Constrained Sampling from a Kernel Density Estimator to Generate Scenarios for the Assessment of Automated Vehicles

de Gelder, Erwin; Cator, Eric ; Paardekooper, Jan Pieter; Op den Camp, Olaf; De Schutter, Bart

DOI
10.1109/IVWorkshops54471.2021.9669213

Publication date
2021

Document Version
Accepted author manuscript

Published in
Proceedings of the 2021 IEEE Intelligent Vehicles Symposium Workshops (IV Workshops)

Constrained Sampling from a Kernel Density Estimator to Generate Scenarios for the Assessment of Automated Vehicles

Erwin de Gelder[1,2*], Eric Cator[3], Jan-Pieter Paardekooper[1,4], Olaf Op den Camp[1], Bart De Schutter[2]

***Abstract*— The safety assessment of Automated Vehicles (AVs) is an important aspect of the development cycle of AVs. A scenario-based assessment approach is accepted by many players in the field as part of the complete safety assessment. A scenario is a representation of a situation on the road to which the AV needs to respond appropriately. One way to generate the required scenario-based test descriptions is to parameterize the scenarios and to draw these parameters from a probability density function (pdf). Because the shape of the pdf is unknown beforehand, assuming a functional form of the pdf and fitting the parameters to the data may lead to inaccurate fits. As an alternative, Kernel Density Estimation (KDE) is a promising candidate for estimating the underlying pdf, because it is flexible with the underlying distribution of the parameters. Drawing random samples from a pdf estimated with KDE is possible without the need of evaluating the actual pdf, which makes it suitable for drawing random samples for, e.g., Monte Carlo methods. Sampling from a KDE while the samples satisfy a linear equality constraint, however, has not been described in the literature, as far as the authors know.**

In this paper, we propose a method to sample from a pdf estimated using KDE, such that the samples satisfy a linear equality constraint. We also present an algorithm of our method in pseudo-code. The method can be used to generating scenarios that have, e.g., a predetermined starting speed or to generate different types of scenarios. This paper also shows that the method for sampling scenarios can be used in case a Singular Value Decomposition (SVD) is used to reduce the dimension of the parameter vectors.

I. Introduction

An essential facet in the development of Automated Vehicles (AVs) is the assessment of quality and performance aspects of the AVs, such as safety, comfort, and efficiency [1]–[3]. Because public road tests are expensive and time consuming [4], [5], a scenario-based approach has been proposed [2], [6]–[10]. As a source of information for the scenarios for the assessment, real-world driving data has been proposed, such that the scenarios relate to real-world driving conditions [7], [8], [11].

When using scenarios extracted from real-world driving data as a direct source for describing scenario-based tests, two problems arise. First, not all possible variations of the scenarios might be found in the data. Therefore, the failure modes of the AVs might not be reflected in the tests that are based on the scenarios that are extracted from real-world driving data [5]. Second, using scenarios extracted from real-world driving data might not reduce the actual testing load, because the set of extracted scenarios is largely composed of non-safety critical scenarios [5]. As a solution to this, so-called importance sampling has been introduced in order to put more emphasis on scenarios that are likely to trigger safety-critical situations [5], [9], [12], [13]. These methods [5], [9], [12], [13] have in common that they describe scenarios using parameters for which a probability density function (pdf) is estimated.

As already presented in [9], [14], we propose to estimate the pdf using Kernel Density Estimation (KDE). KDE [15], [16] is often referred to as a non-parametric way to estimate the pdf, because no use is made of a predefined functional form of the pdf for which certain parameters are fitted to the data. Because KDE produces a pdf that adapts itself to the data, it is flexible regarding the shape of the actual underlying distribution of the parameters.

Sampling from a KDE is straightforward. In some cases, however, one wants to sample from the estimated pdf while a part of the random sample is fixed. For example, one may want to assess the performance of an AV when it is driving at its maximum allowable speed. Conditional sampling allows to generate scenario-based test cases in which, e.g., the ego vehicle has a fixed speed. One approach to performing conditional sampling is to evaluate the conditional pdf and to use this for sampling. This method, however, would be highly cumbersome, especially with a higher-dimensional pdf, because marginal integrals of codimension 1 of the conditional pdf must be evaluated.

We will propose an algorithm to sample parameters from a KDE while the parameters are subject to a linear equality constraint. Our work differs from [17], [18] because these works consider (shape) constraints on the estimated pdf. The proposed algorithm can be regarded as a generalization of the conditional density estimation in [19]. Our proposed method is efficient, because the actual (conditional) pdf does not need to be evaluated. We illustrate the proposed sampling technique and its practical usefulness using an example. Furthermore, we will explain the usefulness of sampling with linear equality constraints in case parameter reduction techniques are used to avoid the curse of dimensionality that pdf estimation techniques, such as KDE, are suffering from.

This paper is organized as follows. In Section II, we first describe the problem in more detail. The proposed method is presented in Section III. Through an example, we illustrate

[1]TNO, Integrated Vehicle Safety, Helmond, The Netherlands
[2]Delft University of Technology, Delft Center for Systems and Control, Delft, The Netherlands
[3]Radboud University, Applied Stochastics, Nijmegen, The Netherlands
[4]Radboud University, Donders Institute for Brain, Cognition and Behaviour, Nijmegen, The Netherlands
[*]Corresponding author.
E-mail address: erwin.degelder@tno.nl

the correct performance of the algorithm in Section IV. We also apply the proposed method to sample different types of scenarios in Section IV. In Section V, some implications and limitations of this work are discussed. Conclusions of the paper are provided in Section VI.

II. Problem definition

In KDE, the pdf $f(\cdot)$ is estimated as follows:

$$\hat{f}(x) = \frac{1}{N}\sum_{i=1}^{N} K_H(x - x_i). \tag{1}$$

Here, $x_i \in \mathbb{R}^d$ represents the i-th data point of dimension d. In total, there are N data points, so $i \in \{1, \ldots, N\}$. In (1), $K_H(\cdot)$ is the so-called scaled kernel with a positive definite symmetric bandwidth matrix $H \in \mathbb{R}^{d\times d}$. The kernel $K(\cdot)$ and the scaled kernel $K_H(\cdot)$ are related using

$$K_H(u) = |H|^{-1/2} K\left(H^{-1/2}u\right), \tag{2}$$

where $|\cdot|$ denotes the matrix determinant. The choice of the kernel function is not as important as the choice of the bandwidth matrix [20], [21]. Often, a Gaussian kernel is opted and this paper is no exception. The Gaussian kernel is given by

$$K(u) = \frac{1}{(2\pi)^{d/2}} \exp\left\{-\frac{1}{2}\|u\|_2^2\right\}, \tag{3}$$

where $\|u\|_2^2 = u^\mathsf{T}u$ denotes the squared 2-norm of u. Substituting (2) and (3) into (1) gives

$$\hat{f}(x) = C\sum_{i=1}^{N} \exp\left\{-\frac{1}{2}(x - x_i)^\mathsf{T} H^{-1}(x - x_i)\right\}, \tag{4}$$

where $C = \frac{1}{N(2\pi)^{d/2}|H|^{1/2}}$ is a constant.

The bandwidth matrix is an important parameter of the KDE. Several methods have been proposed to estimate the bandwidth matrix based on the available data. Perhaps the simplest method is Silverman's rule of thumb [22], in which $H = h^2 I_d$ with I_d denoting the d-by-d identity matrix[1] and

$$h = 1.06 \min\left(\sigma, \frac{R}{1.34}\right) N^{-\frac{1}{5}}. \tag{5}$$

Here, σ denotes the standard deviation of the data and R is the interquartile range of the data. Another strategy is to use cross validation. As a special case, with one-leave-out cross validation, the bandwidth matrix equals

$$\arg\min_H \prod_{i=1}^{N}\left(\frac{1}{N-1}\sum_{j=1, j\neq i}^{N} K_H(x_i - x_j)\right). \tag{6}$$

Selecting the bandwidth matrix by this method minimizes the Kullback-Leibler divergence between $f(\cdot)$ and $\hat{f}(\cdot)$ [20]. Another often-used strategy is to use plug-in methods. The idea of plug-in methods is to select an initial H and then plug $\hat{f}(\cdot)$ into an equation that calculates the optimal[2] bandwidth based on a given pdf. This process is then iterated until H converges. We refer the interested reader to [20], [21], [23], [24] for details on the estimation of H. In this paper, we assume that H is given.

Sampling new data points from $\hat{f}(\cdot)$ of (4) is straightforward. First, an integer $j \in \{1, \ldots, N\}$ is randomly chosen with each integer having equal likelihood. Next, a random sample is drawn from a Gaussian with covariance H and mean x_j.

In this paper, we want to sample from (4), such that the samples satisfy the linear equality constraint:

$$Ax = b. \tag{7}$$

Here $A \in \mathbb{R}^{n_\mathrm{c}\times d}$ and $b \in \mathbb{R}^{n_\mathrm{c}}$ denote the constraint matrix and the constraint vector, respectively. It is assumed that the constraint matrix A has full rank. Note that if A has not full rank, the constraint of (7) can easily be reformulated using Gaussian elimination, resulting in a similar constraint with a constraint matrix that has full rank. In total, there are $n_\mathrm{c} < d$ constraints.

III. Method

To deal with the constraint (7), we will perform a rotation of x, such that a part of the resulting vector is fixed by the constraint (7), while the other part of the resulting vector can be freely chosen. To perform the rotation, we employ a Singular Value Decomposition (SVD) [25] of A:

$$A = U\begin{bmatrix}\Sigma & 0\end{bmatrix}V^\mathsf{T} = U\begin{bmatrix}\Sigma & 0\end{bmatrix}\begin{bmatrix}V_1^\mathsf{T}\\ V_2^\mathsf{T}\end{bmatrix} = U\Sigma V_1^\mathsf{T}. \tag{8}$$

Here, $U \in \mathbb{R}^{d_\mathrm{c}\times d_\mathrm{c}}$ and $V \in \mathbb{R}^{d\times d}$ are orthonormal matrices, i.e., $U^{-1} = U^\mathsf{T}$ and $V^{-1} = V^\mathsf{T}$. The first d_c columns of V are denoted by V_1 while V_2 denotes the remaining columns of V. Moreover, $\Sigma \in \mathbb{R}^{d_\mathrm{c}\times d_\mathrm{c}}$ is a diagonal matrix with its so-called singular values on its diagonal. Because A has full rank and $n_\mathrm{c} < d$, all singular values are strictly positive. As such, evaluating Σ^{-1} is straightforward. Now, let $\bar{x} \in \mathbb{R}^{d_\mathrm{c}}$ and $\tilde{x} \in \mathbb{R}^{d_\mathrm{u}}$ such that

$$x = V_1\bar{x} + V_2\tilde{x} = \begin{bmatrix}V_1 & V_2\end{bmatrix}\begin{bmatrix}\bar{x}\\ \tilde{x}\end{bmatrix} = V\begin{bmatrix}\bar{x}\\ \tilde{x}\end{bmatrix}. \tag{9}$$

Note that because $V^{-1} = V^\mathsf{T}$, we have $\bar{x} = V_1^\mathsf{T}x$ and $\tilde{x} = V_2^\mathsf{T}x$. Moreover, $V_1^\mathsf{T}V_1 = I_{d_\mathrm{c}}$ and $V_1^\mathsf{T}V_2 = 0$, such that substituting (8) and (9) into (7), gives

$$U\Sigma V_1^\mathsf{T}(V_1\bar{x} + V_2\tilde{x}) = U\Sigma\bar{x} = b. \tag{10}$$

This means that in order to satisfy the constraint (7), $\tilde{x}$ can take any value whereas $\bar{x}$ is fixed:

$$\bar{x} = \Sigma^{-1}U^\mathsf{T}b. \tag{11}$$

[1] If $H = h^2 I_d$, the same smoothing is applied in every direction. Therefore, the data are often normalized before applying KDE with $H = h^2 I_d$.

[2] Optimal in the sense that it minimizes a specific value, which is usually the asymptotic mean integrated squared error.

Similar as $\bar{x}$ and $\tilde{x}$, let $\bar{x}_i = V_1^\mathsf{T} x_i$ and $\tilde{x}_i = V_2^\mathsf{T} x_i$. Using this and (9), we can rewrite (4):

$$\hat{f}(x) = C \sum_{i=1}^{N} \exp\left\{ -\frac{1}{2} \begin{bmatrix} \bar{x} - \bar{x}_i \\ \tilde{x} - \tilde{x}_i \end{bmatrix}^\mathsf{T} V^\mathsf{T} H^{-1} V \begin{bmatrix} \bar{x} - \bar{x}_i \\ \tilde{x} - \tilde{x}_i \end{bmatrix} \right\}. \tag{12}$$

To ease the notation, let us use the following notation:

$$V^\mathsf{T} H^{-1} V = \begin{bmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{bmatrix}, \tag{13}$$

with $\Lambda_{11} \in \mathbb{R}^{d_\mathrm{c} \times d_\mathrm{c}}$, $\Lambda_{12} \in \mathbb{R}^{d_\mathrm{c} \times d_\mathrm{u}}$, $\Lambda_{21} \in \mathbb{R}^{d_\mathrm{u} \times d_\mathrm{c}}$, and $\Lambda_{22} \in \mathbb{R}^{d_\mathrm{u} \times d_\mathrm{u}}$. Using the Schur complement [26]

$$\Lambda_\mathrm{S} = \Lambda_{11} - \Lambda_{12} \Lambda_{22}^{-1} \Lambda_{21}, \tag{14}$$

we can write (14) as

$$V^\mathsf{T} H^{-1} V = \begin{bmatrix} I_{d_\mathrm{c}} & \Lambda_{12}\Lambda_{22}^{-1} \\ 0 & I_{d_\mathrm{u}} \end{bmatrix} \begin{bmatrix} \Lambda_\mathrm{S} & 0 \\ 0 & \Lambda_{22} \end{bmatrix} \begin{bmatrix} I_{d_\mathrm{c}} & 0 \\ \Lambda_{22}^{-1}\Lambda_{21} & I_{d_\mathrm{u}} \end{bmatrix}.$$

Substituting this in the exponent of (12) gives

$$\begin{aligned}
&\begin{bmatrix} \bar{x} - \bar{x}_i \\ \tilde{x} - \tilde{x}_i \end{bmatrix}^\mathsf{T} H^{-1} \begin{bmatrix} \bar{x} - \bar{x}_i \\ \tilde{x} - \tilde{x}_i \end{bmatrix} \\
&= \begin{bmatrix} \bar{x} - \bar{x}_i \\ \tilde{x} - \tilde{x}_i + \Lambda_{22}^{-1}\Lambda_{12}^\mathsf{T}(\bar{x} - \bar{x}_i) \end{bmatrix}^\mathsf{T} \begin{bmatrix} \Lambda_\mathrm{S} & 0 \\ 0 & \Lambda_{22} \end{bmatrix} \\
&\qquad \begin{bmatrix} \bar{x} - \bar{x}_j \\ \tilde{x} - \tilde{x}_j + \Lambda_{22}^{-1}\Lambda_{21}(\bar{x} - \bar{x}_j) \end{bmatrix} \\
&= (\bar{x} - \bar{x}_i)^\mathsf{T} \Lambda_\mathrm{S} (\bar{x} - \bar{x}_i) + \\
&\qquad \left(\tilde{x} - \tilde{x}_i + \Lambda_{22}^{-1}\Lambda_{12}^\mathsf{T}(\bar{x} - \bar{x}_i)\right)^\mathsf{T} \Lambda_{22} \\
&\qquad \left(\tilde{x} - \tilde{x}_i + \Lambda_{22}^{-1}\Lambda_{12}^\mathsf{T}(\bar{x} - \bar{x}_i)\right).
\end{aligned}$$

Using this, (12) can be written as

$$\hat{f}(x) = C \sum_{i=1}^{N} w_i \exp\left\{ -\frac{1}{2} (\tilde{x} - \tilde{x}_i')^\mathsf{T} \Lambda_{22} (\tilde{x} - \tilde{x}_i') \right\},$$

with

$$w_i = \exp\left\{ -\frac{1}{2} (\bar{x} - \bar{x}_i)^\mathsf{T} \Lambda_\mathrm{S} (\bar{x} - \bar{x}_i) \right\}, \forall i \in \{1, \ldots, N\}, \tag{15}$$

$$\tilde{x}_i' = \tilde{x}_i - \Lambda_{22}^{-1}\Lambda_{21}(\bar{x} - \bar{x}_i), \forall i \in \{1, \ldots, N\}. \tag{16}$$

To generate samples from (4) that satisfy (7), two random numbers need to be generated. First, an integer $j \in \{1, \ldots, N\}$ is randomly chosen with the likelihood of the integer j proportional to the weight w_j of (15). Next, a random sample is drawn from a Gaussian with covariance Λ_{22}^{-1} and mean $\tilde{x}_j'$. Finally, this random sample is mapped according to (9) to obtain the final random sample. The procedure for sampling is summarized in Algorithm 1.

Algorithm 1: Sampling with linear equality constraints and full bandwidth matrix.

Input : $x_1, \ldots, x_N$, A, b, H
Output: Sample x from (4) while satisfying $Ax = b$

U, Σ, V_1, $V_2 \leftarrow$ Perform an SVD of A; see (8)
$\bar{x}_1, \ldots, \bar{x}_N \leftarrow$ Map the data points using $\bar{x}_i = V_1^\mathsf{T} x_i$
$\tilde{x}_1, \ldots, \tilde{x}_N \leftarrow$ Map the data points using $\tilde{x}_i = V_2^\mathsf{T} x_i$
$\bar{x} \leftarrow$ Compute $\bar{x}$ using (11)
Λ_{11}, Λ_{12}, Λ_{21}, Λ_{22}, $\Lambda_\mathrm{S} \leftarrow$ Compute $V^\mathsf{T} H^{-1} V$ according to (13) and Λ_S according to (14)
$w_1, \ldots, w_N \leftarrow$ Compute the weights according to (15)
$j \leftarrow$ Generate a random integer $j \in \{1, \ldots, N\}$ with the likelihood of j proportional to w_j
$\tilde{x}_j' \leftarrow$ Compute the mean of the Gaussian to generate a sample from according to (16)
$\tilde{x} \leftarrow$ Generate a random sample from a Gaussian with covariance Λ_{22}^{-1} and mean $\tilde{x}_j'$
$x \leftarrow$ Compute x according to (9)

IV. Example

To illustrate the proposed method, we have applied it to generate a new set of parameters that describe scenarios. The Next Generation SIMulation (NGSIM) data set is used as a data source. The NGSIM data set contains vehicles' trajectories obtained from video footage of cameras that were located at several motorways in the US [27]. In total, 18182 longitudinal interactions between two vehicles are analyzed. In each of these longitudinal interactions, we look at the speed profile of the leading vehicle. The speed profile is split into parts of 5 s, resulting in $N = 99840$ data samples.

We first apply the method of conditional sampling in a straightforward example to illustrate that Algorithm 1 produces correct results. The second example explains the usefulness of sampling with a linear equality constraint in case a parameter reduction technique is used.

A. Sampling with a linear equality constraint

In this first example, 2 parameters describe a scenario:

1) The speed of the leading vehicle at a certain time t and
2) The speed of the same vehicle at time $t + \Delta t$ with $\Delta t = 5$ s.

The bandwidth matrix is estimated using the plug-in selector of Wand and Jones [28]. We want to sample the initial speed in case the speed is reduced by $5\,\mathrm{m/s}$. To achieve this, we use

$$A = \begin{bmatrix} 1 & -1 \end{bmatrix}, b = \begin{bmatrix} 5 \end{bmatrix}.$$

In Figure 1, the result of Algorithm 1 is shown. In total, 10^6 samples are generated and shown by the histogram. Because the histogram follows the same pattern as the actual density of the speed difference according to the KDE, it illustrates that the provided algorithm correctly samples from the KDE.

B. Applying conditional sampling with parameter reduction

Typically, more than 2 parameters are needed to describe a scenario. If the number of parameters is too high, however, the pdf estimation using the KDE suffers from the curse of dimensionality [29]. To avoid the curse of dimensionality, a

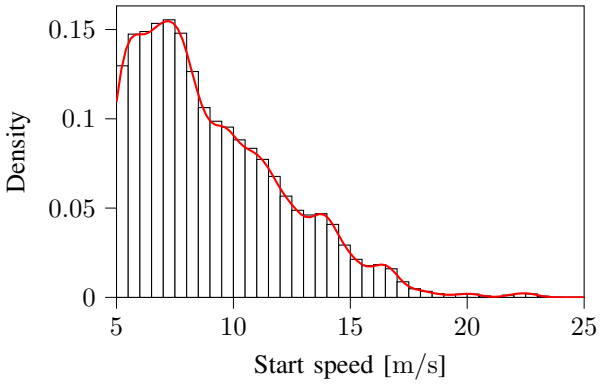


Fig. 1: The histogram shows the result of the conditional sampling according to Algorithm 1. The red line represents the true pdf.

reduction of parameters can be obtained. In this example, we use an SVD to reduce the number of parameters.

Instead of using the speed at only two time instances, we consider the following scenario parameters:

$$x_i = \begin{bmatrix} v\left(t_i\right) \\ v\left(t_i + \Delta t\right) \\ \vdots \\ v\left(t_i + n_t \Delta t\right) \end{bmatrix} \in \mathbb{R}^{n_t+1}, \tag{17}$$

where $v\left(\cdot\right)$ denotes the speed of the leading vehicle, t_i denotes the time of the i-th data point, Δt denotes the time step, and n_t denote the number of time steps. We used $\Delta t = 0.1\,\mathrm{s}$ and $n_t = 50$, so this results in 51 parameters. To reduce these 51 parameters to d parameters, consider the following matrix:

$$X = \begin{bmatrix} x_1 - \mu & \cdots & x_N - \mu \end{bmatrix} \in \mathbb{R}^{(n_t+1)\times N},$$

with $\mu = \frac{1}{N}\sum_{i=1}^{N} x_i$ and the following SVD of this matrix:

$$X = \begin{bmatrix} \bar{U}_1 & \bar{U}_2 \end{bmatrix} \begin{bmatrix} \bar{\Sigma}_1 & 0 \\ 0 & \bar{\Sigma}_2 \end{bmatrix} \begin{bmatrix} \bar{V}_1^\mathsf{T} \\ \bar{V}_2^\mathsf{T} \end{bmatrix} \approx \bar{U}_1 \bar{\Sigma}_1 \bar{V}_1^\mathsf{T},$$

with $\bar{U}_1 \in \mathbb{R}^{(n_t+1)\times d}$, $\bar{U}_2 \in \mathbb{R}^{(n_t+1)\times(n_t+1-d)}$, $\bar{\Sigma}_1 \in \mathbb{R}^{d\times d}$, $\bar{\Sigma}_2 \in \mathbb{R}^{(n_t+1-d)\times(N-d)}$, $\bar{V}_1 \in \mathbb{R}^{N\times d}$, and $\bar{V}_2 \in \mathbb{R}^{N\times(N-d)}$. Using this approximation, it follows that

$$x_i \approx \bar{U}_1 \bar{\Sigma}_1 \bar{v}_i + \mu, \forall i \in \{1, \ldots, N\}, \tag{18}$$

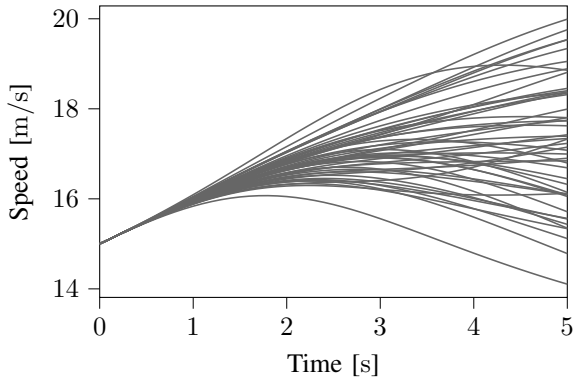
where $\bar{v}_i \in \mathbb{R}^d$ is the i-th column of $\bar{V}_1^\mathsf{T}$.

Instead of using the original data points x_i, the vectors $\bar{v}_i$ are used for the estimation of the KDE. For each sample of this KDE, the mapping of (18) is then applied to obtain the scenario parameters according to (17). In this example, $d = 4$ is used. As with the first example, the bandwidth matrix is estimated using the plug-in selector of Wand and Jones [28].

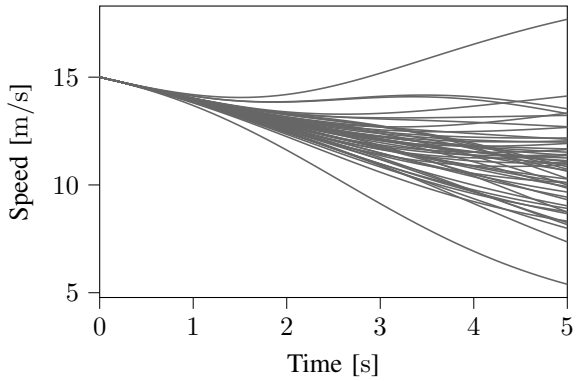
In Figure 2, 50 scenarios are sampled from the KDE in which the initial speed v_{init} and initial acceleration a_{init} are fixed by a linear equality constraint. To achieve this, the following constraint matrix and constraint vector are used:

$$A = \begin{bmatrix} \bar{u}_1 \\ \bar{u}_2 \end{bmatrix} \bar{\Sigma}_1, b = \begin{bmatrix} v_{\mathrm{init}} - \mu_1 \\ v_{\mathrm{init}} + \Delta t \cdot a_{\mathrm{init}} - \mu_2 \end{bmatrix},$$

where $\bar{u}_1$ and $\bar{u}_2$ denote the first and second row of $\bar{U}_1$, respectively, and μ_1 and μ_2 denote the first and second entry of μ, respectively. Figure 2a shows the result with $v_{\mathrm{init}} = 15\,\mathrm{m/s}$ and $a_{\mathrm{init}} = 1\,\mathrm{m/s^2}$. In Figure 2b, $a_{\mathrm{init}} = -1\,\mathrm{m/s^2}$ is used instead.



(a) $v_{\mathrm{init}} = 15\,\mathrm{m/s}$ and $a_{\mathrm{init}} = 1\,\mathrm{m/s^2}$.



(b) $v_{\mathrm{init}} = 15\,\mathrm{m/s}$ and $a_{\mathrm{init}} = -1\,\mathrm{m/s^2}$.

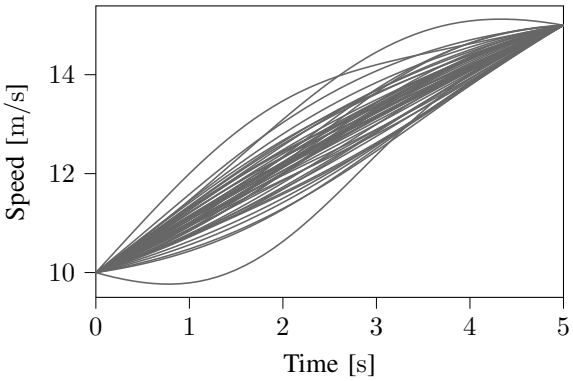
Fig. 2: 50 scenarios sampled from the KDE with a constraint on the initial speed and the initial acceleration.

In Figure 3, 50 scenarios are sampled from the KDE in which the initial speed v_{init} and end speed v_{end} are fixed by a linear equality constraint. To achieve this, the following constraint matrix and constraint vector are used:

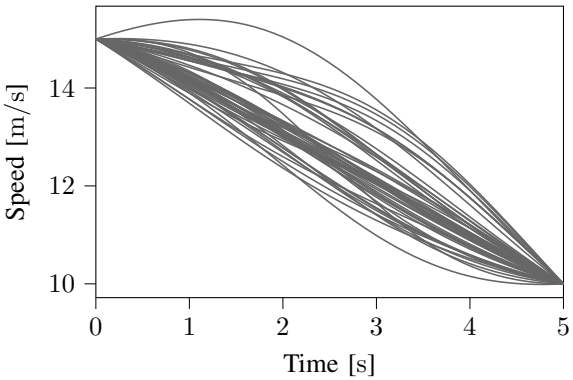
$$A = \begin{bmatrix} \bar{u}_1 \\ \bar{u}_{n_t+1} \end{bmatrix} \bar{\Sigma}_1, b = \begin{bmatrix} v_{\mathrm{init}} - \mu_1 \\ v_{\mathrm{end}} - \mu_{n_t+1} \end{bmatrix}.$$

This illustrates that the conditional sampling can be used to generated scenarios in which a leading vehicle is accelerating (Figure 3a) or decelerating (Figure 3b). Figure 3a shows the result with $v_{\mathrm{init}} = 10\,\mathrm{m/s}$ and $v_{\mathrm{end}} = 5\,\mathrm{m/s}$. In Figure 3b, the start and end speed are opposite: $v_{\mathrm{init}} = 15\,\mathrm{m/s}$ and $v_{\mathrm{end}} = 10\,\mathrm{m/s}$.

Note that all speed profiles in Figures 2 and 3 are all drawn from the same KDE. The only difference between



(a) $v_{\text{init}} = 10\,\text{m/s}$ and $v_{\text{end}} = 15\,\text{m/s}$.



(b) $v_{\text{init}} = 15\,\text{m/s}$ and $v_{\text{end}} = 10\,\text{m/s}$.

Fig. 3: 50 scenarios sampled from the KDE with a constraint on the initial speed and the end speed.

these scenarios is that different conditions are used.

V. Discussion

This paper has provided a method for sampling from a KDE such that the samples satisfy a linear equality constraint. The provided algorithm calculates weights that are used to weigh the contribution of each input data point to the overall pdf. Samples are drawn by randomly picking a random data point with the likelihood proportional to the calculated weights and adding an offset to this data point using a zero-mean multivariate Gaussian with a covariance matrix that equals the original bandwidth matrix (pre)multiplied with a rotation matrix.

The provided algorithm can be regarded as a generalization of the conditional sampling in [19]. With conditional sampling, few parameters are fixed. This is the same as considering the linear equality constraint of (7) with

$$A = \begin{bmatrix} I_{n_c} & 0 \end{bmatrix}.$$

In this particular case, the rotation of the data is not needed, so steps 2, 3, 4, and 10 of Algorithm 1 can be skipped. Note that this way of sampling becomes impractical if a parameter reduction is used as shown in Section IV-B, because the set of reduced parameters have no physical meaning anymore.

The computational cost of the provided algorithm scales quadratically with respect to number of parameters that are used to describe a single scenario (d) and linearly with the number of data points (N). Because the number of data points is generally much larger than the dimension of the data points, let us assume that $N \gg d$. Looking at Algorithm 1 and considering $N \gg d$, steps 2, 3, and 6 are the most time consuming, because these steps contain a loop over the data points. It is easy to see that the number of computations of these steps scales linearly with N. Since these computations contain a multiplication of a d-by-d matrix and a vector with d rows, the computational cost scales quadratically with d.

If we want to sample multiple times using the same linear constraint, it suffices to perform steps 1 till 6 of Algorithm 1 only once. Step 7 of Algorithm 1 does not depend on d and scales linearly with N [30]. Steps 8 till 10 of Algorithm 1 do not depend on N and scale quadratically with d. Because these steps do not depend on N and $N \gg d$, the computational cost of these steps is minor compared to step 7 of Algorithm 1. Therefore, if we want to draw many samples, i.e., more than N, the computational cost is dominated by the computational cost of step 7, which means that, in that case, the computational cost scales linearly with N.

Another application of the conditional sampling is to predict how the future will develop based on some initial conditions. For example, Figures 2a and 2b each show 50 possibilities for the speed of the leading vehicle in the next $5\,\text{s}$ given an initial speed and an initial acceleration. This could be used, for instance, to determine real-time a worst-case scenario, such that an AV could proactively respond to such a scenario. Similarly, when using Bayesian networks for predicting continuous variables [31], our algorithm provides a way to sample from the conditional densities.

Note that for an efficient scenario-based assessment of an AV, scenarios that might lead to critical behavior need to be emphasized. Several techniques are proposed in literature to emphasize these scenarios, such as reachability analysis [32], boundary searching [33], and importance sampling [5], [9], [12], [13]. With importance sampling, a different pdf is used to sample scenario parameters, such that more emphasis is put on scenarios that might lead to critical behavior. Our proposed method for conditional sampling can be combined with the importance sampling techniques explained in [9], [12], [13].

This work comes with limitations. It should be emphasized that the method only works when using a Gaussian kernel for the KDE. In practice, this is usually not a problem, because the choice of the kernel is not crucial [21]. Another limitation is that our method cannot be extended to deal with (linear) inequality constraints. If these inequality constraints are not too severe, however, straightforward rejection sampling could be used in that case, i.e., sample data points until a data point satisfies the linear inequality constraint. It should also be noted that in practice, more parameters for describing a scenario might need to be considered. For example, the state of neighboring vehicles (instead of only the leading

vehicle), lane curvature, etc. Although these parameters are not considered in the example of this paper, a parameter reduction technique as explained in Section IV-B can still be useful in case these parameters are considered.

VI. Conclusions

It is expected that scenario-based test descriptions become more and more important for the assessment of Automated Vehicles (AVs). One way to generate these scenario-based test descriptions is to sample the scenario parameters from a probability density function (pdf). To deal with the unknown shape of the pdf, it is proposed to estimate the pdf using Kernel Density Estimation (KDE). In this paper, we have shown how these parameters can be drawn from the estimated pdf while these parameters are subject to a linear equality constraint. Through an example, we have illustrated the effectiveness of our method by generating different scenarios of a longitudinal interaction with a leading vehicle.

Future work involves applying this method using more complex scenarios, e.g., scenarios that contain several different actors, to generate scenario-based test cases for the safety assessment of AVs.

References

[1] K. Bengler, K. Dietmayer, B. Färber, M. Maurer, C. Stiller, and H. Winner, "Three decades of driver assistance systems: Review and future perspectives," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 4, pp. 6–22, 2014.

[2] J. E. Stellet, M. R. Zofka, J. Schumacher, T. Schamm, F. Niewels, and J. M. Zöllner, "Testing of advanced driver assistance towards automated driving: A survey and taxonomy on existing approaches and open questions," in *IEEE 18th International Conference on Intelligent Transportation Systems*, 2015, pp. 1455–1462.

[3] P. Koopman and M. Wagner, "Challenges in autonomous vehicle testing and validation," *SAE International Journal of Transportation Safety*, vol. 4, pp. 15–24, 2016.

[4] N. Kalra and S. M. Paddock, "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?" *Transportation Research Part A: Policy and Practice*, vol. 94, pp. 182–193, 2016.

[5] D. Zhao, X. Huang, H. Peng, H. Lam, and D. J. LeBlanc, "Accelerated evaluation of automated vehicles in car-following maneuvers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 733–744, 2018.

[6] S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer, "Survey on scenario-based safety assessment of automated vehicles," *IEEE Access*, vol. 8, pp. 87 456–87 477, 2020.

[7] H. Elrofai, J.-P. Paardekooper, E. de Gelder, S. Kalisvaart, and O. Op den Camp, "Scenario-based safety validation of connected and automated driving," Netherlands Organization for Applied Scientific Research, TNO, Tech. Rep., 2018.

[8] A. Pütz, A. Zlocki, J. Bock, and L. Eckstein, "System validation of highly automated vehicles with a database of relevant traffic scenarios," in *12th ITS European Congress*, 2017, pp. 1–8.

[9] E. de Gelder and J.-P. Paardekooper, "Assessment of automated driving systems using real-life scenarios," in *IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 589–594.

[10] J. Antona-Makoshi, N. Uchida, K. Yamazaki, K. Ozawa, E. Kitahara, and S. Taniguchi, "Development of a safety assurance process for autonomous vehicles in Japan," in *26th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, 2019, pp. 1–18.

[11] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "The highD dataset: A drone dataset of naturalistic vehicle trajectories on German highways for validation of highly automated driving systems," in *IEEE 21st International Conference on Intelligent Transportations Systems (ITSC)*, 2018, pp. 2118–2125.

[12] Y. Xu, Y. Zou, and J. Sun, "Accelerated testing for automated vehicles safety evaluation in cut-in scenarios based on importance sampling, genetic algorithm and simulation applications," *Journal of Intelligent and Connected Vehicles*, vol. 1, pp. 28–38, 2018.

[13] S. Jesenski, N. Tiemann, J. E. Stellet, and J. M. Zöllner, "Scalable generation of statistical evidence for the safety of automated vehicles by the use of importance sampling," in *IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2020, pp. 1–8.

[14] E. de Gelder, A. Khabbaz Saberi, and H. Elrofai, "A method for scenario risk quantification for automated driving systems," in *26th International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, 2019.

[15] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[16] M. Rosenblatt, "Remarks on some nonparametric estimates of a density function," *The Annals of Mathematical Statistics*, vol. 27, no. 3, pp. 832–837, 1956.

[17] P. Hall and B. Presnell, "Density estimation under constraints," *Journal of Computational and Graphical Statistics*, vol. 8, no. 2, pp. 259–277, 1999.

[18] M. A. Wolters and W. J. Braun, "A practical implementation of weighted kernel density estimation for handling shape constraints," *Stat*, vol. 7, no. 1, e202, 2018.

[19] M. P. Holmes, A. G. Gray, and C. L. Isbell, "Fast nonparametric conditional density estimation," in *23rd Conference on Uncertainty in Artificial Intelligence*, 2007, pp. 175–182.

[20] B. A. Turlach, "Bandwidth selection in kernel density estimation: A review," Institut für Statistik und Ökonometrie, Humboldt-Universität zu Berlin, Tech. Rep., 1993.

[21] T. Duong, "ks: Kernel density estimation and kernel discriminant analysis for multivariate data in R," *Journal of Statistical Software*, vol. 21, no. 7, pp. 1–16, 2007.

[22] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*. CRC press, 1986.

[23] M. C. Jones, J. S. Marron, and S. J. Sheather, "A brief survey of bandwidth selection for density estimation," *Journal of the American Statistical Association*, vol. 91, no. 433, pp. 401–407, 1996.

[24] A. Gramacki and J. Gramacki, "FFT-based fast bandwidth selector for multivariate kernel density estimation," *Computational Statistics & Data Analysis*, vol. 106, pp. 27–45, 2017.

[25] G. H. Golub and C. F. Van Loan, *Matrix Computations*. John Hopkins University Press, 2013, vol. 3.

[26] F. Zhang, *The Schur Complement and Its Applications*. Springer Science & Business Media, 2006, vol. 4.

[27] V. G. Kovvali, V. Alexiadis, and L. Zhang, "Video-based vehicle trajectory data collection," in *Transportation Research Board 86th Annual Meeting*, 2007.

[28] M. P. Wand and M. C. Jones, "Multivariate plug-in bandwidth selection," *Computational Statistics*, vol. 9, no. 2, pp. 97–116, 1994.

[29] D. W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization*. John Wiley & Sons, 1992.

[30] M. D. Vose, "A linear algorithm for generating random numbers with a given distribution," *IEEE Transactions on Software Engineering*, vol. 17, no. 9, pp. 972–975, 1991.

[31] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[32] M. Althoff and S. Lutz, "Automatic generation of safety-critical test scenarios for collision avoidance of road vehicles," in *IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2018, pp. 1326–1333.

[33] J. Zhou and L. del Re, "Safety verification of ADAS by collision-free boundary searching of a parameterized catalog," in *Annual American Control Conference (ACC)*, IEEE, 2018, pp. 4790–4795.