A Model-Based Approach for Predicting the Remaining Driving Range in Electric Vehicles

Javier A. Oliva[1], Christoph Weihrauch[1], Torsten Bertram[1]

[1] *Institute of Control Theory and Systems Engineering, Technische Universität Dortmund, Germany*
javier.oliva@tu-dortmund.de
christoph.weihrauch@tu-dortmund.de
torsten.bertram@tu-dortmund.de

ABSTRACT

The limited driving range has been pointed out as one of the main technical factors affecting the acceptance of electric vehicles. Offering the driver accurate information about the remaining driving range (RDR) reduces the range anxiety and increases the acceptance of the driver. The integration of electric vehicles into future transportation systems demands advanced driving assistance systems that offer reliable information regarding the RDR. Unfortunately the RDR is, due to many sources of uncertainty, difficult to predict. The driving style, the road conditions or the traffic situation are some of these uncertain factors. A model-based approach for predicting the RDR by combining unscented filtering and Markov chains is introduced in this paper. Detailed models are implemented for representing the electric vehicle and its energy storage system. The RDR prediction is validated by a set of simulation based experiments for different driving scenarios. Whereas traditional approaches consider the RDR as a deterministic quantity, to our knowledge, this approach is the first to represent the RDR by a probability density function. We aim to provide initial steps towards a solution for generating reliable information regarding the RDR which can be used by driving assistance systems in electric vehicles.

1. INTRODUCTION

Electric vehicles have emerged as a promising solution for reducing the oil dependence in transportation systems. Nevertheless, their integration into modern transportation systems is largely limited by the higher cost and the long charging times, on the one hand, and by the low driving range, on the other hand. The limited driving range has been considered as one of the major factors that affect the acceptance of electric vehicles. However, it has been shown (Franke, Neumann, Bühler, Cocron, & Krems, 2012) that reliable information regarding the remaining driving range (RDR) may help to overcome the range anxiety, i.e., the fear that the range of the vehicle is insufficient to reach the desired destination. Unfortunately the RDR is not easy predictable. Many stochastic factors such as the driving style, the traffic situation, the road conditions or the weather largely influence the RDR. It is therefore necessary to take these factors into account in order to meaningfully predict the RDR. To the best of our knowledge, few studies have addressed the RDR prediction. An approach that predicts the driving load of an electric vehicle based on driving pattern identification has been introduced by Yu, Tseng, and McGee (2012). To this aim, a library of identified driving patterns is used. Other approaches address the RDR prediction from a technological point of view. Conradi and Hanssen (2011) introduced an approach that combines a web server, a digital map and a mobile application. The mobile device sends the position of the vehicle and the current state of charge (SOC) of the battery to the web server, which first estimates the energy consumption along all possible routes and then, based on the SOC, calculates the maximum driving range. The main drawback presented in these approaches is that the RDR is treated as a deterministic quantity.

We introduce an approach that predicts the RDR under a stochastic framework. The basic theoretical foundation of this work is based on the work introduced by Daigle and Goebel (2011), where a model-based approach is applied to predict the remaining useful life (RUL) of pneumatic valves. The remainder of this paper is organized as follows. Section 2 introduces the proposed RDR prediction methodology. In section 3 the model of the electric vehicle is presented. Section 4 briefly discusses the algorithm for the state estimation. Section 5 explains the steps needed to predict the RDR. Section 6 presents simulation results for validating the proposed approach. Finally, section 7 concludes the findings of this paper and provides an outlook on the future work.

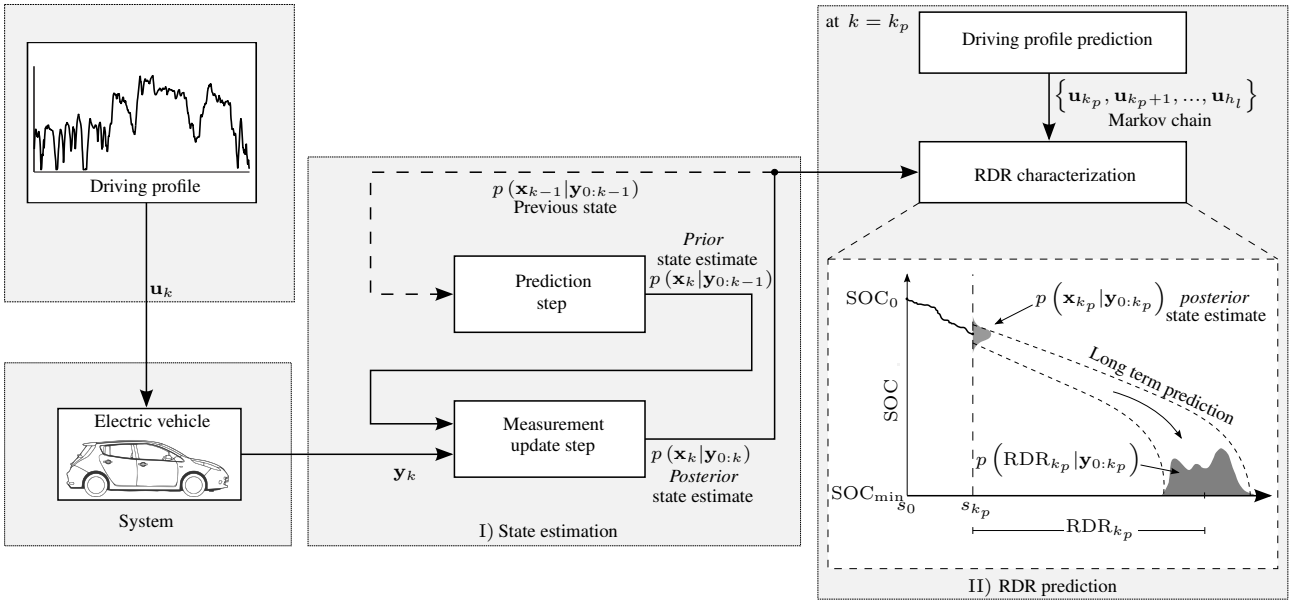

Figure 1. RDR prediction architecture.

2. RDR PREDICTION METHODOLOGY

This section formulates the RDR prediction problem in electric vehicles and introduces the proposed prediction architecture.

2.1. Problem Statement

The RDR is defined as the distance that an electric vehicle can drive, with the energy stored in its battery, before recharging is required. Analogous to the Remaining Useful Life (RUL) calculation problem (Daigle, Saxena, & Goebel, 2012), the RDR prediction problem is formally formulated by considering the electric vehicle as a nonlinear system represented, in a discrete-time form, by

$$\begin{aligned} \mathbf{x}_k &= \mathbf{f}\left(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{v}_k, \mathbf{w}_k\right) \\ \mathbf{y}_k &= \mathbf{h}\left(\mathbf{x}_k, \mathbf{u}_k, \mathbf{n}_k, \mathbf{w}_k\right), \end{aligned} \tag{1}$$

where $\mathbf{x}_k$ is the state vector, $\mathbf{w}_k$ is the parameter vector, $\mathbf{v}_k$ is the process noise vector, $\mathbf{u}_k$ is the input vector, $\mathbf{y}_k$ is the output vector and $\mathbf{n}_k$ is the measurement noise vector. $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$ represent the state and output function respectively. The RDR prediction problem is concerned with predicting the power demand of the electric vehicle, at a given time k_p, and identifying the distance between the position s_{k_p} and the location at which the electric vehicle must be recharged. By defining a threshold in the form

$$T(\cdot) = \begin{cases} 1 \\ 0 \end{cases} \tag{2}$$

it is possible to mathematically determine the recharging point and therefore the RDR. The challenge lies in determining the variables on which $T(\cdot)$ depends. This work considers the battery state of charge (SOC) to be the indicator that determines the threshold condition. Accordingly, the threshold is expressed as $T(\mathrm{SOC})$. Thus, $T(\mathrm{SOC}) = 1$ if $\mathrm{SOC}_{\min}$ is reached and $T(\mathrm{SOC}) = 0$ otherwise. The $\mathrm{SOC}_{\min}$ is usually dictated by the battery management system (BMS) of the electric vehicle in order to protect the battery cells from a possible total charge depletion.

2.2. Prediction Architecture

The RDR is a random variable that is influenced by many sources of uncertainty. This causes the RDR to be difficult to predict. For example, the lack of knowledge about the state variables, such as the SOC, the noise presented in the measurements or the ignorance regarding the future power demand, are some of the factors that largely contribute to the uncertainty of the RDR. Therefore, properly predicting the RDR requires accounting for these sources of uncertainty. To this aim we adopt a model-based methodology, as shown in Fig. 1. The approach proceeds basically in two phases, namely the *state estimation* (I) and the RDR *prediction* (II).

In the first phase, the states are recursively estimated. The *posterior* state estimate is computed in two steps. First, a prediction is made to obtain a *prior* state estimate $p(\mathbf{x}_k|\mathbf{y}_{0:k-1})$. In the second step, as new measurements $\mathbf{y}_k$ become available, the predicted states are updated to compute $p(\mathbf{x}_k|\mathbf{y}_{0:k})$. This estimate establishes the starting point for the RDR prediction phase.

In the second phase, at given time k_p, the RDR is predicted

in two steps. First, future values of the driving profile[1] are predicted by means of Markov chains. In this way the driving profile is generated as a sequence of random variables $\{\mathbf{u}_{k_p}, \mathbf{u}_{k_p+1}, ..., \mathbf{u}_{h_l}\}$ which representatively capture driving patterns that occur in real-world driving situations. The index h_l denotes the horizon length of the driving profile prediction. In the second step, the uncertainty represented by the posterior state estimate $p\left(\mathbf{x}_{k_p}|\mathbf{y}_{0:k_p}\right)$ is propagated through the predicted driving profile until the $\mathrm{SOC}_{\min}$ is reached.

To carry out such an *uncertainty propagation* a sample-based approach in terms of a Monte Carlo simulation is employed. In this approach the probability density function $p\left(\mathbf{x}_{k_p}|\mathbf{y}_{0:k_p}\right)$ is approximated by a set of samples. Each sample is independently propagated through the predicted driving profile until the $\mathrm{SOC}_{\min}$ is reached. Once this happens, the RDR of all samples is identified and used to approximate the posterior $p\left(\mathrm{RDR}_{k_p}|\mathbf{y}_{0:k_p}\right)$.

3. Electric Vehicle Modeling

From a physical standpoint, an electric vehicle can be modeled by a forward-facing (dynamic) or by a backward-facing (quasi-static) approach (Guzzella & Sciarretta, 2005). In the forward-facing approach the vehicle is controlled to follow a desired speed. This approach considers the physical properties of each component of the powertrain and the dynamic interaction between them. Although this approach accurately describes the behavior of the electric vehicle, it requires high computational effort to solve the differential equations of the model.

The backward-facing approach overcomes this issue by assuming that the vehicle reaches the reference speed. With an imposed speed profile, the model calculates the forces acting on the wheels and processes them backwards through the powertrain. The calculation of the power demand depends only on algebraic equations, which decreases the computational burden of the model.

Nevertheless, the battery of the electric vehicle cannot be modeled using this approach since, as already mentioned, the SOC represents the indicator that determines the threshold condition of the prediction algorithm. Since this value is determined in the state estimation step, it cannot be represented by a quasi-static model. For this reason a dynamic model describes the behavior of the battery. The electric vehicle is modeled by combining these two approaches, as shown in Fig. 2.

In the following two sections both parts of the model are explained in detail. For the sake of better understanding, we omit expressing the variables of the quasi-static model as time dependent, since this model is described by a set of algebraic

[1]The driving profile is characterized by the speed (v) and acceleration (a) of the vehicle and by the slope (α) of the road.

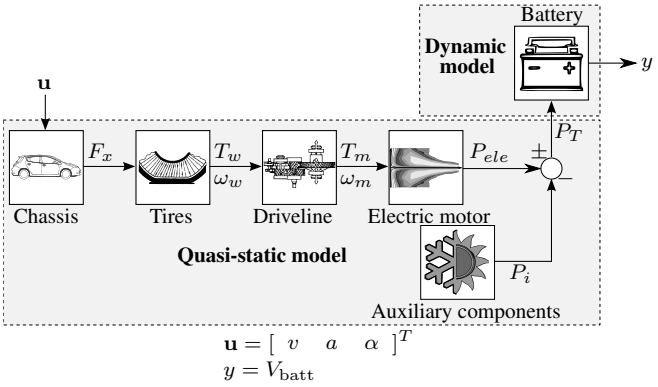

Figure 2. Combined quasi-static/dynamic model of the electric vehicle.

equations. The differential equations of the dynamic model are expressed in a discrete time form, since both, the state estimation and the RDR prediction modules, require a discrete-time representation of the battery model.

3.1. Quasi-static Model

An electric vehicle is composed by many components which, for simplification purposes, can be considered to move uniformly. Thus, the electric vehicle can be represented as a single lumped mass. As shown in Fig. 3, the force F_x required by the vehicle is given by

$$F_x = F_{air} + F_g + F_r + F_i. \tag{3}$$

The forces affecting the motion of the electric vehicle are:

- $F_{air} = \frac{1}{2}\rho_{air} c_w A v^2$ is the aerodynamic drag force,
- $F_g = mg\sin(\alpha)$ is the hill climbing force,
- $F_r = mgK_r$ is the rolling resistance,
- $F_i = ma$ is the force needed to accelerate/decelerate the electric vehicle,

where ρ_{air} is the density of air, c_w is the aerodynamic drag coefficient, A and m are the frontal area and the mass of the vehicle, g is the gravitational acceleration, K_r is the rolling resistance coefficient, α is the inclination (slope) of the road segment and v is the speed of the vehicle.

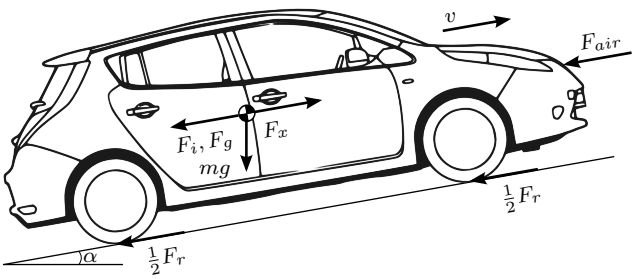

Figure 3. Forces acting during the motion of a vehicle.

The mechanical power P_{mec} demanded by the electric motor is easily calculated by means of a polynomial power requirement model (Kim, Lee, & Shin, 2013) as follows

$$P_{mec} = F_x v = \frac{1}{2}\rho_{air} c_w A v^3 + mg\sin(\alpha)\, v + \\ + mgK_r v + mav. \quad (4)$$

This model accurately calculates the mechanical power demand of a vehicle with a very low computational cost. The electrical power demand P_{ele} of the electric motor is then computed by

$$P_{ele} = \frac{P_{mec}}{\eta(\omega_m, T_m)}, \quad (5)$$

where η represents the electric motor's efficiency, $\omega_m = \frac{v i_d}{r_{tire}}$ is the rotational speed of the rotor and $T_m = \frac{F_x r_{tire}}{i_d}$ is the torque demand of the motor. Here r_{tire} and i_d are the tire's radius and the gear ratio of the driveline respectively.

Modern electric vehicles are able to recover a certain amount of the kinetic energy by means of regenerative braking. Such systems operate the electric motor in generator mode for delivering back the recovered energy to the battery. It is worth mentioning that η depends on whether the electric drive operates in motor or in generator mode. Accordingly,

$$\eta = \begin{cases} \eta_m(\omega_m, T_m) \leq 1, & \text{motor mode} \\ \eta_g(\omega_m, T_m) > 1, & \text{generator mode.} \end{cases} \quad (6)$$

As shown in Fig. 2, the total electrical power P_T of the electric vehicle is composed of the electrical power demanded by the electric motor P_{ele} and by the sum of the power P_i consumed by each of the auxiliary components

$$P_T = P_{ele} + \sum_{i=1}^{n} P_i. \quad (7)$$

For the sake of simplicity, the power consumed by each of the auxiliary components is assumed to be constant.

3.1.1. Input Variables of the Quasi-static Model

To properly employ Eq.(4) in the RDR prediction algorithm, we need to differentiate between input variables and parameters. The input variables of the quasi-static model must be easily acquirable and should be highly dynamic, so that they cannot be considered as constant. Table 1 summarizes the dynamics and the dependency of the quasi-static model parameters. The parameters g and ρ_{air}, even though they can be easily determined, depend on the altitude and rarely change drastically during a trip. Also m, c_w and A are easily observed. They also change slowly since they depend on the vehicle design. The friction coefficient K_r, despite its high dynamic, cannot be easily determined. For this reason it is considered as a constant under the assumption that the road conditions do not change during the trip.

Table 1. Dynamics and dependency of the quasi-static model parameters.

Parameter	Dynamics	Dependency
a $(\mathrm{m/s^2})$	Very high	Driver, road, traffic
v $(\mathrm{m/s})$	High	Driver, road, traffic
m (kg)	Very low	Vehicle design
g $(\mathrm{m/s^2})$	Very low	Altitude
K_r	High	Road
α $(^\circ)$	High	Road
ρ_{air} $(\mathrm{kg/m^3})$	Low	Altitude
c_w	Very low	Vehicle design
A $(\mathrm{m^2})$	Very low	Vehicle design

Our approach considers a, v and α as the input variables for the quasi-static model, since they meet the requirements previously mentioned. Accordingly, the input vector is given by

$$\mathbf{u} = \begin{bmatrix} v & a & \alpha \end{bmatrix}^T. \quad (8)$$

3.2. Battery Model

Our approach employs the model of a Li-ion cell shown in Fig. 4. The model combines the Kinetic Battery Model (KiBaM) (Manwell & McGowan, 1994) for capturing the nonlinear effects in the battery capacity, such as the recovery and the rate capacity effect, with a second order equivalent circuit based model which captures the dynamic response of the Li-ion cell. Furthermore, the combined model demands low computational effort, which makes it suitable for real-time applications.

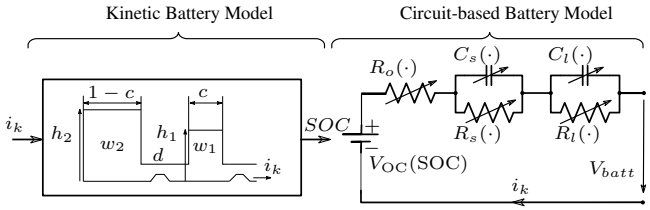

Figure 4. Combined battery model.

Even though the KiBaM was initially developed for lead acid batteries, it has been shown to be suitable for modeling the capacity behavior of Li-ion cells (Jongerden & Haverkort, 2009).

The Kinetic Battery Model abstracts the chemical processes of the battery discharge to its kinetic properties. The model assumes that the total charge of the battery is distributed with a capacity ratio $0 < c < 1$ between two charge wells. The first well contains the available charge and delivers it directly to the load. The second well supplies charge only to the first well by means of the parameter d. The rate of charge that flows from the second to the first well depends on both d and

on the height difference between the wells $(h_2 - h_1)$. If the first well is empty, then the battery is considered to be fully discharged.

By applying load to the battery, the charge in the first well is reduced, which leads to an increment in the height difference between both wells. After removing the load, certain amount of charge flows from the second well to the first well until the height of both wells is the same. In this way the recovery effect is taken into account by the model. The rate capacity effect is also considered in this model. For high discharge currents, the charge in the first well is delivered faster to the load in comparison to the charge that flows from the second well. In this scenario there is an amount of charge that remains unused. The consideration of this effect is especially important for applications in electric vehicles, since the unused charge might eventually increase the driving range.

The KiBaM yields two difference equations which describe the change of capacity in both wells in dependence of the load i_k, the conductance d and the capacity ratio c:

$$w_{1,k+1} = a_1 w_{1,k} + a_2 w_{2,k} + b_1 i_k, \tag{9}$$

$$w_{2,k+1} = a_3 w_{1,k} + a_4 w_{2,k} + b_2 i_k, \tag{10}$$

where

$$\begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} = \mathrm{e}^{\begin{pmatrix} -\frac{d}{c} & \frac{d}{1-c} \\ \frac{d}{c} & -\frac{d}{1-c} \end{pmatrix} \Delta t}$$

$$\begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \int_0^{\Delta t} \mathrm{e}^{\begin{pmatrix} -\frac{d}{c} & \frac{d}{1-c} \\ \frac{d}{c} & -\frac{d}{1-c} \end{pmatrix} \vartheta} \mathrm{d}\vartheta \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

The term Δt is the sampling time used in the discretization of the model. The battery SOC is given by

$$\mathrm{SOC}_k = \frac{w_{1,k}}{cC_n 3600}, \tag{11}$$

where C_n is the nominal capacity of the battery. The right-hand-side equivalent circuit of Fig. 4 is compounded of three parts, namely, the open circuit voltage V_{OC}, a resistance R_o and two RC networks.

The voltage V_{OC} changes at different SOC levels, and is given by the following empirical equation

$$\begin{aligned} V_{\mathrm{OC}}(\mathrm{SOC}) \quad = \quad & a_1 - \frac{a_2}{\mathrm{SOC}} - a_3 \mathrm{SOC} + a_4 \ln(\mathrm{SOC}) + \\ & + a_5 \mathrm{e}^{(\frac{-a_6}{1-\mathrm{SOC}})} + a_7 \mathrm{SOC}^2 + a_8 \mathrm{SOC}^4 + \\ & + a_9 \ln(1 - \mathrm{SOC}) + a_{10} \sin(a_{11} \mathrm{SOC}). \end{aligned} \tag{12}$$

The ohmic resistance R_o captures the I-R drop, i.e., the instantaneous voltage drop due to a step load current event. The $R_s C_s$ and $R_l C_l$ networks capture the voltage drops due to the electrochemical and the concentration polarization, respectively. In Fig. 4 the dependency of these parameters on the temperature and on the SOC is represented by the term $(\cdot)$. This part of the model yields two difference equations which describe the transient response of the battery:

$$v_{s,k+1} = \mathrm{e}^{-\frac{\Delta t}{R_s C_s}} v_{s,k} + \left(-R_s \mathrm{e}^{-\frac{\Delta t}{R_s C_s}} + R_s \right) i_k, \tag{13}$$

$$v_{l,k+1} = \mathrm{e}^{-\frac{\Delta t}{R_l C_l}} v_{l,k} + \left(-R_l \mathrm{e}^{-\frac{\Delta t}{R_l C_l}} + R_l \right) i_k. \tag{14}$$

Accordingly, the state vector of the battery model is given by

$$\mathbf{x}_k = \begin{bmatrix} w_{1,k} & w_{2,k} & v_{s,k} & v_{l,k} \end{bmatrix}^T. \tag{15}$$

The output y_k of the system, represented by the terminal voltage $V_{batt,k}$, is then computed as follows

$$y_k = V_{batt,k}(\mathrm{SOC}) = V_{\mathrm{OC}}(\mathrm{SOC}) + R_o i_k + v_{l,k} + v_{s,k}. \tag{16}$$

As presented in the previous section, the quasi-static model computes the total electrical power demand P_T. Nevertheless, the battery model requires the load current i_k as the input variable. Therefore, it is necessary to express i_k in terms of P_T. The load current i_k can be obtained from the definition of electrical power $P = IV$. Considering $P = P_T$ and $V = V_{batt}$ the terminal voltage can be expressed as

$$V_{batt} = \frac{P_T}{i}. \tag{17}$$

By substituting Eq. (17) into Eq. (16) and solving for i, the current at time k is given by

$$i_k = -\frac{C - \sqrt{C^2 - 4 P_T(\mathbf{u}_k) R_o}}{2 R_o}, \tag{18}$$

with

$$C = \left(V_{\mathrm{OC}}(\mathrm{SOC}) + v_{s,k} + v_{l,k} \right).$$

$P_T(\mathbf{u}_k)$ expresses the dependence of the total electrical power demand on the input vector given by Eq. (8). The solution with the positive part in the square root term of Eq. (18) is neglected, since its consideration would cause some current to be supplied by the battery when $P_T = 0$, which in practice is not possible.

4. State Estimation

At the beginning of the estimation and prediction steps the system states, and especially the initial SOC, are unknown. To obtain an accurate prediction of the RDR, the prediction module needs an initial starting point that is as accurate as possible. For that reason the state estimation has to converge to the true value before the prediction is carried out. In the prediction framework shown in Fig. 1 the task of the estimation step is to compute $p(\mathbf{x}_k | \mathbf{y}_{0:k})$, i.e., to represent the most up-to-date knowledge of the state variables at given time k

based on the history of measurements of the system.

For state estimation in nonlinear systems Bayesian tracking algorithms such as the particle filter (PF) (Rigatos, 2009), the extended Kalman filter (EKF) or the unscented Kalman filter (UKF) (Julier & Uhlmann, 2004) are the most established ones. This study uses the UKF as suggested by Daigle et al. (2012), because of the smaller number of sampling points and reduced computational complexity compared to the PF. The next section briefly describes the framework of the unscented Kalman filter that was implemented in this work.

4.1. Unscented Kalman Filter

The UKF applies the so-called Unscented Transform (UT) to approximate the distribution of the state variables. The UT considers each variable x as a random variable with mean $\bar{\mathbf{x}}$ and covariance $\mathbf{P}_x$ and computes the mean $\bar{\mathbf{y}}$ and covariance $\mathbf{P}_y$ of the output variable. This computation is carried out by choosing a set of deterministically weighted points $\mathbf{S}_i = \{w_i, \boldsymbol{\mathcal{X}}_i\}$, which are sampled from the distribution of x and are known as sigma points (Julier & Uhlmann, 2004). The sigma points are then propagated through a nonlinear function $\boldsymbol{\mathcal{Y}}_i = \mathbf{g}(\boldsymbol{\mathcal{X}}_i)$ that relates both the sigma points and the transformed sigma points. The posterior mean and covariance of the output variable can be recovered by

$$\bar{\mathbf{y}} \approx \sum_{i=0}^{2L} w_i \boldsymbol{\mathcal{Y}}_i \tag{19}$$

$$\mathbf{P}_y \approx \sum_{i=0}^{2L} w_i \left(\boldsymbol{\mathcal{Y}}_i - \bar{\mathbf{y}}\right)\left(\boldsymbol{\mathcal{Y}}_i - \bar{\mathbf{y}}\right)^T. \tag{20}$$

Many methods have been developed for selecting sigma points and these methods along with the choice of their parameters play an important role for the accuracy of the state estimation (Daigle & Goebel, 2010). In this study, the symmetric unscented transform is used. Here the set of 2L+1 sigma points are selected as:

$$\begin{aligned} \boldsymbol{\mathcal{X}}_0 &= \bar{\mathbf{x}} && i = 0 \\ \boldsymbol{\mathcal{X}}_i &= \bar{\mathbf{x}} + \left(\gamma\sqrt{(L+\lambda)\,\mathbf{P}_x}\right)_i && i = 1, ..., L \\ \boldsymbol{\mathcal{X}}_i &= \bar{\mathbf{x}} - \left(\gamma\sqrt{(L+\lambda)\,\mathbf{P}_x}\right)_i && i = L+1, ..., 2L, \end{aligned} \tag{21}$$

with the weights given by

$$\begin{aligned} w_0^{(m)} &= \tfrac{\lambda}{L+\lambda} && i = 0 \\ w_0^{(c)} &= \tfrac{\lambda}{L+\lambda} + (1 - \alpha^2 + \beta) && i = 0 \\ w_i^{(c)} &= \tfrac{\lambda}{2(L+\lambda)} && i = 1, ..., 2L, \end{aligned} \tag{22}$$

where L refers to the number of states in the state vector and $\left(\sqrt{(L+\lambda)\,\mathbf{P}_x}\right)_i$ is the i^{th} column of the square root of the weighted covariance matrix. The parameters λ, α, β and γ serve for scaling the sigma points in the state space and are chosen according to the (heuristic) recommendations of Julier and Uhlmann (2004). The algorithm 1 summarizes the main steps for state estimation using the unscented Kalman Filter.

Algorithm 1 Unscented Kalman Filter for State Estimation

Initialize:
$\hat{\mathbf{x}}_0 = E\left[\mathbf{x}_0\right], \mathbf{P}_0 = E\left[\left(\mathbf{x}_0 - \hat{\mathbf{x}}_0\right)\left(\mathbf{x}_0 - \hat{\mathbf{x}}_0\right)^T\right]$
For $k = 1..., \infty$
1. Calcualte sigma points:
$\boldsymbol{\mathcal{X}}_{k-1} = \left[\begin{array}{ccc} \hat{\mathbf{x}}_{k-1} & \hat{\mathbf{x}}_{k-1} \pm \gamma\sqrt{\mathbf{P}_{x_{k-1}}} \end{array}\right]$
2. State prediction:
a. Propagate the sigma points through the system model:
$\boldsymbol{\mathcal{X}}_{k|k-1} = \mathbf{f}\left(\boldsymbol{\mathcal{X}}_{k-1}, \mathbf{u}_{k-1}\right)$
b. Calculate the propagated mean and covariance:
$\hat{\mathbf{x}}_k^- = \sum_{i=0}^{2L} w_i^{(m)} \boldsymbol{\mathcal{X}}_{i,k|k-1}$
$\mathbf{P}_{x_k}^- = \sum_{i=0}^{2L} w_i^{(c)} \left(\boldsymbol{\mathcal{X}}_{i,k|k-1} - \hat{\mathbf{x}}_k^-\right)\left(\boldsymbol{\mathcal{X}}_{i,k|k-1} - \hat{\mathbf{x}}_k^-\right)^T + \mathbf{R}_v$
c. Sigma point propagation through the output model:
$\boldsymbol{\mathcal{Y}}_{k|k-1} = \mathbf{h}\left(\boldsymbol{\mathcal{X}}_{k-1}\right)$
d. Calculate the propagated mean:
$\hat{\mathbf{y}}_k^- = \sum_{i=0}^{2L} w_i^{(m)} \boldsymbol{\mathcal{Y}}_{i,k|k-1}$
3. Measurement update:
a. Calculate the estimated covariance:
$\mathbf{P}_{y_k} = \sum_{i=0}^{2L} w_i^{(c)} \left(\boldsymbol{\mathcal{Y}}_{i,k|k-1} - \hat{\mathbf{y}}_k^-\right)\left(\boldsymbol{\mathcal{Y}}_{i,k|k-1} - \hat{\mathbf{y}}_k^-\right)^T + \mathbf{R}_n$
$\mathbf{P}_{x_k y_k} = \sum_{i=0}^{2L} w_i^{(c)} \left(\boldsymbol{\mathcal{X}}_{i,k|k-1} - \hat{\mathbf{x}}_k^-\right)\left(\boldsymbol{\mathcal{Y}}_{i,k|k-1} - \hat{\mathbf{y}}_k^-\right)^T$
b. Calculate the Kalman gain $\mathbf{K}$:
$\mathbf{K}_k = \mathbf{P}_{x_k y_k}\mathbf{P}_{y_k}^{-1}$
c. Update the state estimation and covariance:
$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k\left(\mathbf{y}_k - \hat{\mathbf{y}}_k^-\right)$
$\mathbf{P}_{x_k} = \mathbf{P}_{x_k}^- - \mathbf{K}_k\mathbf{P}_{y_k}\mathbf{K}_k^T$

The UKF estimate of $\mathbf{x}_k = \left[\begin{array}{cccc} w_{1,k} & w_{2,k} & v_{s,k} & v_{l,k} \end{array}\right]^T$ is used to calculate the output $y_k = V_{\mathrm{batt},k}(\mathrm{SOC})$ by using Eq. (16). The output voltage y_k depends on the SOC which is not part of the state vector. Therefore the SOC has to be calculated from the states in the algebraic Eq. (11). As further explained in the following section, if a RDR prediction is desired at given time k_p, the prediction module uses the last estimation $\hat{\mathbf{x}}_{k_p}$, from which a new set of sigma points is generated. The set of new sigma points is then used as the initial condition for the prediction step.

5. RDR PREDICTION

A prediction starts at given time k_p. Here, the posterior estimate $p\left(\mathbf{x}_{kp}|\mathbf{y}_{0:kp}\right)$ serves as the starting point for the prediction. As already mentioned in section 2, a sample-based approach for predicting the RDR is used. Therefore, the set of sigma points $\mathbf{S}_{kp}^i = \{\boldsymbol{\mathcal{X}}_{kp}^i, w_{kp}^i\}$, calculated by the UKF

in the state estimation step, is used and propagated forward in time through simulation. All sigma points are propagated independently by following the predicted driving profile until $T_{kp}^{i} = 1$, i.e., until each sigma point reaches the $\mathrm{SOC}_{\min}$. The posterior mean $\overline{\mathrm{RDR}}_{k_p}$ and its covariance $\mathbf{P}_{\mathrm{RDR}_{k_p}}$ can be recovered, in a similar fashion as in Eq. (19) and Eq. (20), by

$$\overline{\mathrm{RDR}}_{k_p} \approx \sum_{i=0}^{2L} w_{k_p}^{i} \mathrm{RDR}_{k_p}^{i}, \tag{23}$$

$$\mathbf{P}_{\mathrm{RDR}_{k_p}} \approx \sum_{i=0}^{2L} w_{k_p}^{i} (\mathrm{RDR}_{k_p}^{i} - \overline{\mathrm{RDR}}_{k_p})(\mathrm{RDR}_{k_p}^{i} - \overline{\mathrm{RDR}}_{k_p})^{T} \tag{24}$$

As stated above, the propagation of all sigma points requires an hypothesized future driving profile. This work employs a stochastic approach based on Markov chains to predict the driving profile. The predictions are generated in such a way, that characteristic driving patterns of real-world driving situations are captured.

5.1. Driving Profile Prediction

Driving profiles can be modeled as a discrete-time Markov chain (T. Lee & Filipi, 2011). In this work two Markov chains are used. First, future values of speed and acceleration are generated by a 2D chain. Second, the slope profile is predicted by means of a 1D Markov chain independent of the speed and the acceleration. To apply a Markov chain the input space is quantized, for the speed/acceleration pair and for the slope, in such a way that each input variable takes a finite number of values. The inputs are given by $\{\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_{h_l}\}$, with $\mathbf{u}_k = \begin{bmatrix} \mathbf{u}_k^{va} & u_k^{\alpha} \end{bmatrix}^T$. Here $\mathbf{u}_k^{av} = \begin{bmatrix} v_k & a_k \end{bmatrix}$ and $u_k^{\alpha} = \alpha_k$ represent parts of the input space given by the speed/acceleration pair and by the slope respectively, with h_l as the horizon length of the predicted profiles.

The Markov chain assumes that the transition probability from $\mathbf{u}_k$ to $\mathbf{u}_{k+1}$ only depends on the current state and not on the history of previous states.

The transition probabilities among the states are grouped in a transition probability matrix (TPM) $\mathbf{\Phi}$ such that

$$p_{ij} = \mathbf{\Phi}\left(\mathbf{u}_{k+1} = j | \mathbf{u}_k = i\right), \tag{25}$$

where p_{ij} is the ij^{th} element of $\mathbf{\Phi}$.

In this paper two transition probability matrices are used, namely $\mathbf{\Phi}^{va}$ and $\mathbf{\Phi}^{\alpha}$. The transition probabilities of $\mathbf{\Phi}^{va}$ are estimated from historical driving data and from standard driving cycles. The resolution of $\mathbf{\Phi}^{va}$ for the speed is 1 km/h in the interval $[0, 130]\mathrm{km/h}$ and for the acceleration it is 0.2 $\mathrm{m/s^2}$ in the interval $[-3, 3]\mathrm{m/s^2}$. $\mathbf{\Phi}^{\alpha}$ is estimated from real road height profiles and has a resolution of 0.5° in the interval $[-10, 10]^{\circ}$. These resolutions offer a good trade-off between computational effort and accuracy in the prediction.

For estimating both TPMs the maximum likelihood estimation method (T. C. Lee, Judge, & Zellner, 1970) is applied. The transition probability p_{ij} is computed by

$$p_{ij} = \frac{n_{ij}}{\sum_{j=1}^{s} n_{ij}} = \frac{n_{ij}}{n_i}, \tag{26}$$

where n_{ij} is the number of times a transition from $\mathbf{u}_i$ to $\mathbf{u}_j$ has occurred, and n_i is the total number of occurrences of $\mathbf{u}_i$. Algorithm 2 summarizes the steps required for the prediction of a driving profile.

Algorithm 2 Driving Profile Prediction

Require: $\mathbf{\Phi}^{va}, \mathbf{\Phi}^{\alpha}, v_{k_p}, a_{k_p}, \alpha_{k_p}, h_l$
Ensure: $\{\mathbf{u}_k, \mathbf{u}_{k+1}, ..., \mathbf{u}_{h_l}\}$
  Initialize:
  $i \leftarrow 0$
  $k \leftarrow k_p$
  $v_i \leftarrow v_{k_p}$, $a_i \leftarrow a_{k_p}$, $\alpha_i \leftarrow \alpha_{k_p}$
  for $l = 1$ **to** h_l **do**
    Randomly draw $\mathbf{u}_j^{av} = \begin{bmatrix} v_j & a_j \end{bmatrix}$ for the next state according to $\mathbf{\Phi}^{va}\left(\mathbf{u}_{k+1}^{va} = \mathbf{u}_j^{va} | \mathbf{u}_k^{va} = \mathbf{u}_i^{va}\right)$
    $v_{k+1} \leftarrow v_j$
    $a_{k+1} \leftarrow a_j$
    $\mathbf{u}_{k+1}^{av} \leftarrow \begin{bmatrix} v_{k+1} & a_{k+1} \end{bmatrix}$
    Randomly draw $u_j^{\alpha} = \alpha_j$ for the next state according to $\mathbf{\Phi}^{\alpha}\left(\mathbf{u}_{k+1}^{\alpha} = u_j^{\alpha} | \mathbf{u}_k^{\alpha} = u_i^{\alpha}\right)$
    $\alpha_{k+1} \leftarrow \alpha_j$
    $u_{k+1}^{\alpha} \leftarrow \alpha_{k+1}$
    Create the driving profile
    $\mathbf{u}_{k+1} \leftarrow \begin{bmatrix} \mathbf{u}_{k+1}^{av} & u_{k+1}^{\alpha} \end{bmatrix}^T$
    $i \leftarrow j$
    $k \leftarrow k + 1$
  end for

The prediction is initialized with the current values v_{k_p}, a_{k_p} and α_{k_p}. Both Markov chains are generated separately in an iterative manner, until the desired length h_l of the profile is reached.

5.2. RDR Characterization

Until now the prediction of the RDR, as computed by Eq. (23), requires propagating the set of sigma points through a single predicted driving profile. Nevertheless, this procedure only accounts for the uncertainty related to the state estimation. Therefore, the uncertainty presented in the driving profile prediction is not taken into account. This issue is approached by generating multiple hypotheses about the future driving profile and by propagating the set of sigma points through each of them. In this way the uncertainty of the driving profile is accounted making the prediction of the RDR more meaningful.

After all sigma points, along all predicted driving profiles, reach the $\mathrm{SOC}_{\min}$, i.e., $T_{kp}^{ij} = 1$, the posterior $p\left(\mathrm{RDR}_{k_p}|\mathbf{y}_{0:k_p}\right)$ is approximated by means of the kernel density estimation (Bowman & Azzalini, 1997)

$$p\left(\mathrm{RDR}_{k_p}|\mathbf{y}_{0:k_p}\right) \approx \frac{1}{N_u h}\sum_{j=1}^{N_u} K\left(\frac{\mathrm{RDR}_{k_p} - \overline{\mathrm{RDR}}_{k_p}^{j}}{h}\right), \tag{27}$$

where N_u is the number of predicted driving profiles, $K(\cdot)$ is the kernel, which in this work is Gaussian, and h is a smoothing factor known as bandwidth. It must be noted that the estimate kernel density estimate is based only on the set of mean posteriors $\left\{\overline{\mathrm{RDR}}_{k_p}^{j}\right\}_{j=1}^{N_u}$ calculated with Eq. (23), as shown in Fig. 5. This implies that the covariances $\left\{\mathbf{P}_{\mathrm{RDR}_{k_p}}^{j}\right\}_{j=1}^{N_u}$ are not taken into account. In our case this is justified since the uncertainty added by driving profile prediction is considered to be large in comparison to the uncertainty related to the state estimation. For the sake of better understanding, Fig. 5 shows the propagation of only one sigma point through all predicted driving profiles.

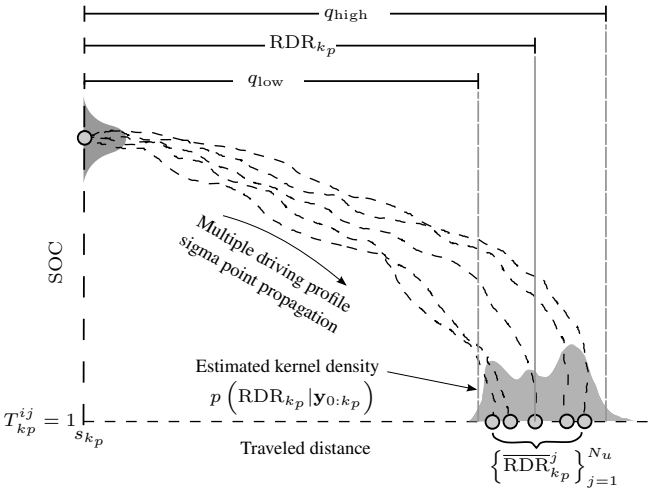

Figure 5. Propagation of one sigma point through multiple driving profiles.

Since $p\left(\mathrm{RDR}_{k_p}|\mathbf{y}_{0:k_p}\right)$ is usually non Gaussian, we rely on the median for estimating the RDR and on quantiles, here represented by q_{low} and q_{high}, as the measure of spread (Hoaglin, Moesteller, & Tukey, 1983).

6. Results and Discussions

This section demonstrates and validates the proposed approach for predicting the RDR. The performance and the accuracy of the proposed approach for different driving scenarios, namely the city, rural areas and the highway, is investigated. Accordingly, the standard driving cycles shown in Fig. 6 are used in the simulations.

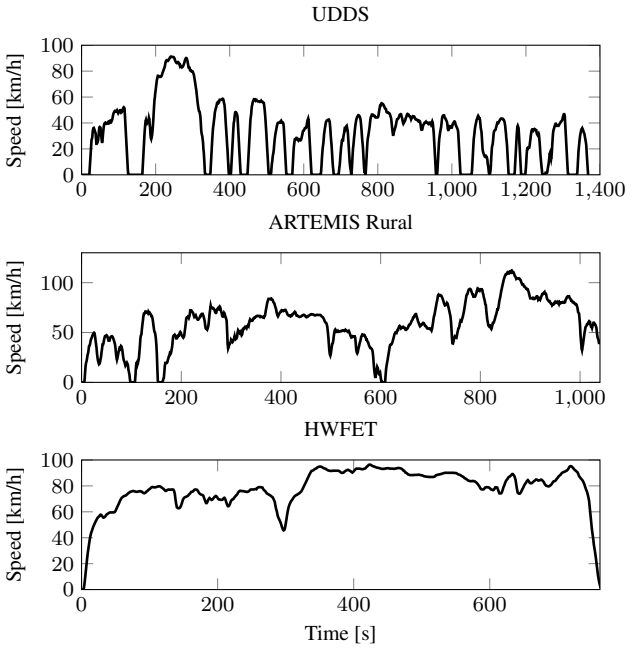

Figure 6. Standard drive cycles used in the simulation.

The parameters of both, the quasi-static and the battery model, are presented in table 2. The parameter of the quasi-static model are obtained from manufacturer's data sheets of the Nissan Leaf (Hayes, Oliveira, Vaughan, & Egan, 2011). The rolling resistance coefficient K_r is chosen to represent a dry road. It must be noted that the parameters of the battery model, correspond to those of one cell. These parameters are identified with the help of experimental data from a Li-ion cell. Since the parameters of the equivalent-circuit based model depend on both, the temperature and the battery SOC, they can not be considered as constant. Therefore, lookup tables are used to store them. To properly simulate the behavior of the entire battery pack, the cell capacity and the nominal voltage are scaled up to 24 kWh and 403.2 V respectively. For all experiments a temperature of 25 °C is assumed.

Table 2. Parameters of the quasi-static and the battery model.

Quasi-static Model		Battery Model	
Parameter	Value	Parameter	Value
A	2.29 m^2	C_n	2.15 Ah
c_w	0.28	V_{nom}	4.2 V
m	1520 kg	V_{lim}	2.8 V
K_r	0.7	d	1.4×10^{-5}
$T_{m,\max}$	280 Nm	c	0.96
$P_{ele,\max}$	80 kW		
r_{tire}	0.3 m		
ρ_{air}	1.226 $\mathrm{kg/m}^3$		
g	9.81 $\mathrm{m/s}^2$		

Each experiment is carried out as follows. First, for each driving scenario a sufficiently large drive cycle is created, so that the electric vehicle can drive until it reaches the $\mathrm{SOC}_{\min}$. During the simulation both the current and the terminal voltage of the battery are measured, which are then used by the UKF for estimating the states of the battery. The RDR is predicted every 1000 seconds for the city scenario and every 500 seconds for both, the simulation in rural areas and on the highway. The reason for this is that the time needed for simulating each scenario largely depends on the speed range of the drive cycle. The higher the speed of the drive cycle, the shorter the time it takes to finish the simulation and therefore the lower the amount of RDR predictions that can be done.

For the sake of demonstration, Figs. 7, 8 and 9 depict the RDR prediction process at prediction time $k_p = 2$, i.e., after 1000 seconds for the city scenario and after 500 seconds for the rural and highway scenarios. In all cases, the initial battery SOC is approximately 0.9. Multiple simulations are carried out simultaneously based on the predicted driving profiles. We have set $\mathrm{SOC}_{min} = 0.1$ for all experiments. As it can be seen, the SOC curves evolve tighter during the simulation of the highway scenario than in the first two scenarios. This causes the uncertainty related to the predicted driving profiles to be lower and therefore the estimated RDR kernel density is narrower.

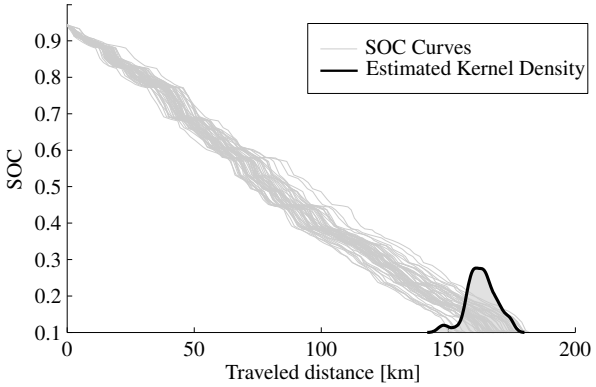

Figure 7. Predicted RDR represented by a kernel density in the city scenario.

6.1. Performance of the RDR Prediction

To evaluate the performance of the RDR prediction, the relative accuracy (RA) and the alpha-lambda ($\alpha - \lambda$) metrics are employed (Saexena, Celaya, Saha, Saha, & Goebel, 2009).

The relative accuracy measures the error in the RDR prediction relative to the true RDR. The RA is given by

$$\mathrm{RA}_{k_p} = 100\left(1 - \frac{\left|\mathrm{RDR}^*_{k_p} - \mathrm{RDR}_{k_p}\right|}{\mathrm{RDR}^*_{k_p}}\right), \qquad (28)$$

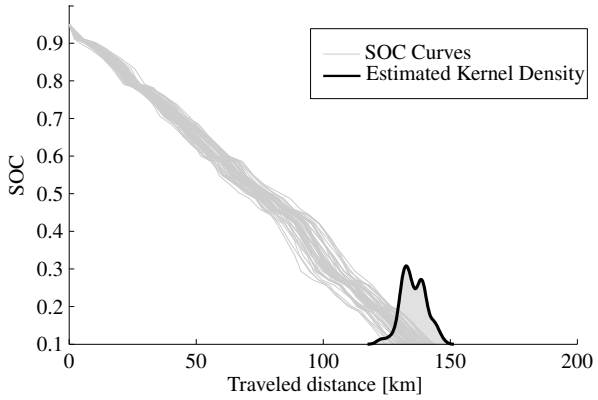

Figure 8. Predicted RDR represented by a kernel density in the rural scenario.

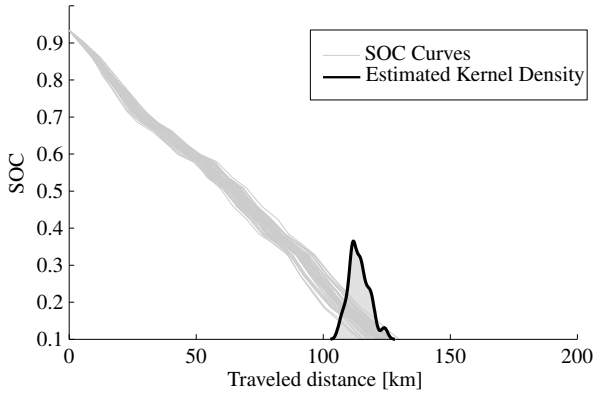

Figure 9. Predicted RDR represented by a kernel density in the highway scenario.

where $\mathrm{RDR}^*_{k_p}$ represents the ground truth RDR at time k_p and RDR_{k_p} is the predicted RDR at that time. The $\alpha-\lambda$ metric serves to evaluate whether the predicted RDR's lie withing specified bounds. These bounds are usually calculated as a fraction of the true RDR. We set a value $\alpha = 0.15$. Table 3 summarizes the calculated RA for each driving cycle at different prediction times.

As it can be seen in Figs. 10, 11 and 12, the proposed approach performs similar for each driving scenario. The simulation for the city scenario was done based on the UDDS drive cycle. The predicted RDR falls near the true RDR for the entire simulation. Also quantiles q_5 and q_{95} fall within the bounds of the α metric for most part of the simulation. The second scenario, namely the rural areas, is simulated base on the ARTEMIS rural drive cycle (Andre, 2004). The results also show an acceptable accuracy. Nevertheless, in this case the RDR is overestimated for most part of the simulation and the area between q_5 and q_{95} doesn't fall withing the bounds. The reason for this is, that the transition probability matrix used for the prediction of the driving profiles in this scenario combines information of both, the city and the highway. For this reason, the predicted profiles don't always properly

Table 3. RDR prediction performance.

k_p	RA_{UDDS}	RA_{RURAL}	RA_{HWFET}
1	96.67	88.90	98.39
3	99.17	90.81	97.01
5	99.48	91.89	97.59
7	98.64	92.22	92.65
9	98.69	91.63	96.25
11	96.48	92.40	86.01
13	98.11	99.70	42.68
15	85.47	97.41	−−
17	91.92	53.47	−−
19	73.17	−−	−−
21	90.56	−−	−−

represent this driving scenario. The third scenario is represented by the HWFET driving cycle. As it can be observed in Fig. 12, the predicted RDR's during the simulation are very close to the true values. Moreover, the uncertainty band given by the quantiles lies very tightly withing the α bounds. The deviation presented towards the end for both the rural and the highway scenario is due to the acceleration phase at the beginning of each prediction, since it always starts from the standing position. This makes the predicted energy consumption at the beginning of the prediction to be larger in comparison to the real one, where such acceleration phase is not present.

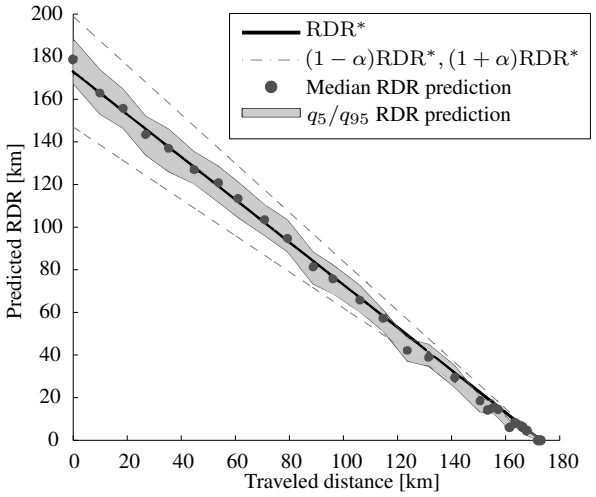

Figure 10. RDR predictions for the UDDS drive cycle.

7. Conclusions and Future Work

This work presents a model-based approach for predicting the RDR in electric vehicles. The proposed approach proceeds in two steps, namely the state estimation and the prediction step. Detailed models for determining the power demand of electric vehicles and for describing the dynamic behavior of the battery are also presented. Our approach takes into account the sources of uncertainty that influence the RDR. First, a Bayesian tracking algorithm, namely the unscented Kalman filter was implemented to estimate the SOC of the battery. In the second step, this estimate is used as the starting point

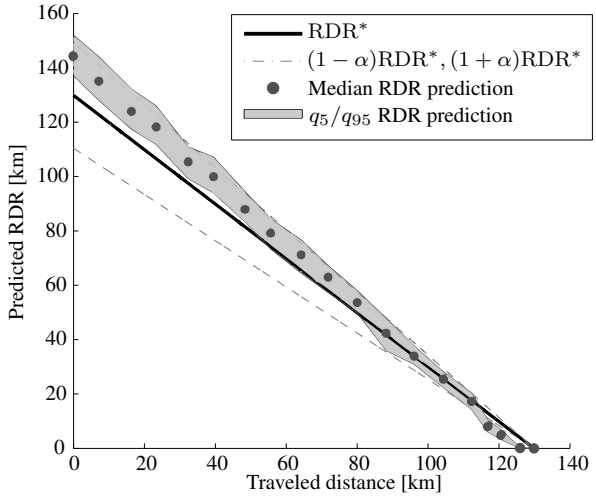

Figure 11. RDR predictions for the ARTEMIS rural drive cycle.

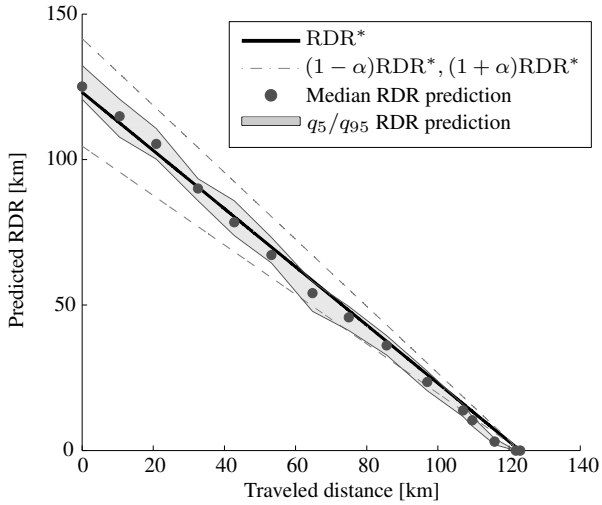

Figure 12. RDR predictions for the HWFET drive cycle.

for the RDR prediction. Here, the set of sigma points, obtained from the state estimation, is propagated forward in time by letting them follow multiple predicted driving profiles. A stochastic approach based on Markov chains for predicting the driving profiles is employed. The RDR is then computed as a probability density function approximated by the distribution of the propagated sigma points. The proposed approach is demonstrated and validated by means of a series of simulation experiments. The experiments allowed to predict the RDR under different driving situations. The obtained results have shown that the RDR can be accurately predicted for given scenarios with our approach. Nevertheless, the efficiency of the algorithm still largely depends on the number of evaluated driving profiles. An aspect we aim to investigate in the future is therefore to describe the driving profile parametrically. In this way the uncertainty, i.e., the probability distribution of the parameters can be incorporated in

the unscented transform. Other aspect that we plan to investigate is the adaption of the driving profile prediction to the driving style, which may increase the accuracy of the RDR prediction. To this aim it would be necessary to update $\mathbf{\Phi}^{va}$ according to new observed driving data.

ACKNOWLEDGMENT

The funding for this work was provided by the federal state of North Rhine-Westphalia (NRW) in frame of the project "Technology and test platform for a competence center for interoperable electromobility, infrastructure and networks" (TIE-IN).

REFERENCES

Andre, M. (2004). The ARTEMIS European driving cycles for measuring car pollutant emissions. In *Science of the total environment.*

Bowman, A. W., & Azzalini, A. (Eds.). (1997). *Applied smoothing techniques for data analysis.* New York: Oxford University Press Inc.

Conradi, P., & Hanssen, S. (2011). Dynamic cruising range prediction for electric vehicles. In *Advanced microsystems for automotive applications 2011* (p. 269-277). Springer-Verlag Berlin Heidelberg.

Daigle, M., & Goebel, K. (2010). Improving computational efficiency of prediciton in model-based prognostics using the unscented transform. In *Annual conference of the prognostics and health management society 2010.*

Daigle, M., & Goebel, K. (2011). A model-based prognostics approach applied to pneumatic valves. In *International journal of prognostics and health management.*

Daigle, M., Saxena, A., & Goebel, K. (2012). An efficient deterministic approach to model-based prediction uncertainty estimation. In *Annual conference of the prognostics and health management society 2012.*

Franke, T., Neumann, I., Bühler, F., Cocron, P., & Krems, J. (2012). Experiencing range in an electric vehicle - understanding psychological barriers. In *Applied psychology: An international review* (Vol. 61, p. 368-391).

Guzzella, L., & Sciarretta, A. (Eds.). (2005). *Vehicle propulsion systems: Introduction to modeling and optimization*. Springer Verlag, Heildelberg.

Hayes, J., Oliveira, R. de, Vaughan, S., & Egan, M. (2011). Simplified electric vehicle power train models and range estimation. In *Vehicle power and propulsion conference (VPPC), 2011 IEEE* (p. 1-5).

Hoaglin, D. C., Moesteller, F., & Tukey, J. C. (Eds.). (1983). *Understanding robust and exploratory data analysis.* Wiley.

Jongerden, M., & Haverkort, B. (2009). Which battery model to use? In *Software, IET* (Vol. 15, p. 445-457).

Julier, S., & Uhlmann, J. (2004). Unscented filtering and nonlinear estimation. In *Proceedings of the IEEE.*

Kim, E., Lee, J. L., & Shin, K. G. (2013). Real-time prediction of battery power requirements for electric vehicles. In *ACM/IEEE 4th international conference on cyber-physical systems (ICCPS 13).*

Lee, T., & Filipi, Z. (2011). Representative real-world driving cycles in Midwestern US. In *Les rencontres scientifiques d'lFP energies nouvelles - RHEVE 2011.*

Lee, T. C., Judge, G. G., & Zellner, A. (Eds.). (1970). *Estimating the parameters of the markov probability model from aggregate time series data.* North-Holland, 2nd edition.

Manwell, J. F., & McGowan, J. G. (1994). Extension fo the kinetic battery model for wind-hybrid power systems. In *Proceedings of EWEC.*

Rigatos, G. (2009). Particle filtering for state estimation in nonlinear industrial systems. In *Instrumentation and measurement, IEEE transactions on.*

Saexena, A., Celaya, J., Saha, B., Saha, S., & Goebel, K. (2009). On applying the prognostics performance metrics. In *Annual conference of the prognostics and health management society 2009.*

Yu, H., Tseng, F., & McGee, R. (2012). Driving pattern identification for EV range estimation. In *Electric vehicle conference (IEVC), 2012 IEEE international.*

BIOGRAPHIES

Javier A. Oliva received his B.S. degree in Mechanical Engineering from the University Landivar in Guatemala in 2006 and his M.S. degree in Automation and Robotics from the Technische Universität Dortmund in 2010. His research interests include probabilistic methods, diagnosis and prognostics applied to electric vehicles. He is currently working as researcher at the Institute of Control Theory and Systems Engineering from the TU Dortmund in the area of driver assistance systems for electric vehicles.

Christoph Weihrauch graduated in 2012 from the Ruhr-Universität Bochum with the Master's degree in Electrical Engineering. He is currently working as researcher at the Institute of Control Theory and Systems Engineering from the TU Dortmund in the area of charging systems for electric vehicles.

Torsten Bertram is Professor at the Technische Universität Dortmund and he directs the Institute of Control Theory and Systems Engineering. He has carried out applied research in the areas of drive systems, service robotics and development methodology.