MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Battery State of Charge Estimation Based on Regular/Recurrent Gaussian Process Regression

Sahinoglu, Gozde Ozcan; Pajovic, Milutin; Sahinoglu, Zafer; Wang, Yebin; Orlik, Philip V.; Wada, Toshihiro

TR2017-124 October 19, 2017

Abstract

This paper presents novel machine-learning based methods for estimating state of charge (SoC) of Lithium-ion (Li-ion) batteries which use Gaussian process regression (GPR) framework. The measured battery parameters, such as voltage, current, temperature, are used as inputs for regular GPR whereas the SoC estimate at the previous sample is fed back, and incorporated into the input vector for recurrent GPR. The proposed methods consist of two parts. In the first part, training is performed wherein the optimal hyperparameters of a chosen kernel function are determined to model data properties. In the second part, online SoC estimation is carried out according to the trained model. One of the practical advantages of a GPR framework is to quantify estimation uncertainty, and hence to enable reliability assessment of the battery SoC estimate. The performance of the proposed methods is evaluated by using simulated dataset and two experimental datasets, one with constant and the other with dynamic charge and discharge currents. The simulations and experimental results show the superiority of the proposed methods in comparison to state-of-the-art techniques including support vector machine (SVM), relevance vector machine (RVM) and Neural Network (NN).

IEEE Transactions on Industrial Electronics

Battery State of Charge Estimation Based on Regular/Recurrent Gaussian Process Regression

Gozde O. Sahinoglu, *Member, IEEE*, Milutin Pajovic, *Member, IEEE*, Zafer Sahinoglu, *Senior Member, IEEE*, Yebin Wang, *Member, IEEE*, Philip V. Orlik, *Senior Member, IEEE*, and Toshihiro Wada

***Abstract*—This paper presents novel machine-learning based methods for estimating state of charge (SoC) of Lithium-ion (Li-ion) batteries which use Gaussian process regression (GPR) framework. The measured battery parameters, such as voltage, current, temperature, are used as inputs for regular GPR whereas the SoC estimate at the previous sample is fed back, and incorporated into the input vector for recurrent GPR. The proposed methods consist of two parts. In the first part, training is performed wherein the optimal hyperparameters of a chosen kernel function are determined to model data properties. In the second part, online SoC estimation is carried out according to the trained model. One of the practical advantages of a GPR framework is to quantify estimation uncertainty, and hence to enable reliability assessment of the battery SoC estimate. The performance of the proposed methods is evaluated by using simulated dataset and two experimental datasets, one with constant and the other with dynamic charge and discharge currents. The simulations and experimental results show the superiority of the proposed methods in comparison to state-of-the-art techniques including support vector machine (SVM), relevance vector machine (RVM) and Neural Network (NN).**

***Index Terms*—Battery management system, Lithium-ion battery, recurrent/regular Gaussian process regression, state of charge estimation.**

I. Introduction

Since commercial introduction in the early 1990s, Lithium-ion (Li-ion) batteries have become a part of daily life, i.e., powering electric vehicles, consumer electronics such as cellphones and tablets, and supporting residential photovoltaic systems, smart grid systems, etc. Li-ion batteries have gained increasing popularity over other types of batteries with different chemistries due to low self-discharge rate, small memory effect, high cell voltage, high energy density, lightweight, long lifetime, and low maintenance [1]. However, a battery can support only a finite, limited number of charge and discharge cycles. Thus, an improper use deteriorates battery's performance and eventually shortens its life span. Therefore, battery management system (BMS) is of great importance in order to ensure that a battery is operated within its specified safety limits. One of the key tasks of the BMS is to determine the state of charge (SoC), which is defined as the percentage of available charge remaining in the battery. The SoC indicates when the battery should be recharged, and hence it enables the BMS to prolong the battery life by protecting the battery from over-discharge and over-charge events. However, the SoC cannot be directly measured because Li-ion batteries store energy in a chemical form within a closed system, and this energy cannot be directly accessed. Instead, the SoC can be estimated based on measurable battery parameters such as voltage, current, temperature, etc. Accurate SoC estimation of Li-ion battery is still a challenging task due to nonlinear battery dynamics as well as variations of operating conditions such as temperature.

Manuscript received October 18, 2016; revised February 24, 2017, May 26, 2017 and July 25, 2017; accepted September 13, 2017.

G. O. Sahinoglu is with Broadcom Limited, Irvine, CA, 92617, USA (e-mail: gozde.sahinoglu@broadcom.com).

M. Pajovic, Y. Wang, and P. V. Orlik are with Mitsubishi Electric Research Laboratories, 201 Broadway, Cambridge, MA 02139, USA (e-mail: pajovic@merl.com; yebinwang@merl.com; porlik@merl.com).

Z. Sahinoglu is with Mitsubishi Electric US Inc., Cypress, CA, 90630, USA (e-mail: zafer.sahinoglu@meus.mea.com).

T. Wada is with the Advanced Technology R&D Center, Mitsubishi Electric Corporation, 8-1-1, Tsukaguchi-honmachi, Amagasaki City, 661-8661, Japan (e-mail:Wada.Toshihiro@bx.MitsubishiElectric.co.jp).

M. Pajovic is the corresponding author, phone:+1-617-6217531, fax:+1-617-6217550, e-mail:pajovic@merl.com.

A. Literature Review

A variety of SoC estimation methods have been developed (see e.g. [2–24]). For instance, the open circuit voltage (OCV) method in [2], [3] was proposed to estimate the battery SoC based on OCV measurements. However, the proposed method is not suitable for online SoC estimation since the battery has to be disconnected from any load for some time to reach a steady state before measuring OCV. Also, mapping from OCV measurements to corresponding SoC values is difficult for batteries exhibiting a wide flat region on the OCV-SoC curve. In contrast to the OCV method, the Columb counting method in [4], [5] was employed to estimate the SoC during battery operation by measuring the current and integrating it over some time interval. This method suffers from sensitivity to the initial SoC value, and accumulation of measurement and calculation errors due to integration.

In order to overcome these limitations, simplified equivalent circuit models, which approximate the battery dynamics, have been used for SoC estimation. In particular, the authors in [6] applied a Kalman filter (KF) to estimate the SoC of Li-ion batteries using a linear battery model. However, the battery is a highly nonlinear system. Therefore, the authors in [7] employed an extended Kalman filter (EKF) for nonlinear battery

state estimation. In order to further improve the accuracy, an adaptive EKF was adopted in [8], where the noise statistics are estimated and modified based on observation data. The recent work in [9] developed a novel adaptive SoC estimation scheme based on an iterated EKF, in which the joint estimation of the SoC and model parameters is performed. The authors in [10] estimated both the SoC of Li-ion batteries using the EKF and the internal resistance which is directly correlated with the battery's state-of-health (SoH). However, the EKF linearizes the nonlinear dynamic system, which leads to inevitable linearization error, and hence compromises estimation accuracy. To address this problem, the authors in [11], [12] estimated the SoC of Li-ion batteries using an unscented Kalman filter (UKF), which approximates the state distribution through a set of sample points, called sigma points and hence avoids model linearization at the expense of higher computational complexity. Although the KF based SoC estimation methods achieve acceptable estimation performance, the corresponding estimation accuracy is strongly affected by a chosen battery model and parameters. The authors in [13] recently showed the impact of parameter uncertainties on the SoC estimation.

In recent years, there has been a growing interest in machine-learning based SoC estimation methods, which eliminate the need for detailed physical knowledge of the battery and learn the nonlinear relationship between the SoC and measured quantities such as voltage, current, temperature. For instance, the authors in [14] applied a multi-layer feedforward neural network (NN) to estimate the SoC of Li-ion batteries as a function of current, voltage and temperature of the battery. The estimation accuracy of the proposed method was further improved by applying the UKF to the SoC estimates of the NN. Also, in [15], a hybrid SoC estimation method was introduced based on a combination of a radial basis function NN, an orthogonal least-squares algorithm and an adaptive genetic algorithm. In [16], a fuzzy NN based SoC estimation method was proposed. Additionally, the authors in [17] trained a three layer feed-forward NN using parallel chaos immune evolutionary programming to estimate the SoC of Nickel-metal hydride batteries as a function of battery terminal voltage, voltage derivative, voltage second derivative, discharge current and temperature. The authors in [18] applied the NN to find the state-space model of the SoC, which is then employed by the EKF for estimating the SoC of Li-ion batteries. Moreover, the SoC estimation methods based on a support vector machine (SVM) were introduced in [19–24]. In particular, the SoC of the battery was estimated as a function of current and voltage in [19]. On the other hand, voltage, current and temperature measurements were used as inputs to train the SVM model and estimate the SoC in [20], [21]. In addition, the authors in [22] developed an SoC estimation method based on the SVM in which the hyperparameters were determined through double search optimization process. The proposed method was validated through simulation results obtained by using advanced vehicle simulator [23]. The authors in [24] proposed an SoC estimation method based on fuzzy least square SVM and the method was validated through temperature, voltage and current data experimentally obtained from different driving conditions of the electric vehicle.

B. Main Contributions

In this paper, we propose novel data-driven methods for estimating SoC of Li-ion batteries based on Gaussian process regression (GPR) framework. In particular, the proposed methods are built on GPR, which is a probabilistic, and nonparametric machine learning method, to obtain predictive probability distribution of the SoC rather than just a point estimate of the SoC. The main contributions of this paper can be summarized as follows:

- We developed SoC estimation methods based on recurrent GPR and autoregressive recurrent GPR that model the nonlinear dependence of the SoC on the voltage, current and temperature. The recent works [25–28] applied regular GPR to estimate state of health (SoH) and SoC of Li-ion batteries, respectively. More specifically, the authors in [25], [26] performed prognostic predictions of battery SoH using the number of cycles as an input to a GPR model. The study in [27] analyzed the impact of different covariance functions on the SoC estimation performance of the regular GPR, where the inputs are present values of voltage, current and temperature of the battery. Also, the authors in [28] estimated the battery SoC using sparse GPR, where a subset of training data points, called inducing points are used for training the regression model instead of an entire training dataset as in the regular GPR. Different from these works, we proposed a new data-driven method based on recurrent GPR to estimate SoC of Li-ion batteries wherein the SoC estimate in the previous discrete time instant is fed back to the input. Hence, the measurements of the voltage, current and temperature, together with the previous SoC estimate, are used to estimate the current SoC. To further improve the estimation accuracy, we applied an autoregressive recurrent GPR which exploits the present and past values of voltage, current and temperature measurements together with the previous SoC estimate.
- Different from the work in [27], we performed a comparative study of the performance of the proposed methods and the state-of-the-art techniques including SVM, relevance vector machine (RVM) and NN based SoC estimation methods in terms of estimation error and computational time. The main advantages of the GPR over the SVM, NN and other non-Bayesian machine learning methods are the explicit probabilistic formulation and analytically tractable inference leading to closed-form expressions. A Bayesian framework enables the GPR model to quantify the confidence intervals around the estimates, which further allows to asses the uncertainty in the estimates and hence provide more informative outputs than non-Bayesian models.
- Also, the GPR model has a simple parameterization and the model parameters (e.g., hyperparameters in the kernel function) can be computed by maximizing a marginal log-likelihood function, which is easy to implement and flexible to use, in contrast to commonly used grid-searching, trial-and-error methods used to optimize the SVM. In addition, hyperparameter estimation using a Bayesian

learning in the GPR model leads to a form of automatic relevance determination. Hence, influence of each input variable on the output can be easily determined, which is a practical advantage of the GPR over non-Bayesian machine learning methods.

- The proposed methods are based on data-driven models, they can be easily applied to estimate the SoC of different battery types and chemistries. In addition, they are highly suitable for an online SoC estimation due to low computational complexity in the testing stage and the fact that they do not require a battery be disconnected from the load to perform the SoC estimation.

The rest of the paper is organized as follows: Section II gives a brief overview of the theory of GPR. Section III presents an SoC estimation method using regular GPR. Section IV introduces an SoC estimation method based on recurrent GPR. In order to further smooth out SoC estimates, an SoC estimation method based on autoregressive recurrent GPR is described in Section V. Simulation and experimental results are provided and discussed in Section VI. Finally, main conclusions are drawn in Section VII.

II. Background

In this section, we initially provide a brief overview of the theory of GPR, and then introduce SoC estimation methods based on regular GPR, recurrent GPR and autoregressive recurrent GPR in the following sections.

Let $\mathcal{D} = (\mathbf{X}, \mathbf{y})$ denote a training data set, comprising D-dimensional N input vectors $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$, where $\mathbf{x}_n \in \mathbb{R}^D$, and the corresponding outputs $\mathbf{y} = \{y_n\}_{n=1}^N$, where $y_n \in \mathbb{R}$. It is assumed that there is an underlying latent function $f(.)$, which maps the inputs, $\mathbf{x}_n$, to their corresponding output values, y_n,

$$y_n = f(\mathbf{x}_n) + \varepsilon_n, \tag{1}$$

where ε_n denotes zero-mean additive Gaussian noise with variance σ_n^2, i.e., $\varepsilon_n \sim \mathcal{N}(0, \sigma_n^2)$, and $\{\varepsilon_n\}_{n=1}^N$ form an independent and identically distributed (i.i.d) sequence. The key assumption in the GPR is that any set of function values $\mathbf{f} = [f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots, f(\mathbf{x}_n)]^T$ is distributed according to a multivariate Gaussian distribution [29]

$$p(\mathbf{f}|\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n) = \mathcal{N}(\mathbf{0}, \mathbf{K}). \tag{2}$$

Above, $\mathbf{f} = [f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots, f(\mathbf{x}_n)]^T$, and $\mathbf{0}$ is an $N \times 1$ vector whose elements are all 0. In addition, $\mathbf{K}$ is a kernel matrix, whose entries $\mathbf{K}_{ij} = k_s(\mathbf{x}_i, \mathbf{x}_j)$ correspond to the values of the kernel function evaluated at each pair of training inputs.

The kernel function plays an important role in GPR since it encodes the prior assumptions about the properties of the underlying latent function that we are trying to model. In this work, we adopt the commonly used squared exponential (SE) kernel, defined as [29]

$$k_s(\mathbf{x}_i, \mathbf{x}_j) = \vartheta_0^2 \exp\Big[-\frac{1}{2}\sum_{d=1}^{D}\Big(\frac{x_{id} - x_{jd}}{l_d}\Big)^2\Big], \tag{3}$$

where x_{id} and x_{jd} correspond to d-th element of vectors $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively, and $\Theta = [\vartheta_0, l_1, \ldots, l_D]^T$ denotes the hyperparameters. In particular, ϑ_0^2 quantifies the variation of the underlying latent function from its mean; and l_d represents the characteristic length scale for each input dimension. Effectively, l_d determines the relative importance of each input variable in estimating the target output such that a smaller value of l_d implies that the corresponding input dimension has more impact on the output and is thus more relevant.

We incorporate the additive noise term in (1) into the aforementioned kernel function as follows:

$$k(\mathbf{x}_i, \mathbf{x}_j) = k_s(\mathbf{x}_i, \mathbf{x}_j) + \sigma_n^2 \delta_{ij}, \tag{4}$$

where δ_{ij} denotes the Kronecker delta, which takes value 1 if $i = j$ and 0 otherwise. Then, the distribution of $\mathbf{y}$, given the latent function values $\mathbf{f}$ and the input $\mathbf{X}$, is given by

$$p(\mathbf{y}|\mathbf{f}, \mathbf{X}) = \mathcal{N}(\mathbf{f}, \sigma_n^2 \mathbf{I}), \tag{5}$$

where $\mathbf{I}$ is an $N \times N$ identity matrix. By using (2) and (5), the marginal distribution of $\mathbf{y}$ can be found to be

$$p(\mathbf{y}|\mathbf{X}) = \int p(\mathbf{y}|\mathbf{f}, \mathbf{X}) p(\mathbf{f}|\mathbf{X}) d\mathbf{f} = \mathcal{N}(\mathbf{0}, \mathbf{K} + \sigma_n^2 \mathbf{I}). \tag{6}$$

Then, the marginal log-likelihood of $\mathbf{y}$ can be written as

$$\log p(\mathbf{y}|\mathbf{X}, \Theta) = -\frac{1}{2}\mathbf{y}^T(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y} - \frac{1}{2}\log|\mathbf{K} + \sigma_n^2\mathbf{I}| - \frac{N}{2}\log 2\pi, \tag{7}$$

where $|.|$ is the determinant of a matrix. The hyperparameters are optimized by maximizing the marginal log-likelihood function in (7). Therefore, we calculate the gradient of (7) with respect to the hyperparameters in the following:

$$\frac{\partial \log p(\mathbf{y}|\mathbf{X}, \Theta)}{\partial \theta_i} = -\frac{1}{2}\mathrm{tr}\Big((\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\frac{\partial(\mathbf{K} + \sigma_n^2\mathbf{I})}{\partial \theta_i}\Big) + \frac{1}{2}\mathbf{y}^T(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\frac{\partial(\mathbf{K} + \sigma_n^2\mathbf{I})}{\partial \theta_i}(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y}. \tag{8}$$

It should be noted that computation of the marginal log-likelihood function and its gradient involves an inversion of a matrix, $\mathbf{K} + \sigma_n^2\mathbf{I}$, which requires a computational time of $O(N^3)$. Thus, a simple implementation of the GPR is suitable for data sets with up to few thousands training examples. For larger data sets, sparse approximations to regular GPR based on choosing a small representative subset of training samples can be efficiently applied [30].

The characterization in (8) allows the use of any gradient-based optimization method to optimize the marginal log-likelihood function (7). Note that the objective function is in general a nonconvex function with respect to the hyperparamaters, and hence the gradient-based method may converge to a local optimum. In order to overcome this problem, gradient-based optimization can be performed with different initial points, and the optimal hyperparameters that yield the largest marginal loglikelihood can be chosen. After determining the optimal hyperparameters, we express the joint distribution of

the training outputs $\mathbf{y}$ and the test output y_* as

$$p(\mathbf{y}, y_*|\mathbf{X}, \mathbf{x}_*, \Theta) = \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ 0 \end{bmatrix}, \begin{bmatrix} \mathbf{K} + \sigma_n^2 \mathbf{I} & \mathbf{k}_* \\ \mathbf{k}_*^T & k_{**} + \sigma_n^2 \end{bmatrix} \right), \tag{9}$$

where $\mathbf{k}_* = [k(\mathbf{x}_1, \mathbf{x}_*), \ldots, k(\mathbf{x}_N, \mathbf{x}_*)]^T$ and $k_{**} = k_s(\mathbf{x}_*, \mathbf{x}_*)$. The main goal of the GPR is to find the predictive distribution of the test output y_* conditioned on the dataset $\mathcal{D}$ and test input $\mathbf{x}_*$. Hence, by marginalizing the joint distribution (9) over the training dataset output $\mathbf{y}$, we obtain that the predictive distribution of the test output, y_*, is Gaussian distributed, i.e.,

$$p(y_*|\mathbf{X}, \mathbf{y}, \mathbf{x}_*, \Theta) = \mathcal{N}(\mu_*, \Sigma_*), \tag{10}$$

where the mean and the covariance of the predictive distribution are given, respectively, in the following:

$$\mu_* = \mathbf{k}_*^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{y} \tag{11}$$

$$\Sigma_* = \sigma_n^2 + k_{**} - \mathbf{k}_*^T (\mathbf{K} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{k}_*. \tag{12}$$

We point out that the mean of the predictive distribution μ_* in (11), which is effectively the point estimate of the test output, is obtained as a linear combination of the noisy dataset outputs $\mathbf{y}$. Also, the variance of the predictive distribution, Σ_* in (12) serves as a measure of the uncertainty in the estimate of the test output. Once the inversion of a matrix $\mathbf{K} + \sigma_n^2 \mathbf{I}$ is done, the computational complexity of the testing stage is $O(N^2)$ (and $O(N)$ for only the point estimate), which makes the proposed method highly suitable for online operation. We note that regular GPR is most suitable for training datasets whose size N is up to the order of ~ 1000. For larger datasets, a sparse GPR is more suitable as it significantly reduces the computational complexity at the expense of often negligible performance degradation [28]. Overall, given that $N \sim 1000$, the memory and processing power required for implementation of the regular GPR is relatively easily met with embedded processors widely available on the market (e.g., [31]).

III. SoC Estimation Method Based on Regular GPR

In this section, we present an SoC estimation method based on regular GPR. The SoC of the battery is defined as the ratio of the amount of energy presently stored in the battery to its maximum capacity [4]. In particular, a fully discharged battery has an SoC of 0% and SoC increases while the battery is being charged. Consequently, a fully charged battery reaches 100% SoC.

Fig. 1: SoC estimation using regular GPR.

Fig. 1 shows the structure of the regular GPR based estimator where the inputs are the voltage, current and temperature of the battery at the sampling time k, denoted by $V(k)$, $I(k)$ and $T(k)$, respectively, and the output is the estimated SoC at the sampling time k, represented by $\text{SoC}_*(k)$. In particular, the proposed method consists of two parts. In the first part, GPR is trained offline to learn the relationship between the voltage, current, temperature and SoC of the battery. Then, optimal values of the hyperparameters of the chosen kernel function are determined through conjugate gradient method based on a training data set, $\mathcal{D} = (\{\mathbf{x}_n\}_{n=1}^N, \mathbf{y})$ where $\mathbf{x}_n$ includes the voltage, current and temperature of the battery and $\mathbf{y}$ contains the corresponding normalized SoC values. It should be noted that SoC values in the training data set are first normalized to have zero mean by subtracting their sample mean. In the second part, online SoC estimation of the battery is performed based on voltage, current and temperature measurements of the battery. In particular, the mean of the predictive distribution in (11) represents the SoC point estimate. In order to represent the uncertainty in the estimates, the $100(1-\alpha)\%$ confidence interval is computed by using (11) and (12) as follows:

$$[\mu_* - z_{(1-\alpha)/2} \Sigma_*^{1/2}, \mu_* + z_{(1-\alpha)/2} \Sigma_*^{1/2}], \tag{13}$$

where $\alpha \in [0, 1]$ represents the confidence level and $z_{(1-\alpha)/2}$ is the critical value of the standard normal distribution. Intuitively, the confidence interval provides a range of values which is likely to contain the true value of the test output. As the variance of the predictive distribution decreases, the confidence interval gets narrower, which indicates a more accurate estimate. From the practical point of view, the confidence interval accompanying an SoC point estimate can be used by a human expert or automatically in the BMS for the decision making process, the specifics of which depend on the application at hand. In addition to the battery monitoring, the confidence interval can be utilized in a general control framework to quantify process noises and perform stochastic control designs [32], where without uncertainty quantification, control design typically resorts to worst case design, resulting in conservativeness. The entire process is described in more detail in Algorithm 1.

IV. SoC Estimation Method Based on Recurrent GPR

In this section, we propose a new SoC estimation method based on recurrent GPR. In comparison to the method presented in Section III, the previously estimated SoC value at time $k-1$ is fed back to the input so that the previous SoC, together with the measurements of the voltage, current and temperature, are used to estimate the SoC at time k.

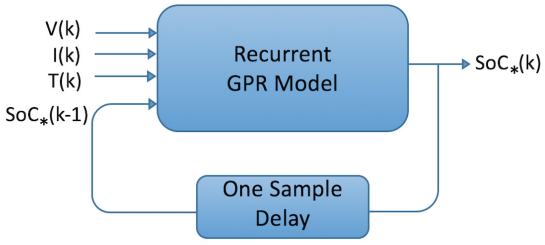

Fig. 2: SoC estimation using recurrent GPR.

Intuitively, the SoC value cannot abruptly change in a short time interval. This fact can be utilized in closed-loop form based on a feedback mechanism. Therefore, we implement

Algorithm 1 The flow chart of an SoC estimation method using regular GPR

1: **Traning part:**
2: **Step 1:** Obtain a training data set, $\mathcal{D} = (\mathbf{X}, \mathbf{y})$, where $\mathbf{X}$ are voltage, current and temperature measurements of the battery, and $\mathbf{y}$ are the corresponding SoC values.
3: **Step 2:** Initialize hyperparameters, Θ,
4: **Step 3:** Apply conjugate gradient method to find the optimal values of the hyperparameters that minimize the negative marginal log-likelihood function (equivalently maximize the marginal log-likelihood function).
5: **Estimation part:**
6: **Step 4:** Obtain the mean and variance of the predictive distribution given optimal hyperparameters, training dataset, $\mathcal{D}$, and test input $\mathbf{x}_*$ (i.e., present voltage, current and temperature measurement of the battery) as follows:

$$\mu_* = \mathbf{k}_*^T(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y}$$
$$\Sigma_* = \sigma_n^2 + k_{**} - \mathbf{k}_*^T(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{k}_*,$$

where μ_* is the SoC estimate.

a feedback loop with one and two-tap delays, respectively in the GPR model to improve the estimation accuracy. The proposed method with a single-tap delay in the feedback loop is illustrated in Fig. 2, where $\text{SoC}_*(k-1)$ denotes the SoC output of the recurrent GPR model at sampling time $k-1$. In the case of feedback loop with two-tap delays, the SoC outputs at the sampling times $k-1$ and $k-2$ are fed back and included in the input vector of the proposed method.

Algorithm 1 can be modified for SoC estimation based on recurrent GPR with a single-tap feedback loop in such a way that the SoC estimate at the previous sample is used as an additional entry in the input vector and hence, the kernel function requires additional hyperparameters associated with this entry. For instance, the SE kernel now has an extra hyperparameter, l_4 which denotes the characteristic length scale for the previous SoC estimate.

V. SoC Estimation Method Based on Autoregressive Recurrent GPR

In this section, we present an SoC estimation method using autoregressive recurrent GPR to further smooth out the SoC estimates.

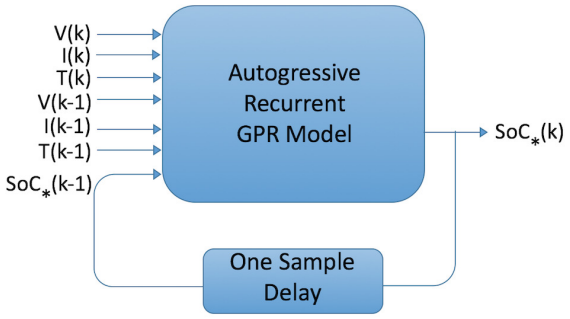

Fig. 3: SoC estimation using autoregressive recurrent GPR.

The autoregressive recurrent GPR takes into account the impact of the present and past values of voltage, current and temperature as well as the previous SoC on the estimation of the present SoC as shown in Fig. 3. Hence, the proposed method improves the estimation accuracy by eliminating abrupt changes in the SoC estimates.

VI. Results and Discussion

In this section, we validate the proposed SoC estimation methods using both simulated and experimental data. We choose the root mean square error (RMSE) and maximum absolute error (MAE) as the main performance metrics, which are respectively defined as follows

$$\text{RMSE} = \sqrt{\frac{1}{N_t}\sum_{i=1}^{N_t}(y_{*,i}^{\text{true}} - \hat{y}_{*,i}^{\text{est}})^2} \tag{14}$$

$$\text{MAE} = \max_{i=1,\ldots,N_t} |y_{*,i}^{\text{true}} - \hat{y}_{*,i}^{\text{est}}|, \tag{15}$$

where N_t denotes the size of test data, $\mathbf{y}_*^{\text{true}}$ is a $1 \times N_t$ vector including SoC values of the test data and $\hat{\mathbf{y}}_*^{\text{est}}$ is a $1 \times N_t$ vector containing the estimated SoC values. In addition, we also report the average computational time per data point of all considered methods, measured by invoking `tic` and `toc` routines in the `Matlab` implementation on a 4 GHz Intel Core i5-6400 CPU computer. In general, using PC for computational complexity evaluation provides a general understanding of algorithms' computational requirements. However, this study also has a practical significance for the scenarios where processing power equivalent to or even larger than a typical PC is available. For example, to enable operation of an autonomous vehicles, strong onboard processing power is needed to support execution of a variety of algorithms [33]. As yet another example, we emphasize an emerging paradigm of cloud-based computations, where cloud-based BMS is already becoming a reality [34].

In the following subsections, we provide simulation and experimental results to illustrate the performance of the proposed methods. We compare our methods with three data-driven methods used for battery SoC estimation, namely, the Support Vector Machine (SVM), Relevance Vector Machine (RVM) and Neural Networks (NN) [35]. We emphasize that all these methods are fully optimized for each dataset used.

In particular, the SVM is tested using linear, Gaussian and polynomial kernel of different orders, with and without data standardization. The RVM [36] is tested using Gaussian kernel, polynomial and homogeneous polynomial kernels of different orders and length scales, spline kernel over different length scales, as well as Cauchy, cubic, R, thin plate spline (TPS) and Laplace kernels. Finally, the NN employs a single hidden layer because of a relatively small amount of training data in each test, which precludes using a multilayer structure. However, the NN is optimized with respect to the number of neurons. In the following parts, we report the optimal performance results of the SVM, RVM and NN achieved in each considered dataset. As an aside remark, our proposed methods are not

optimized with respect to the kernel function, and in all cases the squared exponential kernel (2) is utilized.

A. Simulation Results

1) Simulation Dataset: The simulated battery data was generated by using a battery model based on equivalent circuit model including thermal equation. In particular, the simulated model consists of two RC circuits ($R_{d1} = 0.716\,\mathrm{m}\Omega$, $C_{d1} = 2678.49\,\mathrm{F}$, $R_{d2} = 0.08\,\mathrm{m}\Omega$, $C_{d2} = 2678.49\,\mathrm{F}$) and one resistance ($R_{sct} = 2.037\,\mathrm{m}\Omega$), which are connected in series [37]. The initial temperature was $10°\mathrm{C}$ and the initial SoC was set to 50%.

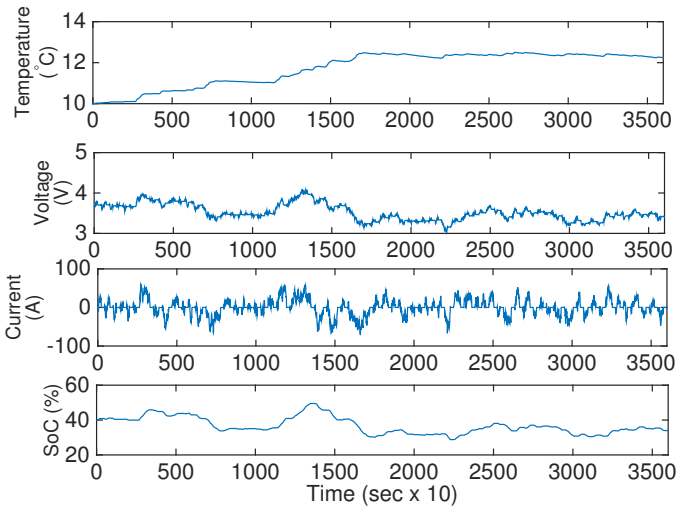

Fig. 4: Simulation dataset: temperature, voltage, current and battery SoC vs. time.

In Fig. 4, we plot the temperature, voltage, current and SoC values as a function of time. The first 2000 samples are used as training data to find the optimal values of the hyperparameters while the remaining 1600 samples are used as testing data to verify the performance of the proposed methods. We initialize all hyperparameters with ones.

2) Performance Results:

a) SoC Estimation Method Based on Regular GPR: In this case, we analyze and display the performance of the SoC estimation method using the regular GPR in Fig. 5(a). The shaded blue area represents the 95% confidence interval. The RMSE and MAE values are given in Table I. We achieve a relatively good accuracy with RMSE = 0.3808%, MAE = 1.8185%. We also show the difference between the actual SoC and the estimated SoC (i.e., estimation error) vs. time in Fig. 5(f). It is observed that when the estimation error is higher, the estimation uncertainty is higher, and consequently, the confidence interval is wider. On the other hand, accurate SoC estimates result in lower uncertainty, and thus narrower confidence intervals. This uncertainty characterization is one of the key advantages of the GPR-based methods over non-probabilistic machine learning methods such as the SVM, RVM, or NN.

Another important remark is that the optimal hyperparameters associated with each input variable enable us to infer the relative importance of the inputs. The optimal values of the characteristic length scales of voltage, current and temperature are 1.1025, 278.27 and 109.76, respectively. Note that smaller values of the characteristic length scales imply that the corresponding input dimension is more important and relevant, thus we can conclude that voltage has more impact than temperature, and temperature has more impact than current on the SoC value.

TABLE I: RMSE and MAE performance of estimators using the simulation dataset.

Methods	**RMSE (%)**	**MAE (%)**	**Computational time (seconds)**
Regular GPR	0.3808	1.8185	8.541e-02
Recurrent GPR with one-tap delay	0.2692	1.3133	8.512e-02
Recurrent GPR with two-tap delays	0.2042	0.8140	8.487e-02
Autoregressive Recurrent GPR	0.1382	0.3536	8.003e-02
Support Vector Machine	0.5337	2.1833	2.886e-04
Relevance Vector Machine	0.7904	6.0401	3.826e-05
Neural Network	0.3713	1.8788	3.589e-03

b) SoC Estimation Method Based on Recurrent GPR: In this case, we evaluate the performance of the SoC estimation method using the recurrent GPR. Along with the actual SoC, the SoC estimated using the recurrent GPR with one and two-tap delays are shown in Fig. 5(b) and 5(c), respectively. It is seen that the recurrent GPR gives better estimation performance (namely, RMSE = 0.2692% and MAE = 1.3133% in the case of one-tap delay in the feedback loop and RMSE = 0.2042% and MAE = 0.8140% in the case of two-tap delay in the feedback loop) than the regular GPR, because the recurrent GPR incorporates into the model the impact of the previous SoC on the present SoC. As can be noted, using previous two SoC estimates, i.e., two taps in the feedback loop, reduces the RMSE and MAE achieved with a single tap by 24% and 38%, respectively. Increasing the number of taps in the feedback loop beyond two leads to only a slight improvement in the performance.

The optimal characteristic length scales of the voltage, current, temperature and previous SoC are 3.78, 507.19, 58.34 and 1.48, respectively, which further validates our intuition about the importance of the previous SoC value.

In addition, we analyze the impact of an inaccurate initial SoC value on the performance of the SoC estimation method based on recurrent GPR. In Fig. 5(c), we display the estimation performance of the proposed method with two different initial SoC values, i.e., 40% and 70%. As can be observed, the proposed method is robust against uncertainty in the initial conditions. As a remark, in all our tests, the initial SoC estimate(s) needed for recurrent and autoregressive GPR (discussed in the sequel) are obtained from the regular GPR.

c) SoC Estimation Method Based on Autoregressive Recurrent GPR: Now, we analyze the performance of SoC estimation method using the autoregressive recurrent GPR.

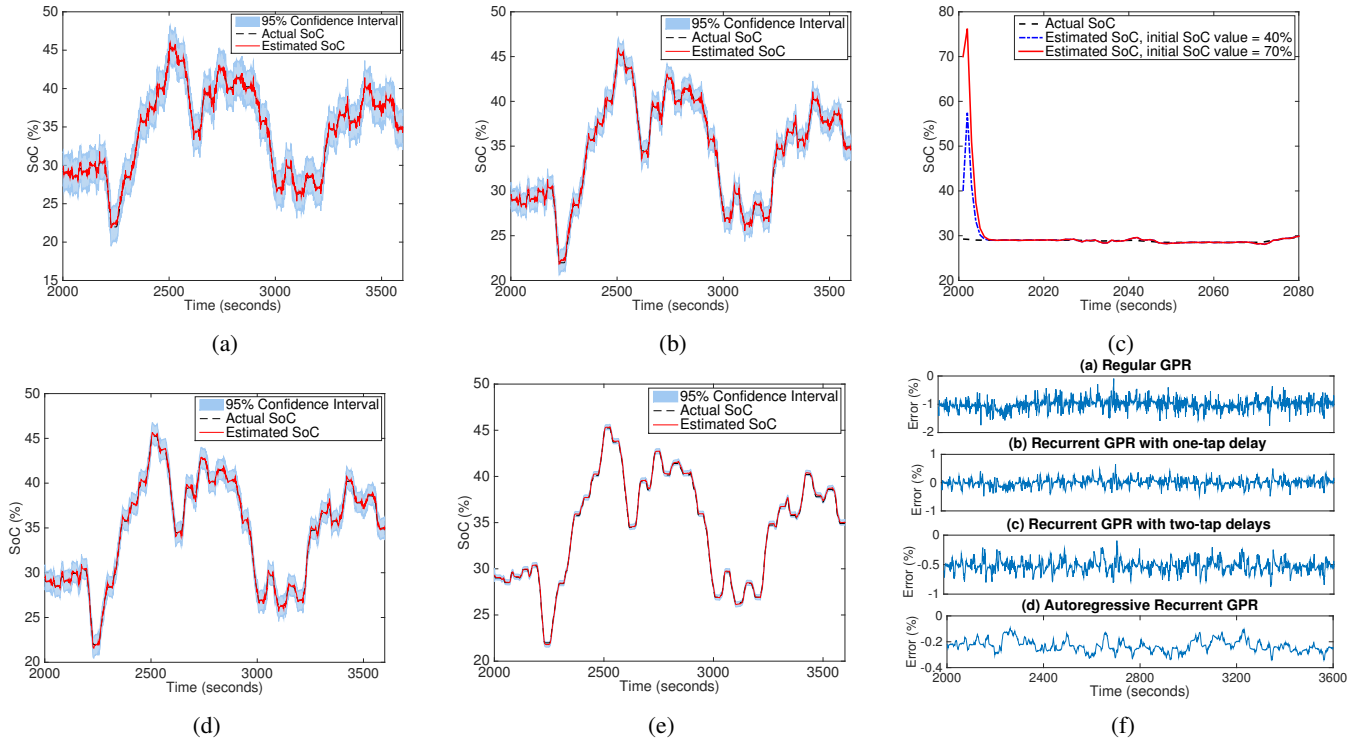

Fig. 5: SoC estimation results using (a) regular GPR, (b) recurrent GPR with one-tap delay, (c) recurrent GPR with one-tap delay with different initial SoC values, (d) recurrent GPR with two-tap delays, (e) autoregressive recurrent GPR with one-tap delay, (f) difference between the actual SoC and the estimated SoC of the proposed methods.

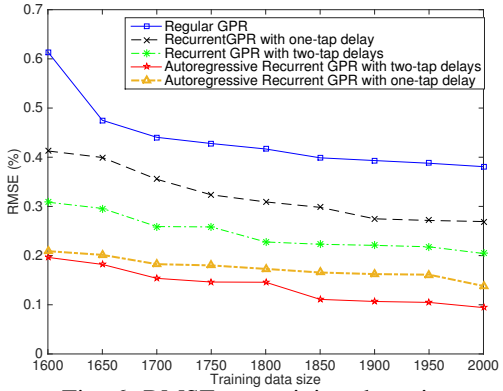

Fig. 6: RMSE vs. training data size.

In Fig. 5(d), we display the SoC estimation result based on the autoregressive recurrent GPR with one-tap delay. The corresponding RMSE and MAE values, and computational times are listed in Table I.

It is seen that the autoregressive recurrent GPR effectively smooths out the SoC estimates and hence leads to significant improvements in the estimation accuracy. In particular, the RMSE and MAE are reduced to the levels below 0.14% and 0.36%, respectively, which is an almost perfect fit to the actual SoC.

In Fig. 6, we plot RMSE as a function of training data size to illustrate how training data size affects the estimation accuracy of the proposed methods. After the training stage, the SoC methods are tested on the remaining data points and the corresponding RMSE is evaluated and reported in Fig. 6. It is seen that including more training data reduces the RMSE and improves the estimation performance since more training samples provides additional information, and hence the GPR model is trained better. Increasing the training data size beyond a certain level does not provide a substantial improvement in the estimation performance.

d) Comparison of the Proposed Methods with the State-of-the-art Techniques: We compare the performances of the proposed methods with the SoC estimation methods based on SVM, RVM and NN. The corresponding RMSE, MAE values, and computational times are listed in Table I. As already pointed out, we report here the best results achieved on this particular dataset. As such, the best SVM performance is achieved with the 2nd order polynomial kernel, the best RVM performance is attained with the 9th order polynomial kernel with the length scale 7.25, while the optimal NN algorithm employs 3 neurons. As can be seen, the recurrent GPR and autoregressive recurrent GPR outperform SVM, RVM and NN in terms of RMSE and MAE. The regular GPR achieves better estimation accuracy than SVM and RVM, while it has a comparable performance as the NN. Finally, even though the average computational time per data point of the proposed methods is higher than that of the SVM, RVM and NN, it is still low, making them suitable for online implementation.

B. Experimental Results

In this section, we present experimental results for SoC estimation performance of the proposed methods under different operating conditions of the battery, i.e., constant and dynamic charging/discharging.

1) Constant Charge−Discharge Test: The experimental dataset for this test was collected from a $LiMn_2O_4$/hard-carbon

battery with a nominal capacity of 4.93 Ah in the Advanced Technology R&D Center, Mitsubishi Electric Corporation.

In Fig. 7(a), we display the experimental dataset, including voltage, current, temperature and SoC of the battery. In particular, five consecutive cycles of charging and discharging at constant 10 C-rate/h were performed using a rechargeable battery test equipment produced by Fujitsu Telecom Networks. The charging and discharging currents are measured in terms of C-rate/h. In this case, 10 C-rate/h corresponds to a current of 49.3 A and the battery will deliver its rated capacity for 6 minutes. The reference (true) SoC was obtained using the Coulomb counting method. The sampling period was chosen to be 1 second. The first 1050 samples are used as training data while the remaining 950 samples are used as testing data.

In Fig. 7(b)-(e), we display the actual SoC and the estimated SoC values attained with the proposed methods. The resulting RMSE and MAE values are listed in Table II.

TABLE II: RMSE and MAE performance of estimators using the experimental constant charging/discharging test dataset.

Methods	**RMSE (%)**	**MAE (%)**	**Computational time (seconds)**
Regular GPR	1.5383	4.8503	2.323e-02
Recurrent GPR with one-tap delay	0.5482	1.3872	2.141e-02
Recurrent GPR with two-tap delays	0.4829	1.2758	2.131e-02
Autoregressive Recurrent GPR	0.2762	0.5491	2.152e-02
Support Vector Machine	1.7567	5.6604	2.968e-04
Relevance Vector Machine	14.4304	18.8209	3.290e-05
Neural Network	0.7069	3.0042	3.676e-03

As can be observed, the recurrent GPR with one-tap delay in the feedback loop outperforms the regular GPR, improving the RMSE and MAE performance by around 64%, and 71%, respectively. In [21], the authors applied the SVM to estimate the SoC of a GBS Energy $LiFeMnPO_4$ battery cell with a nominal capacity of 60 Ah under a constant current profile. In particular, three consecutive cycles of charging at 0.3 C-rate/h and discharging at 0.33 C-rate/h were performed and the experimental results confirmed that MAE was 6% and RMSE 0.71%. It is seen that our proposed method based on recurrent GPR achieves significantly better performance with RMSE around 0.55% and MAE around 1.4% under a constant current profile. The recurrent GPR with two taps in the feedback loop further improves the performance, leading to the overall reduction in the RMSE and MAE by around 69% and 74%, respectively. Finally, autoregressive recurrent GPR leads to RMSE and MAE below 0.28% and 0.55%, respectively.

As seen from Table II, the recurrent GPR and autoregressive recurrent GPR give better estimation performance compared to the SVM, RVM, and NN. As before, the SVM, RVM, and NN are fully optimized for this dataset such that the optimal SVM uses 2nd order polynomial kernel, the optimal RVM employs 2nd order polynomial kernel with kernel length scale 10, while the optimal NN contains 6 neurons. The regular GPR outperforms both SVM and RVM while NN leads to lower estimation error compared to the regular GPR with SE covariance function. The authors in [27] used the same dataset and considered the impact of different covariance functions on the performance of the regular GPR. It was shown that the regular GPR with quasi-periodic kernel function leads to better estimation accuracy with RMSE = 1.0648% and MAE = 2.7701%, which outperforms NN in terms of MAE. The challenging aspect in that dataset is to estimate SoC over time intervals immediately following the time instants at which the current abruptly changes. A notable example is the accuracy achieved with the RVM, where it is observed that even fully optimized RVM fails to find accurate SoC estimates (i.e., RMSE = 14.4304% and MAE = 18.8209%), as shown in Table II. As in the previous case, the proposed methods have a low computational time and admit online implementation.

2) Dynamic Charge Discharge Test: In this section, we analyze the impact of dynamic operating conditions on the estimation performance. In particular, we consider experimental datasets obtained from a battery employed in two standardized driving conditions: Federal urban driving schedule (FUDS), corresponding to urban driving, and the US06 highway driving schedule, corresponding to driving with high acceleration and rapid speed fluctuations. The experimental dataset as shown in Fig. 8(a)-(b), was obtained using INR 18650-20R battery with a rated capacity 2 Ah [38]. The initial SoC was 80% and the initial temperature was 25°C. The first 3000 samples of FUDS test are employed for training the GPR models. We test the proposed methods using the first 3000 samples of US06 driving schedule. The same approach of training an SoC model parameters using one dataset and testing the SoC estimation method using the other dataset is also reported in [38].

In Fig. 8(c)-(e), we show the actual SoC and the estimated SoC values of the proposed methods. We list the resulting RMSE and MAE values of the proposed methods and the optimized state-of-the-art techniques in Table III. In particular, the optimal SVM uses 2nd order polynomial kernel, the optimal RVM employs 17th order polynomial kernel with the length scale 2.25, while the optimal NN contains 6 neurons. As can be seen from the table, the SoC estimation method based on autoregressive recurrent GPR gives the best performance among all the methods.

Finally, the computational time per data point of the proposed methods depends on the size of the training data, which is not surprising given that whole training data, along with the hyperparameters, is utilized in the testing stage. However, with the training data size of 3000, used in the last dataset, the computation time per data point is still well within 1 second, which enables online implementation of the proposed methods. As a general remark, if the training data size becomes exceedingly large, the framework of sparse GPR can be used to considerably reduce the computational time, at the expense of an insignificant performance deterioration, as has been demonstrated in [28].

Finally, we note that in all considered cases, the 95% confidence interval contains the true SoC values, which validates its

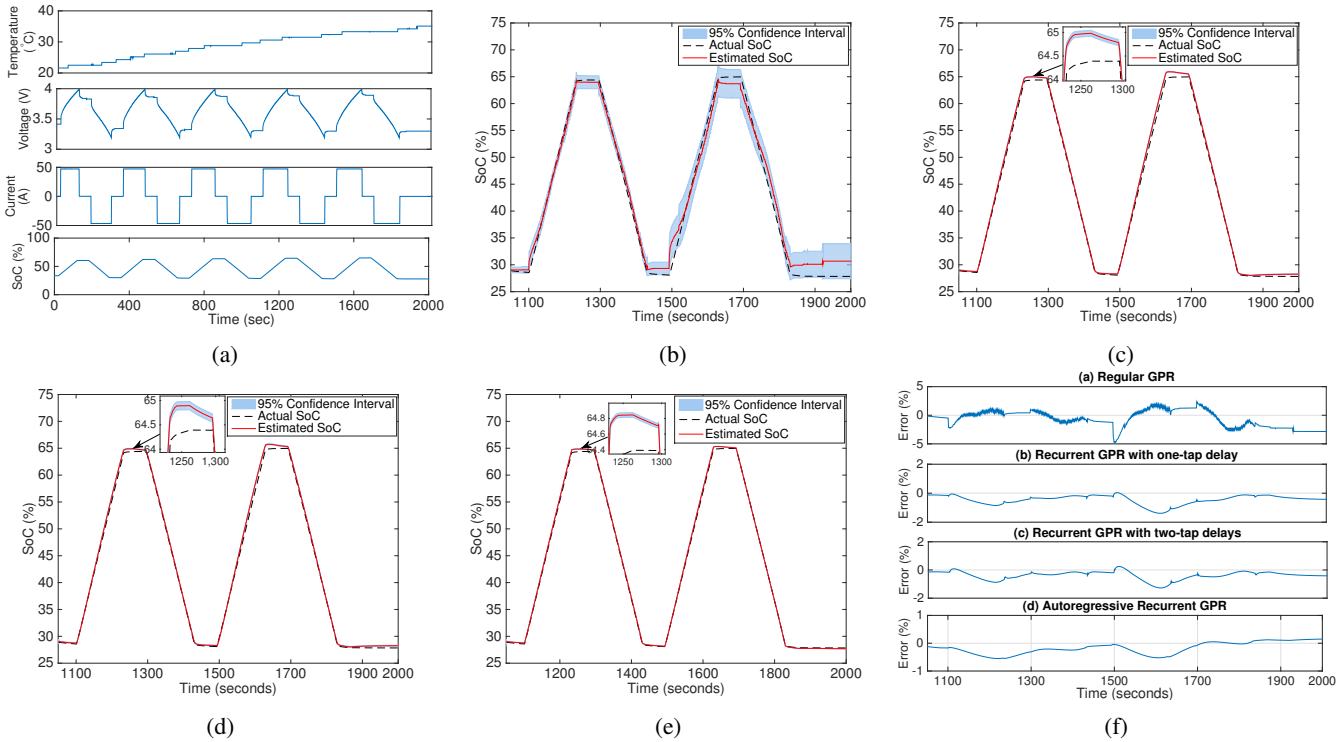

Fig. 7: (a) Experimental dataset of constant discharge/charge test: temperature, voltage, current and SoC of the battery vs. time, SoC estimation results using (b) regular GPR, (c) recurrent GPR with one-tap delay in the feedback loop, (d) recurrent GPR with two-tap delays in the feedback loop, (e) autoregressive recurrent GPR with one-tap delay in the feedback loop, (f) difference between the actual SoC and the estimated SoC of the proposed methods.

accuracy. Also, the confidence interval in all cases is relatively narrow, which indicates its usefulness.

TABLE III: RMSE and MAE performance of estimators using the experimental dynamic charging/discharging test dataset.

Methods	**RMSE (%)**	**MAE (%)**	**Computational time (seconds)**
Regular GPR	0.760	2.250	2.137e-01
Recurrent GPR with one-tap delay	0.2853	0.8177	6.574e-01
Recurrent GPR with two-tap delays	0.630	0.820	6.484e-01
Autoregressive Recurrent GPR	0.2419	0.8119	6.542e-01
Support Vector Machine	0.7111	2.0562	3.170e-06
Relevance Vector Machine	0.7194	2.0499	3.629e-05
Neural Network	0.7643	2.3437	3.411e-03

VII. Conclusion

In this paper, we have proposed novel SoC estimation methods based on regular/recurrent GPR. The inputs to the regular GPR-based method are voltage, current and temperature of the battery while the previous SoC estimate is used as an additional input for the recurrent method. The GPR model is trained offline where the optimal hyperparameters of the kernel function are determined. After the training stage, the SoC is estimated online. Both experimental and simulation results validate high accuracy of the proposed methods. We observe that the recurrent GPR gives better estimation performance than the regular GPR. Also, we further improved the estimation performance by applying the autoregressive recurrent GPR, which exploits the present and past values of voltage, current and temperature measurements together with the previous SoC estimate. As such, we achieved RMSE below 0.14% and MAE below 0.36% under a dynamic charging-discharging condition.

The proposed methods provide confidence intervals of the obtained point estimates, a feature stemming from the fact that the GPR infers the probability distribution of the unknown SoC value. Another important property of the proposed methods is their ability to identify the relative importance of the inputs in estimating the battery SoC.

We have further compared the performance of the proposed methods with the state-of-the-art techniques including fully optimized SVM, RVM, and NN. It is shown that the autoregressive recurrent GPR gives the best estimation performance. Finally, we have measured that the compute time per data point of the proposed methods is much less than the duration of 1 second, which admits their online implementation.

We believe that the SoC estimation based on the GPR models is a stepping stone for further progress in this domain. With that in mind, our future work will address the impact of battery aging by combining the proposed methods with the battery capacity degradation, which is the main indicator of the battery SoH.

References

[1] V. Ramadesigan, P. W. C. Northrop, S. De, S. Santhanagopalan, R. D. Braatz, and V. R. Subramanian, "Modeling and simulation of lithium-

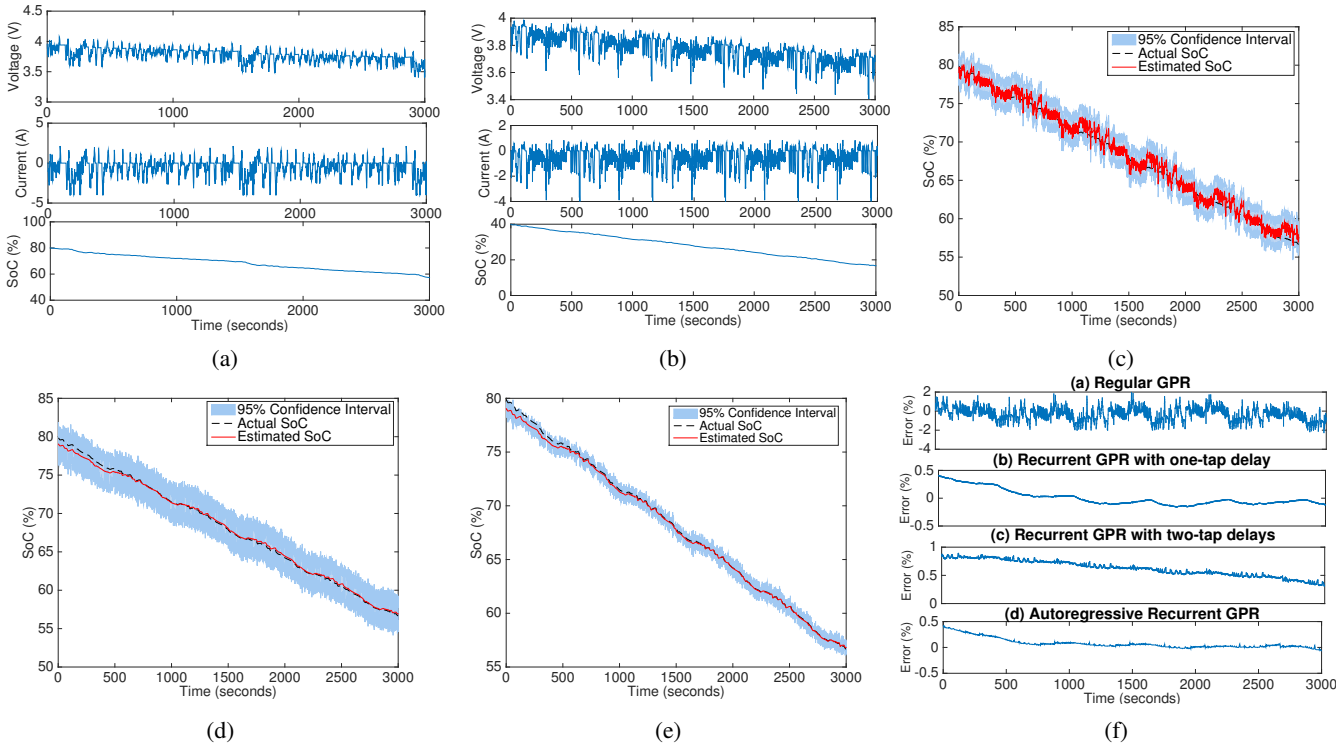

Fig. 8: Experimental dataset: voltage, current and SoC of the battery vs. time for (a) FUDS, (b) US06 driving schedule, SoC estimation results using (c) regular GPR, (d) recurrent GPR with one-tap delay in the feedback loop, (e) autoregressive recurrent GPR with one-tap delay in the feedback loop, (f) difference between the actual SoC and the estimated SoC of the proposed methods.

ion batteries from a systems engineering perspective," *Journal of the Electrochem. Soc.*, vol. 159, no. 3, pp. R31–R45, 2012.

[2] J. Chiasson and B. Vairamohan, "Estimating the state of charge of a battery," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 3, pp. 465–470, May. 2005.

[3] K. S. Ng, C. S. Moo, Y. P. .Chen, and Y. C. Hsieh, "State-of-charge estimation for lead-acid batteries based on dynamic open circuit voltage," in *Proc. IEEE Inter. Power and Energy Conf. (PECon)*, pp. 972–976, Dec. 2008.

[4] S. Piller, M. Perrin, and A. Jossen, "Methods for state-of-charge determination and their applications," *Journal of Power Sources*, vol. 96, no. 1, pp. 113–120, Jun. 2001.

[5] K. S. Ng, C. S. Moo, Y. P. .Chen, and Y. C. Hsieh, "Enhanced Coulomb counting method for estimating state-of-charge and state-of-health of lithium-ion batteries," *Applied Energy*, vol. 86, no. 9, pp. 1506–151, Sep. 2009.

[6] M. W. Yatsui and B. Hua, "Kalman filter based state-of-charge estimation for lithium-ion batteries in hybrid electric vehicles using pulse charging," in *Proc. IEEE Vehicle Power and Propulsion Conf. (VPPC)*, pp. 1–5, Sep. 2011.

[7] Z. Chen, Y. Fu, and C. C. Mi, "State of charge estimation of Lithium-ion batteries in electric drive vehicles using extended Kalman filtering," *IEEE Trans. Veh. Technol.*, vol. 62, no. 3, pp. 1020–1030, Mar. 2013.

[8] H. He, R. Xiong, X. Zhang, F. Sun, and J. Fan, "State-of-charge estimation of the Lithium-ion battery using an adaptive extended Kalman filter based on an improved Thevenin model," *IEEE Trans. Veh. Technol.*, vol. 60, no. 4, pp. 1461–1469, May. 2011.

[9] H. Fang, Y. Wang, Z. Sahinoglu, T. Wada, and S. Hara, "State of charge estimation for Lithium-ion batteries: an adaptive approach," *Control Eng. Pract.*, vol. 25, pp. 45–54, Apr. 2014.

[10] A. E. Mejdoubi, A. Oukaour, H. Chaoui, H. Gualous, J. Sabor, and Y. Slamani, "State-of-charge and state-of-health Lithium-ion batteries' diagnosis according to surface temperature variation," *IEEE Trans. Ind. Electron.*, vol. 63, no. 4, pp. 2391–2402, Apr. 2016.

[11] W. He, N. Williard, C. Chen, and M. Pecht, "State of charge estimation for electric vehicle batteries using unscented kalman filtering," *Microelectronics Reliability*, vol. 53, no. 6, pp. 840–847, Jun. 2013.

[12] S. Santhanagopalan and R. E. White, "State of charge estimation using an unscented filter for high power lithium ion cells," *Int. Journal of Energy Research*, vol. 34, no. 2, pp. 152–163, 2010.

[13] C. Zhang, L. Y. Wang, X. Li, W. Chen, G. G. Yin, and J. Jiang, "Robust and adaptive estimation of state of charge for Lithium-ion batteries," *IEEE Trans. Ind. Electron.*, vol. 62, no. 8, pp. 4948–4957, Aug. 2015.

[14] W. He, N. Williard, C. Chen, and M. Pecht, "State of charge estimation for Li-ion batteries using neural network modeling and unscented Kalman filter-based error cancellation," *Int. Journal of Electrical Power & Energy Systems*, vol. 62, pp. 783–791, Nov. 2014.

[15] W. Y. Chang, "Estimation of the state of charge for a LFP battery using a hybrid method that combines a RBF neural network, an OLS algorithm and AGA," *Int. Journal of Electrical Power & Energy Systems*, vol. 53, pp. 603–611, Dec. 2013.

[16] Y. S. Lee, W.-Y. Wang, and T. Y. Kuo, "Soft computing for battery state-of-charge (BSOC) estimation in battery string systems," *IEEE Trans. Ind. Electron.*, vol. 55, no. 1, pp. 229–239, Jan. 2008.

[17] C. Bo, B. Zhifeng, and C. Binggang, "State of charge estimation based on evolutionary neural network," *Journal of Power Sources*, vol. 49, no. 10, pp. 2788–2794, Oct. 2008.

[18] M. Charkhgard and M. Farrokhi, "State-of-charge estimation for Lithium-ion batteriesusing neural networks and EKF," *IEEE Trans. Ind. Electron.*, vol. 57, no. 12, pp. 4178–4187, Dec. 2010.

[19] T. Hansen and C.-J. Wang, "Support vector based battery state of charge estimator," *Journal of Power Sources*, vol. 141, no. 2, pp. 351–358, Mar. 2005.

[20] J. A. Antón, P. J. G. Nieto, F. C. Juez, F. S. Lasheras, M. G. Vega, and M. R. GutiÉrrez, "Battery state-of-charge estimator using the SVM technique," *Applied Mathematical Modeling*, vol. 37, no. 9, pp. 6244–6253, May. 2013.

[21] J. A. Antón, P. J. G. Nieto, C. B. Viejo, and J. A. V. Vilán, "Support vector machines used to estimate the battery state of charge," *IEEE Trans. Power Electron.*, vol. 28, no. 12, pp. 5919–5926, Dec. 2013.

[22] J. N. Hu, J. J. Ju, H. B. Lin, X. P. Li, C. L. Jiang, X. H. Qiu, and W. S. Li, "State-of-charge estimation for battery management system using optimized support vector machine for regression," *Journal of Power Sources*, vol. 269, pp. 682–693, Dec. 2014.

[23] T. Markel, A. Brooker, T. Hendricks, V. Johnson, K. Kelly, B. Kramer, M. O'Keefe, S. Sprik, and K. Wipke, "ADVISOR: a systems analysis tool for advanced vehicle modeling," *Journal of Power Sources*, vol. 110, no. 2, pp. 255–266, Aug. 2012.

[24] H. Sheng and J. Xiao, "Electric vehicle state of charge estimation: Nonlinear correlation and fuzzy support vector machine," *Journal of Power Sources*, vol. 281, pp. 131–137, May. 2015.

[25] D. Liu, J. Pang, J. Zhou, Y. Peng, and M. Pecht, "Prognostics for state of

health estimation of lithium-ion batteries based on combination gaussian process functional regression," *Microelectronics Reliability*, vol. 53, pp. 832–839, Apr. 2013.
[26] F. Li and J. Xu, "A new prognostics method for state of health estimation of lithium-ion batteries based on a mixture of gaussian process models and particle filter," *Microelectronics Reliability*, vol. 55, no. 7, pp. 1035–1045, Jun. 2015.
[27] G. Ozcan, M. Pajovic, Z. Sahinoglu, Y. Wang, P. V. Orlik, and T. Wada, "Online state of charge estimation for lithium-ion batteries using gaussian process regression," in *Proc. of 42nd Annual Conf. of the IEEE Industrial Electronics Society (IECON)*, pp. 998–1003, Oct. 2016.
[28] G. Ozcan, M. Pajovic, Z. Sahinoglu, Y. Wang, P. V. Orlik, and T. Wada, "Online battery state-of-charge estimation based on sparse gaussian process regression," in *Proc. of Power and Energy Society General Meeting (PESGM)*, Jul. 2016.
[29] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.
[30] J. Q.-Candela and C. E. Rasmussen, "A unifying view of sparse approximate Gaussian process regression," *Journal of Machine Learning Research*, vol. 6, pp. 1939–1959, Dec. 2005.
[31] [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/
[32] B. S. Chen and W. Zhang, "Stochastic H_2/H_∞ control with state-dependent noise," *IEEE Trans. Autom. Control*, vol. 49, no. 1, pp. 45–57, Jan. 2004.
[33] J. L. et. al., "Towards fully autonomous driving: Systems and algorithms," *IEEE Intelligent Vehicles Symposium*, pp. 163–168, Jun. 2011.
[34] T. Tanizawa, T. Suzumiya, and K. Ikeda, "Cloud-connected battery management system supporting e-mobility," *FUJITSU Sci. Tech. J.*, vol. 51, no. 4, pp. 27–35, Oct. 2015.
[35] C. M. Bishop, *Pattern Recognition and Machine Learning*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.
[36] M. E. Tipping. Sparsebayes: An efficient matlab implementation of the sparse bayesian modeling algorithm (version 2.0). [Online]. Available: http://www.miketipping.com
[37] G. L. Plett, *Battery Management Systems, Volume I: Battery Modeling*. Artech House, 2015.
[38] L. Xing and M. Pecht. Calce battery group. [Online]. Available: http://www.calce.umd.edu/batteries/data.htm

Gozde O. Sahinoglu received the B.S. degree from Bilkent University, Turkey in 2011, and the Ph.D. degree in Electrical Engineering from Syracuse University in 2016. She has been TPC member of VTC'2014-Fall, VTC'2017-Spring and Globecom.

She was a research intern with Mitsubishi Electric Research Laboratories, Cambridge, MA, USA, in 2015 and 2016. In 2017, she was with Silvus Technologies, Los Angeles, CA as a Research Engineer. She is currently Research & IC Design Engineer with Broadcom Limited, Irvine, CA. Her research interests include wireless communications, radio resource management, energy-efficient transmission techniques, statistical signal processing, machine learning and cognitive radio systems.

Milutin Pajovic (M'15) received the Dipl.-Ing. degree in Electrical Engineering from the University of Montenegro, Podgorica, Montenegro, in 2005, the M.S. degree in Ocean Engineering from the Florida Atlantic University, Boca Raton, FL, USA, in 2009, and the Ph.D. degree in Electrical and Oceanographic Engineering from the Massachusetts Institute of Technology and Woods Hole Oceanographic Institution, Cambridge, MA, USA, in 2014.

Dr. Pajovic has been with the Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, USA, since 2014, where he is currently a Principal Research Scientist. Previously, he was a research intern at MERL and ExxonMobil's Upstream Research Company, Houston, TX, USA, in Summers 2013 and 2012, respectively, and engineer for communication networks at the Agency for Electronic Communications, Podgorica, Montenegro, from 2005 to 2007. His current research interests include statistical signal processing, machine learning, and wireless and optical communications.

Zafer Sahinoglu (SM'01) received the M.Sc. degree in biomedical engineering and the Ph.D. degree in electrical engineering from the New Jersey Institute of Technology, Newark, NJ, USA, in 2001 and 2004, respectively, and the M.B.A. degree from the Massachusetts Institute of Technology Sloan School of Management, Cambridge, MA, USA, in 2013.

He worked at MERL between 2001 and 2016 as a senior principal research scientist. He technically contributed to and served in officer positions in IEEE, ZigBee and MPEG21 standards. He has published over 80 journal and conference papers, and also written two books published by Cambridge University Press. He is currently Senior Director, Business Innovation, at Mitsubishi Electric US Inc.

Yebin Wang (M'10-SM'16) received the B.Eng. degree in Mechatronics Engineering from Zhejiang University, Hangzhou, China, in 1997, M.Eng. degree in Control Theory & Control Engineering from Tsinghua University, Beijing, China, in 2001, and Ph.D. in Electrical Engineering from the University of Alberta, Edmonton, Canada, in 2008.

Dr. Wang has been with Mitsubishi Electric Research Laboratories in Cambridge, MA, USA, since 2009, and now is a Senior Principal Research Scientist. From 2001 to 2003 he was a Software Engineer, Project Manager, and Manager of R&D Dept. in industries, Beijing, China. His research interests include nonlinear control and estimation, optimal control, adaptive systems and their applications including mechatronic systems.

Philip V. Orlik (M'97-11, SM'12) was born in New York, NY in 1972. He received the B.E. degree in 1994 and the M.S. degree in 1997, both from the State University of New York at Stony Brook. In 1999 he earned his Ph.D. in electrical engineering also from SUNY Stony Brook.

Since 2000 he has been with Mitsubishi Electric Research Laboratories Inc. located in Cambridge, MA where he is currently the Manager of the Electronics and Communications Group. His primary research focus is on advanced wireless and wired communications, sensor/IoT networks. Other research interests include vehicular/car-to-car communications, mobility modeling, performance analysis, and queuing theory.

Toshihiro Wada received the B.Eng. degree in informatics and mathematical science, and the M.Informatics degree in applied analysis and complex dynamical systems from Kyoto University, Kyoto, Japan, in 2005 and 2007, respectively.

His current research interests include sensing and optimization of battery systems.