Adaptive importance sampling for probabilistic validation of advanced driver assistance systems

O. Gietelink, B. De Schutter, and M. Verhaegen

If you want to cite this report, please use the following reference instead:
Adaptive Importance Sampling for Probabilistic Validation of Advanced Driver Assistance Systems

Olaf Gietelink, Bart De Schutter, and Michel Verhaegen

Abstract—We present an approach for validation of advanced driver assistance systems, based on randomized algorithms. The new method consists of an iterative randomized simulation using adaptive importance sampling. The randomized algorithm is more efficient than conventional simulation techniques. The importance sampling pdf is estimated by a kernel density estimate, based on the results from the previous iteration. The concept is illustrated with a simple adaptive cruise control problem.

I. INTRODUCTION

A. Advanced Driver Assistance Systems

The increasing demand for safer passenger cars has stimulated the development of advanced driver assistance systems (ADASs). An ADAS is a control system that uses environment sensors to improve comfort and traffic safety by assisting the driver. An example is adaptive cruise control (ACC), which tries to maintain a pre-defined velocity set-point, unless a slower vehicle is detected ahead [14]. The ACC then controls the vehicle to follow the slower vehicle at a desired distance $x_d$ (Fig. 1). Further defined are the position $x$, velocity $v$, and acceleration $a$ of both vehicles, the relative velocity $v_r = v_1 - v_2$, clearance $x_r = x_1 - x_2$, and separation error $e_s = x_d - x_r$.

The demand for safety naturally increases with increasing automation of the driving task, since the driver must fully rely on a flawless operation of the ADAS. The ADAS should therefore be validated for a wide set of operating conditions.

An iterative process of simulations and test drives is often used for validation. Test drives give realistic results, but can never cover the entire set of operating conditions. Results are also difficult to analyze and not reproducible [3]. On the other hand, simulations have their limitations as well. For a realistic nonlinear model and multiple traffic disturbances, the validation problem will become difficult to solve, and eventually become intractable [15]. To make the simulation phase more efficient, a controller can be validated with a grid search over the operating range of all parameters [4]. However, an exhaustive grid search requires an intractably large number of experiments. Another possibility is a Monte Carlo strategy, where the system is simulated for a representative, but still very large, set of operating conditions, based on the probability that these conditions occur [12].

In [6] we have introduced the use of importance sampling (IS) to make Monte Carlo simulation more efficient, i.e. using a much smaller number of tests. It was shown that IS considerably speeds up the simulation process, although the choice of probability density function (pdf) to sample from was not optimal. The choice of an optimal IS pdf has been the topic of much research, but has usually been constrained to simple parametric approaches [1], [5].

B. Objectives of this Paper

The objective of this paper is to present a new validation approach based on an iterative randomized algorithm (RA) using adaptive importance sampling (AIS), where the IS pdf is based on the results of the previous iteration. This provides an efficient test program in order to cover the entire set of operating conditions with a minimum number of samples.

Section II presents some background theory on RAs. Sections III and IV then present an iterative sampling algorithm and AIS to improve its efficiency, illustrated with the case study. Section V presents the conclusions.

II. RANDOMIZED ALGORITHMS

An alternative approach for solving a complex problem exactly, is to solve it approximately by using an RA. An RA is an algorithm that makes random choices during its execution [10]. The use of an RA can turn an intractable problem into a tractable one, but at the cost that the algorithm may fail to give a correct solution. The probability that the RA fails can be made arbitrarily close to zero, but never exactly equal to zero. This probability $\delta$ mainly depends on the sample complexity $N$, i.e. the number of simulations performed, but also on the specification of the problem to be solved.

A. Problem Specification

The performance of an ACC can be quantified in a number of measures $\rho_i$, e.g. tracking error, control effort, ride comfort, and string stability. In this paper the controller validation is restricted to the measure of safety, expressed as the probability $p$ that an unsafe situation occurs. In [6] we have used a discrete safety measure $\rho_s \in \{0, 1\}$, where...
\( \rho_s = 1 \) means ‘collision’ and \( \rho_s = 0 \) means ‘no collision’. However, a discrete safety measure cannot distinguish in severity between different situations for which \( \rho_s = 1 \). Therefore, a continuous safety measure \( \rho_i \) will be used, defined by the time-to-collision \( t_{i tc} = \frac{\rho_i}{v_i} \). The time-to-collision has a lower value for unsafe situations, with a threshold value \( \gamma_t = 6 \text{s} \), since a traffic situation is subjectively regarded by a driver as ‘dangerous’ when \( \rho_t < \gamma_t \) [3].

The value of \( \rho_i \) for a particular scenario depends on the perturbations imposed by that scenario, i.e. the motion of other vehicles. Apart from the acceleration of the lead vehicle \( a_1 \), also the initial conditions \( x_i(0), v_i(0) \) influence \( \rho_i \). These scenario parameters, together with driver input, and other disturbances form the parameter set \( \mathcal{Q} \): an \( n \)-dimensional bounded set of all possible parameter combinations.

Since the system performance is defined by the scalar performance measure \( \rho_i \), a cost function \( \rho_i(q) : \mathcal{Q} \to \mathbb{R} \) can be defined. The validation objective is then to identify the probability \( p \) that \( \rho_i < \gamma_t \), by estimating it by \( \hat{\rho} \) with a given accuracy \( \epsilon \) and confidence level \( 1 - \delta \):

\[
\Pr\{ | \Pr\{ \rho_i < \gamma_t \} - \hat{\rho} | \leq \epsilon \} \geq 1 - \delta, \quad \text{with} \quad \delta, \epsilon \in (0, 1). \quad (1)
\]

**B. Monte Carlo Sampling**

Consider an arbitrary process with only two possible outcomes, ‘success’ \( (\rho = 0) \) and ‘failure’ \( (\rho = 1) \), and suppose we wish to determine the probability \( p \) for a successful outcome\(^1\). A Monte Carlo sampling method can then be used [9], based on the ‘law of large numbers’, which states that

\[
\lim_{N \to \infty} \Pr\{ | \hat{\rho}_N - p | \geq \epsilon \} = 0, \quad (2)
\]

where the empirical probability \( \hat{\rho}_N \) of a failure of the process can be estimated as

\[
\hat{\rho}_N = \frac{1}{N} \sum_{i=1}^{N} J(q_i), \quad (3)
\]

which is known as the simple sampling estimator. The indicator function \( J \) represents the outcome of every \( i \)-th experiment for a parameter combination \( q_i \), given by

\[
J(q_i) = \begin{cases} 
0, & \text{if } \rho = 0 \\
1, & \text{if } \rho = 1.
\end{cases} \quad (4)
\]

The expected value of \( \hat{\rho}_N \) is \( p \), since \( \hat{\rho}_N \) is an unbiased estimator. The variance of \( \hat{\rho}_N \) is given by

\[
\sigma_{\hat{\rho}}^2 = \text{var} \left\{ \frac{1}{N} \sum_{i=1}^{N} J(q_i) \right\} = \frac{1}{N} \left[ E \{ J(q_i)^2 \} - E \{ J(q_i) \}^2 \right] = \frac{1}{N} (p - p^2) = \frac{p(1 - p)}{N}. \quad (5)
\]

The accuracy of the estimate \( \hat{\rho}_N \) can then be expressed in the relative rms error

\[
\sqrt{\frac{\text{var} \{ \hat{\rho}_N \}}{\hat{\rho}^2}} = \sqrt{\frac{1 - p}{pN}}. \quad (6)
\]

\(^1\)Please note the difference in notation between the performance level \( \rho \) for one particular experiment and its probability \( p \) for all experiments.

From (6) the sample complexity could be calculated, given a desired relative error. However, this measure also requires a priori knowledge of \( p \), which is exactly the parameter we wish to estimate. Furthermore, (6) does not give any information on the confidence interval for any particular estimate \( \hat{\rho}_N \). In that respect, \( \hat{\rho}_N \) is unlikely to be exactly equal to the real probability \( p \), although it is reasonable to expect that \( \hat{\rho}_N \) will approach \( p \), as \( N \to \infty \), and as long as the samples are chosen to be representative of the set \( \mathcal{Q} \).

The question thus arises how many samples \( N \) are necessary to give a reliable estimate \( \hat{\rho}_N \), such that it differs from the real (unknown) value \( p \) by no more than \( \epsilon > 0 \), i.e.

\[
p - \hat{\rho}_N \leq \epsilon. \quad (7)
\]

Obviously, it is important to know whether the probability that the real performance \( p \) is worse than the estimated performance \( \hat{\rho}_N \), i.e. the probability that \( p - \hat{\rho}_N > \epsilon \), as in (7). Vice versa, the probability that the real \( p \) is better than expected by \( \hat{\rho}_N \), i.e. \( \hat{\rho}_N - p > \epsilon \), is not as important.

Since \( \hat{\rho}_N \) is a random variable, the outcome of the inequality (7) is a random variable as well with a certain probability of realization. Therefore, we cannot always guarantee that \( p - \hat{\rho}_N \leq \epsilon \), even for large \( N \). By introducing a confidence level \( 1 - \delta \), the probability that \( p - \hat{\rho}_N > \epsilon \) for any \( j \)-th set of \( N \) simulations (denoted by \( N_j \)) is then defined as

\[
\Pr\{ p - \hat{\rho}_N \leq \epsilon \} \geq 1 - \delta, \quad \text{with} \quad \delta, \epsilon \in (0, 1). \quad (8)
\]

It is then of interest to know the required \( N \) for (8) to hold.

**C. Sample Complexity**

Since the process is a binomial process, (8) can be calculated by

\[
\Pr \{ p - \hat{\rho}_N \leq \epsilon \} = 1 - \sum_{i=0}^{N^{-}} \binom{N^{-}}{i} p^i(1-p)^{(N^{-}-i)} > 1 - \delta, \quad (9)
\]

where \( N^{-} = (p - \epsilon)N \) is the minimum number of samples for which \( J(q_i) = 1 \). Unfortunately, (9) cannot be solved explicitly, such that the necessary sample complexity has to be approximated. From the central limit theorem follows that the distribution of \( \hat{\rho}_N \) approaches a normal distribution as \( N \to \infty \) [1]. In that case (8) can be approximated by

\[
\Pr \{ p - \hat{\rho}_N < k\sigma \} \geq 1 - \delta. \quad (10)
\]

Here the variance \( \sigma \) can be approximated by the binomial parameter \( \sqrt{Np(1-p)} \), and \( k > 0 \) can be derived from tables for a standard Gaussian distribution with a desired confidence \( 1 - \delta \). In this way, a fast approximation for \( N \), denoted by \( N_{\text{hit}} \), can be calculated. However, this measure again requires a priori knowledge of \( p \).

To avoid this paradox we can use the Chernoff bound [2], which states that the probability \( \delta > 0 \) in (8) is no larger than \( e^{-2N\epsilon^2} \). Therefore, to estimate the unknown quantity \( p \) to an accuracy \( \epsilon \) and with a confidence \( 1 - \delta \), \( N \) should be chosen such that \( 2e^{-2N\epsilon^2} \leq \delta \). This can be rewritten as

\[
N \geq \frac{1}{2\epsilon^2} \ln \frac{1}{\delta}. \quad (11)
\]
which is known as the one-sided additive Chernoff bound. In order to compute \( \hat{p}_N \), we can then use Algorithm 1.

**Algorithm 1 (Probabilistic performance verification [13])**

Given a desired \( \epsilon, \delta \in (0, 1) \), a threshold \( \gamma \geq 0 \), and the true pdf \( f_Q \), this RA returns with a probability of at least \( 1 - \delta \) an estimate \( \hat{p}_N \) for \( p \), such that \( p - \hat{p}_N \leq \epsilon \).

1. Determine the necessary \( N \) with (11).
2. Draw \( N \) independent identically distributed (iid) samples \( q_1, q_2, \ldots, q_N \) according to its pdf \( f_Q \).
3. Return the empirical probability

\[
\hat{p}_N = \frac{1}{N} \sum_{i=1}^{N} J(q_i),
\]

where \( J(q_i) \) is the indicator function

\[
J(q_i) = \begin{cases} 
0, & \text{if } \rho \geq \gamma \\
1, & \text{if } \rho < \gamma.
\end{cases}
\]

**Example 1 (Gaussian distributed disturbance)**

Consider a scenario where the lead vehicle brakes to a full stop. The initial conditions are \( x_r, \dot{\alpha} = x_d = 40 \text{ m} \), and \( v_1(0) = v_2(0) = 30 \text{ m/s} \). For simplicity, assume that the deceleration of the lead vehicle \( \alpha_1 \) is the only disturbance (note however that the approach can easily be extended to an \( n \)-dimensional parameter set). When the lead vehicle brakes hard, the ACC vehicle cannot always brake in time, since the ACC deceleration is limited to \(-2.5 \text{ m/s}^2\). Now, for fine-tuning the control parameters, we would like to know the percentage of situations \( \rho \) for which \( p_t < \gamma_t \). The safety obviously decreases with a stronger deceleration \( \alpha_1 \), such that the function \( p_t(\alpha_1) \) is non-decreasing. For verification of this simple example the ‘true’ outcome can therefore be calculated numerically and is known exactly: \( p = 0.03630 \).

However, in practice it can be impossible to determine \( p \) in a deterministic way, when the dimension of \( Q \) increases and the function \( p_Q \) becomes non-convex. So instead of calculating \( p \) in a deterministic sense, the function is randomized in such a way that it takes a random input \( q_i \) from its pdf \( f(q) \). ACC field tests [3] suggest that the acceleration profile can be roughly described as a random signal with a Gaussian pdf \( \mathcal{N}(0,1.5) \) with mean \( \mu = 0 \) and standard deviation \( \sigma = 1.5 \), denoted as \( \mathcal{N}(0,1.5) \), truncated on the interval \([-10,10] \text{ m/s}^2\).

In order to verify Algorithm 1, we execute it \( M = 10000 \) times. Suppose it is desired that \( \epsilon = 0.01 \) and \( \delta = 0.01 \), then (11) gives \( N = 23026 \). Every \( j \)-th simulation set gives an estimate \( \hat{p}_N \) for \( j = 1, \ldots, M \). The distribution of the estimate is shown in Fig. 2. The empirical mean \( \hat{p}_M \) is 0.03633, as well as the true pdf \( f_Q \), which is known as the \( \mathcal{N}(0,1.5) \), truncated on the interval \([-10,10] \text{ m/s}^2\).

These results suggest that the desired values for \( \delta \) and \( \epsilon \) can be achieved with a much lower \( N \) than given by (11).

**III. ITERATIVE ESTIMATION USING THE BINOMIAL BOUND**

As a solution to the conservatism of the Chernoff bound, we can make an initial estimate of \( p \) using (11), and subsequently use the binomial bound (10), after which \( p \) can be estimated with a lower bound on the sample complexity.

In order to obtain a lower \( N \) in total, the values for \( \epsilon \) and \( \delta \) must be modified. From (11) follows that \( N \) is proportional to \( \log(1/\delta) \) and inversely proportional to \( 1/\epsilon^2 \). Therefore, when \( \delta \) is decreased and \( \epsilon \) increased by a suitably chosen factor \( \kappa \), a lower \( N \) can be obtained in a first iteration, that is \( \delta_1 = \delta/\kappa \) and \( \epsilon_1 = \kappa \epsilon \). To obtain the desired confidence \( 1 - \delta \) for \( \hat{p}_N \) in a second iteration, \( \delta_2 \) should be chosen such that \((1-\delta_1)(1-\delta_2) \geq 1 - \delta \). This is true when \( \delta_2 = \delta - \delta_1 \), since \((1-\delta_1)(1-\delta_2) \geq 1 - \delta \), for all \( \kappa \geq 1 \). The accuracy in the second iteration is set to its desired value \( \epsilon \). This procedure is formalized in Algorithm 2.

**Algorithm 2 (Iterative estimation using binomial bound)**

Given a desired \( \epsilon, \delta \in (0, 1) \) and a threshold \( \gamma \geq 0 \),

1. Draw \( N_{ss,1} \geq \frac{1}{2\delta^2} \ln \frac{2}{\epsilon_1} \) iid samples, where \( \delta_1 = \delta/\kappa \), and \( \epsilon_1 = \kappa \epsilon \) and \( \kappa \) is a suitably chosen real number.
2. Return the empirical probability

\[
\hat{p}_{N,1} = \frac{1}{N_{ss,1}} \sum_{i=1}^{N_{ss,1}} J(q_i).
\]

The real \( p \) is at worst \( p_{bin} = \hat{p}_{N,1} + \epsilon_1 \) with confidence \( 1 - \delta_1 \).
3. Determine \( N_{bin,2} \) for \( p_{bin} \), \( \delta_2 = \delta - \delta_1 \) and \( \epsilon_2 = \epsilon \) using (10).
4. If \( N_{bin,2} \geq N_{ss,1} \), draw \( N_{ss,2} = N_{bin,2} - N_{ss,1} \) samples.
5. Return the empirical probability

\[
\hat{p}_N = \frac{1}{N_{ss,1} + N_{ss,2}} \sum_{i=1}^{N_{ss,1}+N_{ss,2}} J(q_i),
\]

with accuracy \( \epsilon \) and confidence \((1-\delta_1)(1-\delta_2) \geq 1 - \delta \).

In order to choose a suitable value for \( \kappa \), we investigate the bounds on \( N_{ss,1} \) and \( N_{bin,2} \) for different values of \( \kappa, p, \delta, \) and \( \epsilon \), see Fig. 3. This figure shows that Algorithm 2 is
optimal when $N_{\text{bin},2} = N_{\text{ss},1}$. It is also clear that $\kappa$ can better be selected too large instead of too small. In the former case, only a small number of extra samples $N_{\text{ss},2} = N_{\text{bin},2} - N_{\text{ss},1}$ has to be drawn in the second iteration. In the latter case, the first iteration will draw too many samples $N_{\text{ss},1}$, which is a waste of simulation time. The contours are similar for other values of $\delta, \epsilon$ and $p$.

Therefore we choose $\kappa$ at $N_{\text{bin},2} = N_{\text{ss},1}$, based on a reasonable value for $p$, preferably with $p$ as low as possible. Since $p$ should be greater than $\epsilon$ ($p < \epsilon$ implies the possibility of a negative probability), $p$ is lower bounded by $\epsilon$. Therefore $\kappa$ should be chosen such that $N_{\text{bin},2} = N_{\text{ss},1}$ and $p = \epsilon$.

**Example 2 (Iterative estimation using binomial bound)**

The problem in Example 1 is repeated using Algorithm 2, for the same values $\delta = 0.01$, $\epsilon = 0.01$, and the real $p$ at 0.03630. However, since $\epsilon = 0.01$, $p$ is assumed to be lower bounded at 0.01. We therefore choose $\kappa = 3.4$, using the procedure described above and illustrated in Fig. 3. Fig. 4 shows the results for 10000 simulation sets, each with $N_j \in [3326, 4804]$, where $N_{\text{ss},1,j} = 2391$ and $N_{\text{ss},2,j} \in [935, 2413]$. The sample complexity $N_{\text{ss},2}$ in the second iteration is quite large, caused by the fact that the factor $\kappa$ is chosen larger than the optimal value (i.e. $\kappa_2 = 3.4$ at $p = 0.01$, instead of $\kappa_1 = 2.75$ at $p = 0.03630$, see Fig. 3). The results are $\hat{p}_N = 0.03619$ and the variance $\sigma^2 = 8.8040 \cdot 10^{-6}$. When the empirical confidence is set $\delta = 0.01$ the empirical accuracy is found to be $\hat{\epsilon} = 0.0070$, meaning that $N_j \in [935, 2413]$ is still slightly conservative, although better than in Example 1. Compared to the initial Chernoff bound $N = 23026$ for $\epsilon = 0.01$ and $\delta = 0.01$, this means an efficiency improvement of $rac{\max(N_{\text{bin},2,j})}{N_{\text{ss}}} = 4.8$. □

**IV. ADAPTIVE IMPORTANCE SAMPLING (AIS)**

**A. Principle of Importance Sampling (IS)**

Example 2 shows that a significant reduction of $N$ can be achieved, although the samples $q_j$ themselves are not chosen more efficiently. One solution is to reduce the set $Q$ by neglecting certain subsets that are impossible to occur [6]. Obviously, a low $t_{\text{t1c}}$ is more likely with lower values for $x_r, 0, y_r, 0,$ and $a_1$, such that the samples for which $J = 1$ are clustered in a specific subset $Q_{\text{bad}}$. This means that there is structure in the parameter set $Q$ and in the function $\rho_Q$ that can be used to reduce the sampling space by disregarding specific subsets of $Q$, of which the outcome is a priori known.

It therefore makes sense to give more attention to operating conditions that are more likely to become dangerous than others. Another possibility for using a priori knowledge on interesting samples is importance sampling, which is a technique to increase the number of occurrences of the event of which the probability $p$ should be estimated [9].

Suppose that we want to estimate a probability $p$, given the parameter set $Q$. The goal is then to estimate

$$p = \int_Q J(q) f_Q(q) dq = E \{ J(q) \},$$

where we sample $q$ from its joint pdf $f_Q$, denoted as $q \sim f_Q$. In order to highlight the interesting subset $Q_{\text{bad}}$ it thus makes sense not to sample from the original pdf $f_Q$, but instead use an artificial pdf, reflecting the ‘importance’ of the events, and then reweighing the observations to get an unbiased estimate.

We can now define an IS pdf $\varphi$ that is strictly positive on $Q$. We can then write

$$p = \int_Q J(q) f_Q(q) \varphi(q) dq = E \left\{ J(\Phi_j) f_Q(\Phi_j) \varphi(\Phi_j) \right\},$$

where $\Phi \sim \varphi$. The IS estimator based on $\varphi$ is

$$\hat{p}_{N,\text{is}} = \frac{1}{N} \sum_{i=1}^N J(\Phi_i) f_Q(\Phi_i) \varphi(\Phi_i),$$

where $\Phi_1, \ldots, \Phi_N$ are iid with pdf $\varphi$. Every sample drawn from $\varphi$ can be seen as a realization of $\hat{\varphi}$ original samples, which must therefore be reweighed with the factor $\frac{1}{\hat{\varphi}}$. The variance of the importance sampling estimator (15) is

$$\sigma^2_{\text{is}} = \text{var} \{ \hat{p}_{N,\text{is}} \} = \frac{1}{N} \sum_{i=1}^N J(\Phi_i) \frac{f_Q(\Phi_i)}{\varphi(\Phi_i)} \left( E \left\{ J(Q) f_Q(Q) \frac{f_Q(\Phi_i)}{\varphi(\Phi_i)} \right\} - E \left\{ J(\Phi_i) \varphi(\Phi_i) \right\} \right).$$

An RA can then be formulated as follows.
Algorithm 3 (Importance sampling)
Given a desired \(\varepsilon, \delta \in (0, 1)\) and a threshold \(\gamma \geq 0\),
1) Determine a strictly positive IS pdf \(\varphi\).
2) Select an initial number of samples \(N_{is}\).
3) Draw \(N_{is}\) iid samples \(\Phi_i\) according to \(\varphi\).
4) Return the empirical probability
\[
\hat{p}_{N, is} = \frac{1}{N_{is}} \sum_{i=1}^{N_{is}} J(\Phi_i) f(\Phi_i) \varphi(\Phi_i) .
\]

In order to make this algorithm work, good choices for \(N_{is}\) and the IS pdf \(\varphi\) must be made in advance. We will discuss these two issues next.

B. Estimation of \(\varphi\) using Kernel Density Estimation

The performance of Algorithm 3 heavily depends on the reliability of the pdf \(\varphi\) to generate random variables, and of the models used in the simulation. An efficient estimator of \(\hat{p}_{N, is}\) is obtained by choosing \(\varphi\) proportional to the importance of the individual samples, with importance defined as \(|J(q) f_Q(q)|\). A rare but dangerous event can thus be equally important as a frequent but less critical event. Conventional IS methods consist of shifting the mean or variance of the original pdf \(f\) to form the IS pdf \(\varphi\), a so-called parametric approach [1], [11]. However, the optimal IS pdf \(\varphi\) will most likely not be a standard type pdf (e.g. normal), but reflect the models used in the simulation. An efficient estimator of the reliability of the pdf \(\varphi\) is approximated by the empirical estimate of (16)
\[
\tilde{\sigma}^2_{is} = \frac{1}{N_{is}} \sum_{i=1}^{N_{is}} J^2(\Phi_i) \frac{f^2(\Phi_i)}{\varphi^2(\Phi_i)} - \left( \frac{1}{N_{is}} \sum_{i=1}^{N_{is}} J(\Phi_i) \frac{f(\Phi_i)}{\varphi(\Phi_i)} \right)^2 .
\]

Unfortunately, (21) can only be calculated a posteriori, whereas we would like to know \(N\) before we start with importance sampling. We therefore seek to estimate \(\tilde{\sigma}^2_{is}\) a priori, thereby predicting the IS reduction factor in (18), which in turn is used to predict \(N_{is} \approx \lambda_{is} N_{ss} \).

Suppose a limited set of samples has already been evaluated in a first iteration. Then consider the factor \(\frac{f(q_i)}{\varphi(q_i)}\), which is known for every sampled value \(q_i\). We then predict the number of ‘hits’ \((J = 1)\) that correspond to the IS pdf \(\varphi\) in a second iteration (but before these samples are actually drawn). We do this by using the assumption that every single sample obtained with simple sampling \((q_i \sim f)\) corresponds to \(\frac{f(q_i)}{\varphi(q_i)}\) samples using IS \((q_i \sim \varphi)\), as shown in (15).

The expected IS variance \(\tilde{\sigma}^2_{is}\) can then be estimated using the first simple sampling iteration, where the first term in the numerator of (21) is multiplied with \(\frac{f(q_i)}{\varphi(q_i)}\). The second term is equal to \(\hat{p}_{N, is}\) and can be approximated by its simple sampling estimate. Rewriting the samples \(\Phi_i\) to \(q_i\) we get
\[
\hat{\sigma}^2_{is} = \frac{1}{N_{is}} \sum_{i=1}^{N_{is}} J^2(q_i) \frac{f^2(q_i)}{\varphi^2(q_i)} - \left( \frac{1}{N_{is}} \sum_{i=1}^{N_{is}} J(q_i) \right)^2 .
\]

We can then estimate \(\lambda_{is}\) from (18). After the first iteration we can then predict \(N_{is}\), as well as form the IS pdf \(\varphi\). We therefore combine importance sampling with the iterative estimation, as discussed next.

D. Combination of AIS and the Binomial Bound

In the first iteration with sample complexity \(N_{ss,1}\) it is recommended to use a grid-based sampling strategy or Latin hypercube [8], in order to get a representative overview of the parameter set \(Q\). The results of this first iteration can then be used to form an IS pdf \(\varphi\), and estimate the IS reduction factor \(\lambda_{is}\). The remaining number of samples to be taken in the second iteration is then \(N_{is,2} = \lambda_{is} (N_{bin,2} - N_{ss,1})\). If desired, these steps can be repeated to obtain a better IS pdf \(\varphi\). We therefore combine Algorithms 2 and 3 as follows.
Algorithm 4 (Iterative estimation with AIS)

Given a desired \( \epsilon, \delta \in (0, 1) \) and a threshold \( \gamma \geq 0 \),

1) Draw \( N_{ss,1} \geq \frac{1}{\delta_1} \ln \frac{2}{\epsilon} \) iid samples, where \( \delta_1 = \delta / \kappa \),
   and \( \epsilon_1 = \kappa \epsilon \), and \( \kappa \) is a suitably chosen real number.
2) Return the empirical probability
   \[
   \hat{p}_{N,1} = \frac{1}{N_{ss,1}} \sum_{i=1}^{N_{ss,1}} J(q_i),
   \]
   and the empirical variance \( \hat{\sigma}_{ss}^2 \). The real \( p \) is at worst
   \( p_{\text{bin}} = \hat{p}_{N,1} + \epsilon_1 \) with confidence \( 1 - \delta_1 \).
3) Determine \( N_{\text{bin},2} \) for \( p_{\text{bin}} \), \( \delta_2 = \delta - \delta_1 \) and \( \epsilon_2 = \epsilon \)
   using (10).
4) Estimate the IS pdf \( \varphi \), based on the kernel density estimate (17) of samples \( q_{bin} \),
   for which \( \rho_j(q) \geq \gamma_j \).
5) Estimate the IS reduction factor \( \hat{\lambda}_{ss} = \frac{\hat{\sigma}_{ss}^2}{\hat{\sigma}_{ss}^2} \), with \( \hat{\sigma}_{ss}^2 \)
   from (22) and \( \hat{\sigma}_{ss}^2 \) from step 3.
6) If \( N_{\text{bin},2} > N_{ss,1} \), then draw \( N_{ss,2} = \hat{\lambda}_{ss} (N_{\text{bin},2} - N_{ss,1}) \) samples.
7) If \( N_{\text{bin},2} > N_{ss,1} \), then do the remaining samples.
8) Return the empirical probability
   \[
   \hat{p}_N = \frac{1}{N_{ss,1} + N_{ss,2}} \left[ \sum_{i=1}^{N_{ss,1}} J(q_i) + \sum_{i=1}^{N_{ss,2}} J(\Phi_i) \varphi(\Phi_i) \right]
   \]
   with accuracy \( \epsilon \) and confidence \( (1-\delta_1)(1-\delta_2) \geq 1-\delta \).

Example 3 (Iterative estimation using AIS)

To illustrate the efficiency of Algorithm 4, Example 1 is again repeated, but now using Algorithm 4. Fig. 5 shows the results for 10,000 simulation sets, each with \( N_j \in [2519, 2810] \), where \( N_{ss,1,j} = 2391 \) and \( N_{ss,2,j} \in [128, 419] \). The sample complexity \( N_{ss,2} \) in the second iteration is considerably small, since the use of IS reduces \( N_{ss,2} \) in the second iteration. The results are \( \hat{p}_N = 0.03628 \) and the variance \( \hat{\sigma}_{ss}^2 = 1.2140 \cdot 10^{-5} \). When the empirical confidence is set \( \delta = 0.01 \) the empirical accuracy is found to be \( \epsilon = 0.0081 \), meaning that \( N_j \in [2519, 2810] \) is still slightly conservative, although much better than in Example 1. The fact that the bound \( N_{\text{bin},2,j} \) is not strict (such that \( \delta = 0.01 \)) is caused by the fact that the factor \( \kappa \) is not chosen optimal, as discussed in Section III. Nevertheless, there is an efficiency improvement of \( \frac{N_{ss,2}}{\max_j (N_{\text{bin},2,j})} = 8.2 \). This means an improvement with respect to Algorithm 2, where the improvement was a factor of 4.8. Obviously the resulting variance is slightly larger than in Example 2, but the accuracy and confidence are still within the desired values. In other words, the bound on the sample complexity is more strict.

V. CONCLUSIONS

We have presented an approach for probabilistic performance validation of advanced driver assistance systems (ADASs), and applied a randomized algorithm, based on adaptive importance sampling, to a simple adaptive cruise control problem. For this problem the new approach leads to roughly a tenfold increase in efficiency.

VI. ACKNOWLEDGMENTS

Research supported by TNO, TRAIL, Transport Research Centre Delft, and the European Network of Excellence HYCON, contract number FP6-IST-511368.

REFERENCES