

Parallel particle filter for state of charge and health estimation with a long term test

Simon Schwunk¹, Sebastian Straub, Nils Armbruster, Stefan Matting, Matthias Vetter

¹Fraunhofer Institut für Solare Energiesysteme, Freiburg, Germany, simon.schwunk@ise.fraunhofer.de

Abstract

The paper presents a new approach for state estimation of batteries that is able to overcome most of the obstacles for the classical Kalman filter approach. The so called particle filter is able to use any probability density function by applying monte carlo sampling methods for approximating the density functions for state of charge and state of health defined by the remaining capacity. Thereby the restriction of the Kalman filter to zero mean Gaussian distributions for all states and errors is overcome. The paper proves the validity of the approach by testing lithium metal oxide / graphite batteries with different states of health by applying different current and temperature profiles. A special focus of the testing is on electric vehicles and photovoltaic applications. For electric vehicles state of health determination achieves a correctness of 1 % or better and is a bit worse for photovoltaic applications with 3.75 % or better for ageing state between 100 % and 80 % of initial capacity. During long term testing the algorithm is validated with a decreasing state of health over time due to accelerated ageing. The state of charge estimation is always better than 1 % in long term testing and the state of health is correctly tracked over time.

lithium-ion, stochastic filter, state-of-charge, state-of-health, monte carlo, particle filter

1 Introduction

Renewable energies, which have a fluctuating power production, and electric vehicles spur interest in electrical storage, especially lithium-ion batteries, today. For being able to use battery cells in a senseful manner battery systems need electrical connections between cells, mechanical design, cooling, safety devices, electronics and software. Software and electronics are the elements which allow a sophisticated usage of the battery since they provide the possibilities to actively manage the battery by switches, cooling and heating or cell balancing to name a few and by providing the necessary information about errors, voltages, currents, temperatures and the battery's state, mainly state of charge and state of health.

Today sophisticated systems typically estimate state of health and state of charge by using Kalman filters. Since Kalman filters use some particular assumptions about the probability den-

sity functions correctness and applicability are limited. The approach presented here is applicable for all kinds of batteries and is demonstrated for batteries with lithium-metal oxide as cathode and graphite as anode material. The results show that this approach is able to provide very accurate information about the batteries' states in the variety of applications which are tested within the presented work.

2 Description of the parallel particle filter

Like the commonly used Kalman filter [1, 2, 3, 4, 5, 6, 7, 8], the particle filter also belongs to the family of recursive Bayesian filtering. While the Kalman filter for the sake of analytical calculations employs only Gaussian distributions, the different variants of particle filtering offer the possibility to deal with any possible distribution

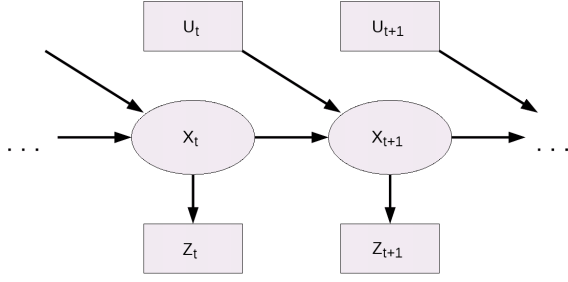


Figure 1: Bayesian network under the assumption of a Markov chain.

by employing monte carlo sampling methods.

2.1 Recursive Bayesian filtering

A recursive Bayesian filter is an algorithm for estimating the state of dynamic systems online. The filter takes the noise during the measurements into account for estimating the state. All quantities are interpreted as random variables. For each point in time there exist three quantities:

Quantity X_t

This is the state of the system at time t . The state cannot directly be observed or measured. The state is therefore the quantity which the filter shall estimate.

Quantity U_t

The quantity U_t has an influence on the state of the system and may be observed and measured.

Quantity Z_t

The quantity Z_t is a measurable quantity of the system and may be interpreted as an output. One is able to infer conclusions about X_t from Z_t .

Under the assumption of a Markov chain [9] the Bayesian network can be displayed as in figure 1. The Markov chain property assumes that a system is completely described by its state at time t and therefore independent from all states, inputs and outputs of the past. Inputs are independent from the inputs of the past and outputs are only dependent on the current state.

Often those conditions are not strictly kept. But as for the Kalman filter nevertheless, the interdependences can partly be substracted and are then insignificant enough for gaining practically useful results.

What one aims at is estimating $P(x_t|z_{1:t}, u_{1:t-1})$ which is a probability density function for being in a certain state x_t under the precondition of given values for the inputs up to time $t-1$ and for the outputs up to time t .

Applying Bayes' theorem [9] one gets the following equation with the denominator written as η .

$$\begin{aligned}
 P(x_t) &= P(x_t|z_{1:t}, u_{1:t-1}) \\
 &= P(z_t|x_t, z_{1:t-1}, u_{1:t-1}) \\
 &\quad \cdot \frac{P(x_t|z_{1:t-1}, u_{1:t-1})}{P(z_t|z_{1:t-1}, u_{1:t-1})} \\
 &= \eta^{-1} P(z_t|x_t, z_{1:t-1}, u_{1:t-1}) \\
 &\quad \cdot P(x_t|z_{1:t-1}, u_{1:t-1}) \quad (1)
 \end{aligned}$$

With marginalizing the equation over x_{t-1} and keeping in mind that only the influences of u_{t-1} and z_t and not the former values have an effect on the current state, we receive:

$$\begin{aligned}
 P(x_t) &= \eta^{-1} P(z_t|x_t) \\
 &\quad \cdot \int P(x_t|x_{t-1}, u_{t-1}) \cdot P(x_{t-1}) dx_{t-1} \quad (2)
 \end{aligned}$$

The probability density function $P(x_{t-1})$ describes the state of the time step before and therefore also all the accumulated information until the last time step. The function $P(x_t|x_{t-1}, u_{t-1})$ represents the influence of the inputs u_{t-1} on the progression of the system from state x_{t-1} to state x_t . The function $P(z_t|x_t)$ represents the probability for observing the measurement z_t given the state x_t .

2.2 Theory of the particle filter

The particle filter is a special filter introduced in [9, 10], which employs a monte carlo method for representing the probability density function. The distribution is represented by a set of samples, also called particles, therefore being able to use any probability density function for approximating the state in question if only the number of samples is high enough. If the probability of a certain value state is high, there are many samples close to this value. If the probability is low there are only a few or none. A possible implementation is shortly described below:

State transition

For estimating the set of samples representing the probability density function $P(x_t|x_{t-1}, u_{t-1})$ for each sample the influence of the input u_{t-1} is calculated with a process noise drawn from a distribution function representing the process noise. Over time there will be an increase of the variance, later on called diffusion, blurring the result of the estimation. State transition is equal to the integral in equation 2.

Weighting

For reacting on the diffusion, the measurement z_t resulting from the state x_t is taken into account. The probability function

$P(z_t|x_t)$ is used for assigning each sample $s_t^{(k)}$ a specific weight $w_t^{(k)}$. Afterwards the generated samples are normalised, so one has the following equation:

$$\sum_{k=1}^N w_t^{(k)} = 1 \quad (3)$$

Theoretically this would already be enough, especially if very large sample lots were employed. But since the sets are not infinitely large, there will be more and more samples with very little weight or even without any over time, making most samples not representing the actual distribution in the end. Therefore a resampling step is introduced.

Resampling

When employing sampling algorithms one draws new samples from a probability distribution described by a probability density function or by a lot of samples. In [9] several methods are introduced for sampling from a distribution. By resampling new samples are generated which all have the same weight making all samples significant and replace the old samples describing the probability density function more accurately.

For making the algorithm work the probability density functions from which process noise and weights are drawn must be reasonably tuned, since too small noise may lead to sample impoverishment which might lead to a collapsing particle filter [11].

2.3 Parallel particle filter for estimating state of charge and state of health

For a battery one needs at least two states for describing the state of a battery cell in a sensible manner: the state of charge and the state of health, which is the remaining capacity divided by the rated capacity. Within the particle filter the number of particles $N_{samples}$ exponentially increases with the number of states s : $N = n^s$. n represents the number of samples per state. Therefore having two states does not double the computational effort but increases it exponentially. Having 100 samples per state does not mean 100 but 10000 overall samples. Therefore analogous to the Kalman filter approach introduced by [2] not one single particle filter is proposed, but two filters working in parallel bringing down the computational cost significantly, since the number of particles only increases linearly for parallel filters: $N = s \cdot n$. The whole concept is illustrated in figure 2.

The particle filter for state of charge estimation employs a process model which basically performs ampere hour counting and a measurement model which calculates the battery's terminal voltage V_{batt} . Since self discharge and other coulombic losses are very low for all lithium-ion

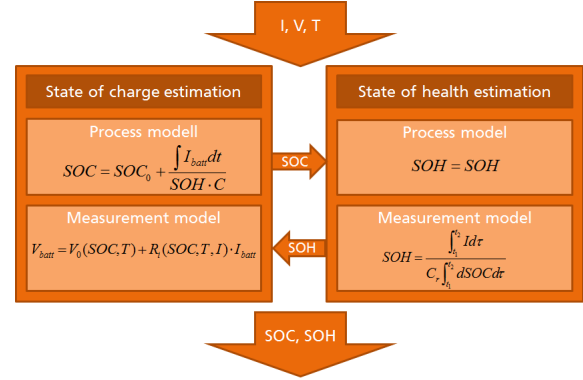


Figure 2: Two parallel particle filters. The left particle filter determines the state of charge of the battery, the right particle filter the state of health, which is the remaining capacity divided by the nominal capacity. The state of charge filter takes the other filter's output as a parameter to its process model, the state of health filter takes the output of the state of charge filter as its measurement value for the measurement process.

batteries the equation for ampere hour counting is:

$$SOC = SOC_0 + \frac{\int I_{batt} d\tau}{SOH \cdot C_r} \quad (4)$$

The measurement model for calculating the battery's terminal voltage V_{batt} may be any model reproducing the electrical behaviour. Mostly the algorithms run on small microcontrollers with many more tasks which are also more critical than estimating the battery's state and therefore will have a higher priority, too. Since the filter is able to handle model errors statistically by assuming a suitable probability density function for the calculated battery's terminal voltage a quasistationary battery model is chosen. It consists of a voltage source representing the open circuit voltage and an ohmic resistance representing the ohmic losses during a quasistationary operation:

$$V_{batt} = V_0(SOC) + R_i(SOC, T, I) \cdot I_{batt} \quad (5)$$

Dynamics are not captured by the model, but the influence of temperature and state of charge are taken into account which is very important for calculating a terminal voltage close to the real one in average. In real applications dynamics only influence the battery's behaviour in its short term while temperature and state of charge have a major impact for longer terms. Thus the influences of the dynamics are only regarded stochastically by the particle filter.

After estimating the state of charge the quantity is passed to the state of health filter and then it is used as a measurement value. The state of health filter assumes within the process model that the state of health of the battery remains the same:

$$SOH(t_0) = SOH(t) = \frac{C_{batt}}{C_r} \quad (6)$$

C_{batt} is the actual battery capacity while C_r is the rated battery capacity.

For the measurement model the basic assumption is that the state of charge estimation still works correctly though the estimated capacity might be wrong. The open circuit voltage of the battery in respect to the state of charge shall not change with the ageing process of the battery as is shown in [2]. The measurement model then compares the state of charge estimation between two points in time with the amount of charge flowing in and out of the battery in the same time interval:

$$SOH = \frac{\int_{t_1}^{t_2} I_{batt} d\tau}{C_r \int_{t_1}^{t_2} dSOC d\tau} \quad (7)$$

For being able to detect the right state of health it is therefore necessary that the measurement model of the state of charge filter has significant influence on the state of charge determination. Otherwise ampere hour counting would be compared resulting in no change of the estimated capacity.

2.4 Implementation of the state of charge filter

First, the state of charge filter is initialized. Since the actual state of charge is generally unknown all samples are equally distributed over the complete state of charge range $[0, 1]$ resulting in a medium state of charge of about 0.5. Within the process model for each sample the current flowing into the battery is integrated over time and a noise ϵ is added:

$$s_{t+1}^{(k)} = s_t^{(k)} + \frac{(I_{batt} + \epsilon_{SOC})\Delta t}{SOH \cdot C_r} \quad \forall k = 1, \dots, N \quad (8)$$

The noise is sampled from a Gaussian distribution with the variance σ^2 and the mean μ :

$$N(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\} \quad (9)$$

The method for drawing noise from a gaussian distribution was implemented by using the Marsaglia-Polar method. But the particle filter also enables using any other possible distribution.

After the prediction of the samples taking the inputs into account they are compared to the measurement values via a model for the terminal voltage. Equation 5 determines the terminal voltage and this value is compared to the measured voltage. A Cauchy-Lorentz distribution which is a special case of the Student-t distribution gives a value for the probability that the sample is correct.

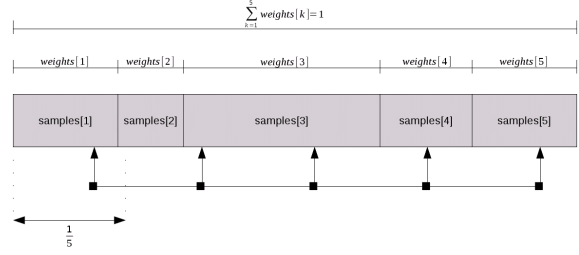


Figure 3: Low variance filtering for resampling from a sample lot of weighted samples.

$$p(x) = \frac{I}{\pi} \cdot \frac{\gamma}{\gamma^2 + (x - x_0)^2} \quad (10)$$

The distribution is parameterised by I , γ and x_0 ; I is the scaling factor for the height, γ is the scaling factor for the width and x_0 is the position of the distribution which is set equal to the measurement value. With having now pairs consisting of samples and their probabilities or weights resampling methods need to be used. Due to the easy implementability and the quick calculation times low variance filtering is used for generating a new set of samples [12]. First of all the probabilities or weights are normalised:

$$\sum_{k=0}^N w_{SOC}^{(k)} = 1 \quad (11)$$

The probabilities are then added to a long line as depicted in figure 3. Then a starting value within the interval $[0, \frac{1}{N}]$ is drawn randomly. Afterwards one goes through the sample lot with steps of size $\frac{1}{N}$ and picks exactly the same amount of samples. Samples with small weight are probably overgone and samples of bigger weight might be drawn several times.

2.5 Implementation of the state of health filter

The state of health filter works accordingly, the only difference being that the process model assumes a constant capacity and the measurement model compares state of charge differences with the charge flowing in and out of the battery.

The process model therefore simply adds a certain ϵ to each sample:

$$C_{sample}(t+1)[k] = C_{sample}(t)[k] + \epsilon_{SOH} \quad \forall k = 1, \dots, N \quad (12)$$

The noise ϵ is drawn from a Gaussian distribution with a low variance.

Between two points in time, t_1 and t_2 , the measurement model first integrates the current flowing in and out of the battery, calculating an amount of charge Q by which the battery has

been charged or discharged. For each of the samples representing the capacity of the battery an expected change in state of charge is calculated as a next step:

$$\frac{\Delta SOC_{expected}[k]}{C_{sample}[k]} = \forall k = 1, \dots, N \quad (13)$$

This value is compared to the change in value determined by the state of charge filter in the time between t_1 and t_2 . The difference is used as the parameter x_0 of the Cauchy-Lorentz distribution in equation 10.

With this distribution a probability/weight is assigned to each sample for the resampling step, which generates the new sample lot describing the capacity. For determining the state of health the average of all capacity samples is calculated and normalized by the rated capacity.

2.6 Comparison with Kalman filter and computational effort

By using the particle filter approach, one solves the problem of a fixed distribution posed by the Kalman filter but one must take into account that the solution is non optimal due to the limited set of samples. The run-time of the particle filter increases linearly with the amount of samples. With a reasonable choice for the amount of states and samples and low computing methods for sampling and resampling the need for resources is reasonable. Therefore the algorithm runs very well on a microcontroller like it is used on many common battery management systems.

2.7 Notes on validation

For validating the method for state determination several sets of measurement data were generated. These data sets came from validation profiles which represent different applications.

On the one hand there is a profile which describes the current behaviour of an electric vehicle (see upper diagramme in figure 4). This validation profile is a combination of four official driving profiles. A European profile (*NEDC*), the Urban Dynamometer Driving Schedule (*UDDS*), the Federal Test Procedure (*FTP*) and the Speed Correction Driving Schedule (*SCDS*). All of them show strong dynamics and high currents.

On the other hand there is a profile which describes the current behaviour of a photovoltaic system (see lower diagramme in figure 4). This validation profile was developed by *Fraunhofer ISE* and shows lower dynamics and lower currents. Nevertheless, the validation profile passes through a wide range of state of charge.

Based on these two profiles more validation profiles were created. For investigating the performance of the method with various temperatures the electric vehicle profile (EV profile) was combined with a temperature profile (for results

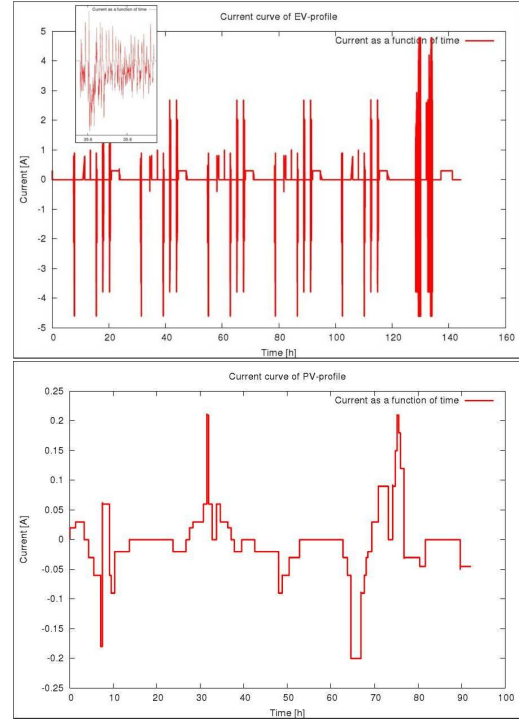


Figure 4: Electric vehicle profile (top) and photovoltaic profile (bottom) are depicted; the EV profile shows strong dynamics and high currents but with often and long zero current phases, a typical profile for battery systems in photovoltaic applications shows lower dynamics and lower currents but passes through a wide range of state of charge and has only short zero current phases.

see figure 6). For studying the long term behaviour of the method with decreasing state of health parts of the electric vehicle profile were performed periodically (for results see figure 7).

2.8 Parameterisation of filters

During implementation both filters were parameterized. The following parameters had to be determined.

Variance σ^2 and mean μ of Gaussian noise function in process model of state of charge filter:

The noise ϵ_{SOC} should represent the measurement error of the current measurement. An error with zero mean ($\mu = 0$) was assumed. Due to a highly accurate current measurement the variance σ^2 might be very low. A small process noise, however, leads to "sample impoverishment" [11] which must be avoided. Because of that, the variance σ^2 was overestimated intentionally which is called "jittering" [13]. Therefore, the Gaussian noise function parameters were set to $\mu = 0$ and $\sigma^2 = 0.25 \text{ A}^2$.

Variance σ^2 and mean μ of Gaussian noise

function in process model of state of health filter:

The noise ϵ_{SOH} should represent the error of the assumption $SOH_{t+\Delta t} = SOH_t$. Again, an error with zero mean ($\mu = 0$) was assumed. The variance σ^2 was difficult to quantify because no measureable errors could be taken into account. Finally, the variance σ^2 was chosen such that "sample impoverishment" is avoided. The Gaussian noise parameters were set to $\mu = 0$ and $\sigma^2 = 2.5 \cdot 10^{-9}$.

Scaling parameter γ of Cauchy-Lorentz distribution in measurement model of state of charge filter:

The scaling parameter γ should define an optimal Cauchy-Lorentz distribution for computing the weight of each sample. Therefore, a parameter search was performed. Various parameters of the interval $[0, 1]$ were tested in an automated manner. Finally, the mean squared error of state of charge estimation and reference state of charge was analysed when $t \geq 2$ h (approximate transient time). Again, to prevent "overfitting" four different data profiles were used and other data sets were used for validation. To minimize stochastic influences every parameter was tested eight times. The results are shown in figure 5. At last, the scaling parameter s of the Cauchy-Lorentz distribution in the measurement model of state of charge filter was set to 0.125 V. The scaling parameter I is set to 1 V as the height of the distribution is not relevant.

Scaling parameter γ of Cauchy-Lorentz distribution in measurement model of state of health filter:

Again, the scaling parameter γ should define an optimal Cauchy-Lorentz distribution for computing the weight of each sample. Therefore, a parameter search tested various parameters in the interval $[0, 0.5]$. The mean squared error of state of health estimation and reference state of health was analysed when $t \geq 20$ h due to the fact that the state of health filter needs more transient time than the state of charge filter. To prevent "overfitting" four different data profiles were used, too. The scaling parameter s of the Cauchy-Lorentz distribution in the measurement model of the state of health filter was set to 0.01. The scaling parameter I is set to 1 as it is not relevant.

Time interval between two resampling steps in the state of charge filter:

The resampling step was introduced to avoid "particle degeneracy", which means that many particles have a probability of close to zero in the end. The resampling step is performed periodically. In the state of charge filter the resampling step is performed every six minutes.

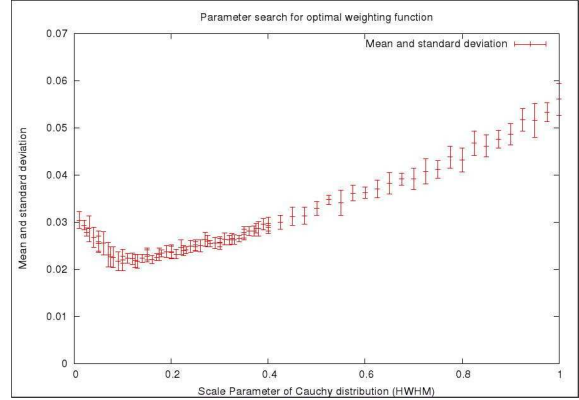


Figure 5: Results of the parameter search for scaling parameter s of Cauchy-Lorentz distribution; mean and standard deviation of mean squared error between state of charge estimation and reference state of charge as a function of scaling parameter s .

Time interval between two resampling steps in the state of health filter.

Again, the resampling step is performed periodically. This time, the resampling step is performed every 21 minutes, $\Delta t_{resampling} = 0.35$ h. In the state of health filter the resampling step is crucial for estimating the state of health. This results from the fact that the process model does not take measurements into account. Because of that only the measurement model is responsible to estimate the state of health. Two additional conditions were implemented to guarantee that the resampling step is only performed when the input data is reliable. Because of that it must apply: $0.002 \leq \int_{t_2}^{t_1} dSOC d\tau \leq 0.1$.

3 Validation

In section 2.7 several validation profiles are introduced. For validation purposes the method estimated the state of charge and state of health for these reference data sets.

For state of charge estimation an average and a maximum error was calculated for the several validation profiles:

$$F_{SOC,average} = \frac{1}{M} \sum_{i=0}^M |SOC_{est}(i) - SOC_{ref}(i)| \quad (14)$$

$$F_{SOC,maximum} = \max(|SOC_{est}(i) - SOC_{ref}(i)|) \quad (15)$$

For pulse testing only the time between 15 h and 50 h was analysed because of the lack for possible recalibration of the reference. For the other

data sets recalibrated reference data was used and the error was determined for any data points after two hours when the algorithm had enough time to find the correct state of charge value. For state of health estimation the average and maximum error after 40 h was determined, since the algorithm needs more time to adjust correctly:

$$F_{SOH,average} = \frac{1}{M} \sum_{i=0}^M |SOH_{est}(i) - SOH_{ref}| \quad (16)$$

$$F_{SOH,maximum} = \max(|SOH_{est}(i) - SOH_{ref}|) \quad (17)$$

For the long term validation only the specific moments in time, when a capacity test was performed, were analysed.

3.1 Electric vehicle profile

Table 1 shows results for batteries with states of health of 0.94, 0.89 and 0.80 at 20 °C. Mean error is highest for the newest battery with 1.25 % while 1 % or less for the other two batteries. Maximum error is highest for the battery at 0.89 of state of health. Overall errors are really low for the state of charge and the algorithm works reliably.

For state of health errors are even better in this applications, with the highest mean error of 1.02 % for the 0.8 state of health battery and the highest maximum error of 2.9 % for the 0.89 state of health battery.

3.2 Photovoltaic profile

Table 1 shows results for batteries with states of health of 1.00, 0.94, 0.89 and 0.80. The state of charge estimation is more difficult than for the electric vehicle. Highest mean and maximum error occur for the newest battery with 3.9 % and 8.9 % of error, being best for the 0.89 state of health battery with 2.88 % and 4.5 % respectively.

For state of health estimation results are by far worst for the newest battery with 3.75 % of mean error and 5.9 % of maximum error, too. It is the lowest for the 0.89 state of health battery resulting in 0.79 % mean error and 1.3 % of maximum error.

3.3 Temperature dependency

The results of state of charge estimation when using a data set based on the electric vehicle profile with various temperature are shown in figure 6. Mean error of state of charge estimation is about 1.5 %. The maximum error is about 6.2 %. The errors are slightly higher than for the fixed temperature, but reliable. The profile put special stress on the battery due to steep ambient temperature gradients which in most applications do not

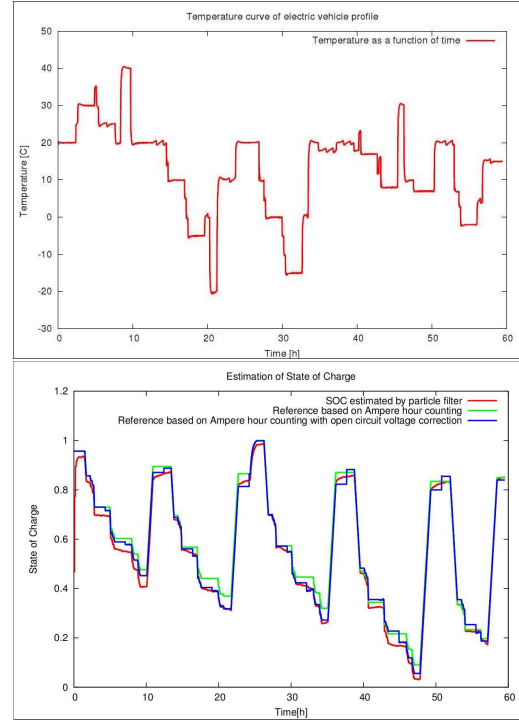


Figure 6: Temperature as a function of time is on top, state of charge estimation and references at the bottom; mean error of state of charge estimation is below 2 %. Since the profile goes for more than 140 h, the accumulated error of the current measurement of the BaSyTec measurement device is already rather high. Therefore the reference is recalibrated after long rest phases by the the open circuit voltage of the battery. The sudden changes in the recalibrated reference state of charge come from the recalibration after three hours of rest.

occur. With a thermal model accuracy could be enhanced.

In real applications accuracy will probably be higher in most cases.

3.4 Long term validation

The results of state estimation when using a long term data set based on parts of the electric vehicle profile with decreasing state of health are shown in figure 7. Mean error of state of charge estimation is about 0.8 %. Maximum error of state of charge estimation is about 3.5 %. State of health errors are below 1.9 % during the long term tests.

3.5 Pulse profile

For very small changes in the state of charge detecting these changes is more difficult which makes it more difficult for the state of health filter to rely on its measurement model. On the other hand errors in the current measurement especially offset errors have a bigger impact on the

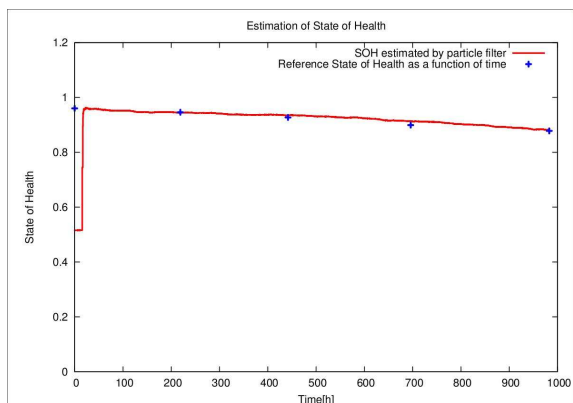


Figure 7: State of health estimation and reference for long term testing at 40 °C ambient temperature. The blue crosses show the capacity determined in regular intervals by a regular capacity test.

accuracy of the state of health determination. To clarify if the effects one expects from theory also occur in practical operation pulse testing of batteries was used. The results are displayed in table 1 under the section pulse profile. The results show that the state of charge range for this kind of battery does not have a significant impact due to the steep and over a wide range quite linear open circuit voltage characteristics. What has a significant impact is the state of charge difference applied in the pulsed current profiles. In principal the algorithm can cope very well with small ranges since the state of charge estimation works accurately, but with smaller state of charge difference the offset of the current measurement gains more influence and increases the error. For $\Delta SoC = 0.005$ the filter has got an error in state of health estimation at least two times as high as for the tests with bigger ΔSoC .

4 Conclusions

This paper introduces a new method for state of charge and state of health estimation, the particle filter. For different applications, changing temperatures, in long term testing and synthetic pulse profiles the method is validated with very good results for electric vehicle applications and good results for photovoltaic applications. First hand this is a surprise due to the higher dynamics of the electric vehicle profile. But the rest phases are longer and there are full charges, which makes it easier for the filter to make good estimations. Overall state of health estimation is more robust than state of charge estimation, for which higher deviations occur.

Even for low state of charge swings the particle filter is able to determine the state of health correctly making it a suitable method even for micro cycling applications.

Overall the framework delivers a new basis for overcoming restrictions of the Kalman filter regarding the assumptions about probability distributions. On the other hand, the computational ef-

fort is still low enough to be embedded in small and low cost microcontrollers.

The method is currently being adapted for use with lithium iron phosphate based batteries and first results provide good results for both state of charge and state of health estimation.

The work was funded by the German Ministry of Education and Research (BMBF).

5 Abbreviations

X_t State of the system at time t

U_t Influence on the state of the system at time t

Z_t Measurable quantity of the system at time t

$P(a)$ Probability of a

SOC/SoC State of Charge

SOH/SoH State of Health

C_r Rated capacity of the battery

$N/N_{samples}$ Number of particles

s Number of states

n Number of particles per state

Fraunhofer ISE Fraunhofer Institute for Solar Energy Systems

EV profile Electric vehicle profile

PV profile Photovoltaic profile

Table 1: Results for the state of charge and state of health estimation for all validation profiles in this paper. For each battery its state of health SoH is indicated in the first column, for the long term tests it is the state of health at the beginning of the testing. For pulse testing two additional columns are introduced: the maximum state of charge SoC during pulse testing and the ΔSoC which indicates the depth of discharge. The results with the mean and maximum errors are indicated in the four columns on the right side. **State of Charge:** (1) Recalibrated ampere hour counting as a reference; mean error and maximum error after two hours. (2) Ampere hour counting as a reference; mean error and maximum error between 15 and 50 hours. **State of Health:** Capacity test as a reference; mean error and maximum error after 40 hours.

Profile	SoH	SoC	ΔSoC	State of Charge		State of Health	
				mean Error	maximum Error	mean Error	maximum Error
EV-profile (1)	0.94	-	-	1.25%	4.2%	0.71%	1.4%
EV-profile (1)	0.89	-	-	1.00%	4.8%	0.85%	2.9%
EV-profile (1)	0.80	-	-	0.82%	3.9%	1.02%	2.7%
PV-profile (1)	1.00	-	-	3.90%	8.9%	3.75%	5.9%
PV-profile (1)	0.94	-	-	2.88%	5.8%	1.72%	2.9%
PV-profile (1)	0.89	-	-	2.88%	4.5%	0.79%	1.3%
PV-profile (1)	0.80	-	-	3.15%	6.7%	0.93%	4.4%
Pulse profile (2)	0.96	0.80	0.01	0.93%	2.0%	2.06%	4.8%
Pulse profile (2)	0.96	0.50	0.01	0.39%	1.0%	5.83%	6.9%
Pulse profile (2)	0.96	0.20	0.01	0.56%	1.4%	6.19%	7.1%
Pulse profile (2)	0.91	0.80	0.01	1.41%	2.2%	2.75%	4.0%
Pulse profile (2)	0.91	0.50	0.02	1.36%	1.8%	0.99%	1.9%
Pulse profile (2)	0.91	0.50	0.01	0.44%	1.2%	3.06%	3.8%
Pulse profile (2)	0.91	0.50	0.005	0.61%	1.4%	6.27%	7.0%
Pulse profile (2)	0.91	0.20	0.01	0.68%	1.5%	1.34%	2.0%
Long-term profile (1)	0.91	-	-	0.58%	7.4%	1.88%	4.2%
Long-term profile (1)	0.96	-	-	0.77%	3.5%	1.27%	3.5%
Temp. dep. profile (1)	1.00			1.59%	6.2%	2.53%	3.2%

References

- [1] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 85:35–45.
- [2] Lee, S., Kim, J., Lee, J., and Cho, B. (2008). State-of charge and capacity estimation of lithium-ion battery using a new open-circuit voltage versus state-of-charge. *Journal of Power Sources*, 185(2):1367–1373.
- [3] Urbain, M., Rael, S., Davat, B., and Desprez, P. (2007). State estimation of a lithium-ion-battery through kalman-filter. *IEEE-PESC'07*, pages 2804–2810.
- [4] Piller, S., Perrin, M., and Jossen, A. (2001). Methods for state-of-charge determination and their application. *Journal of Power Sources*, 96(1):113–120.
- [5] Lee, J., Nam, O., and Cho, B. (2007). Li-ion battery soc estimation method based on the reduced order extended kalman filtering. *Journal of Power Sources*, 174(7):9–15.
- [6] Plett, G. L. (2004a). Extended kalman filtering for battery management systems of lipb-based hev battery packs: Part1. background. *Journal of Power Sources*, 134(2):252–261.
- [7] Plett, G. L. (2004b). Extended kalman filtering for battery management systems of lipb-based hev battery packs: Part2. modeling and identification. *Journal of Power Sources*, 134(2):262–276.
- [8] Plett, G. L. (2004c). Extended kalman filtering for battery management systems of lipb-based hev battery packs: Part3. state and parameter estimation. *Journal of Power Sources*, 134(2):277–292.
- [9] Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.
- [10] Michael Isard, A. B. (1998). Condensation – conditional density propagation for visual tracking. *Int. J. Computer Vision*, 29(1):‘5–28.
- [11] Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian

tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188.

- [12] Chen, Z. (2003). Bayesian filtering: From kalman filters to particle filters, and beyond. *Statistics*, pages 1–69.
- [13] Gustafsson, F., Gunnarson, F., Bergman, N., Forssell, U., Jansson, J., Karlsson, R., and Nordlund, P.-J. (2002). Particle filters for positioning, navigation, and tracking. *IEEE Transactions on Signal Processing*, 50(2):425–437.

Authors

Simon Schwunk, M.Sc., head of team Battery Modules and Systems. Electrical Engineering at the University of Applied Sciences Mannheim, finished in 2006 with Dipl.-Ing. (FH). Electrical engineering at University of Hagen, finished with M.Sc. in 2008. He works as researcher and project manager at Fraunhofer ISE since 2005 and is head of team since 2011. Main fields of work: Modelling and simulation of batteries, state determination, monitoring and control algorithms for batteries, design of complete battery systems.



Sebastian Straub, B.Sc., received a bachelor of science in electrical engineering from the University of Applied Sciences Furtwangen in 2011. He wrote his bachelor thesis at the Fraunhofer Institute for Solar Energy Systems in Freiburg and works there currently as a researcher for state determination for lithium-ion batteries.



Stefan Matting, studying mathematics and IT at the University of Freiburg. Since 2007 he works at the Fraunhofer Institute for Solar Energy Systems in Freiburg as a researcher in the fields simulation of PV hybrid systems and state estimation of lithium-ion batteries.



Nils Armbruster, M.Eng., received a bachelor of engineering in mechatronics from the University of Applied Sciences Offenburg in 2008. In 2009 he received his master of engineering in electrical engineering from the University of Applied Sciences Karlsruhe. Since 2009 he works as a researcher at the Fraunhofer Institute for Solar Energy Systems in Freiburg. Main topics of work are: State determination of lithium-ion batteries and design and implementation of battery management systems.



Dr.-Ing. **Matthias Vetter**, Electrical Engineer, 13 years of experience at Fraunhofer Institute for Solar Energy Systems ISE. PhD thesis in the field of modeling and development of control strategies for fuel cell systems. Until 2005 project manager in the field of modeling, simulation and development of control strategies for distributed power generation systems. Until 2010 head of group Off-Grid Power Supply. Since 2011 head of department PV Off-Grid Solutions And Battery System Technology. Work topics among others: Autonomous systems and mini-grids, decentralized grid connected PV battery systems, development of battery systems for stationary and automotive applications. Development and optimization of battery management systems, energy management systems as well as supervisory control strategies.

