y. Levy and T. J. Ellis, "A systems approach to conduct an effective literature review in support of information systems research." *Informing Science*, vol. 9, 2006.

[20] L. Birkemeyer. (2022) Sfc_scripts. Github repository. [Online]. Available: https://github.com/fatfrog9/SFC_scripts

[21] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[22] A. Zlocki, A. König, J. Bock, H. Weber, H. Muslim, H. Nakamura, S. Watanabe, J. Antona-Makoshi, and S. Taniguchi, "Logical scenarios parameterization for automated vehicle safety assessment: Comparison of deceleration and cut-in scenarios from japanese and german highways," *IEEE Access*, vol. 10, pp. 26 817–26 829, 2022.

[23] R. Huang, "Systematically analyzing event descriptors based on sfc for testing lane change and roundabout maneuvers," 2023, to be published in May 2023.

[24] liaison, "Permutation via backtracking," Medium, 2020, accessed: April 21, 2023. [Online]. Available: https://medium.com/@guguru/permutation-algorithm-via-backtracking-39fc1bf07a33

[25] P. Johansson and H. Holmberg, "On the modularity of a system," 2010.

[26] P. Bengtsson, N. Lassing, J. Bosch, and H. Vliet, "Analyzing software architectures for modifiability," 09 2009.

[27] S. Kumar, S. Madria, and M. Linderman, "M-grid: a distributed framework for multidimensional indexing and querying of location based data," *Distributed and Parallel Databases*, vol. 35, pp. 55–81, 2017.