Adaptive prognosis of Lithium-Ion batteries based on the combination of particle filters and radial basis function neural networks

Claudio Sbarufatti[2], Matteo Corbetta[2], Marco Giglio[2], Francesco Cadini*[1]

[1] Politecnico di Milano, Dipartimento di Energia

Via La Masa 34, I-20156 Milano, Italy

[2] Politecnico di Milano, Dipartimento di Meccanica

Via La Masa 1, I-20156 Milano, Italy

ABSTRACT

Lithium-Ion rechargeable batteries are widespread power sources with applications to consumer electronics, electrical vehicles, unmanned aerial and spatial vehicles, etc. The failure to supply the required power levels may lead to severe safety and economical consequences. Thus, in view of the implementation of adequate maintenance strategies, the development of diagnostic and prognostic tools for monitoring the state of health of the batteries and predicting their remaining useful life is becoming a crucial task. Here, we propose a method for predicting the end of discharge of Li-Ion batteries, which stems from the combination of particle filters with radial basis function neural networks. The major innovation lies in the fact that the radial basis function model is adaptively trained on-line, i.e., its parameters are identified in real time by the particle filter as new observations of the battery terminal voltage become available. By doing so, the prognostic algorithm achieves the flexibility needed to provide sound end-of-discharge time predictions as the charge-discharge cycles progress, even in presence of anomalous behaviors due to failures or unforeseen operating conditions. The method is demonstrated with reference to actual Li-Ion battery discharge data contained in the prognostics data repository of the NASA Ames Research Center database.

Keywords: Lithium-ion batteries, Remaining Time to Discharge, Prognosis, Particle Filters, Radial Basis Functions, Neural Networks.

**Corresponding author, ph.: +39 02 2399 6355; fax: +39 02 2399 6309*

email: francesco.cadini@polimi.it

1. INTRODUCTION

In the last years, lithium-ion (Li-ion) batteries have gained a dominant position with respect to other battery technologies, such as lead-acid, nickel-cadmium or nickel-metal-hydride cells. The main reason of this success is that they offer significant advantages with respect to the other battery types, such as, for example [1]: i) the lower weight, due to the lightweight lithium and carbon-made electrodes, and, at the same time, the larger energy density, due to the high chemical reactivity of lithium; ii) the possibility of being recharged also if they are not completely discharged without any detrimental effects (no memory effect); iii) the lower self-discharge rate, so that they better maintain their charge when not used; iv) the longer life cycle, since they can operate successfully even after hundreds of charge-discharge cycles. These characteristics have contributed to their widespread application to many different types of system, ranging from consumer electronics (cell phones, laptops, etc.), to hybrid/electric vehicles and even solar system exploration probes. Consequently, failures of Li-Ion batteries may have a wide range of possible consequences, characterized by very different degrees of severity, peaking in the two ill-famed accidents of the plane crash in April 2000 due to the failure of the power supply system of its landing gear and the loss of the Mars Global Surveyor in November 2006 due to an excessive heating of the battery radiators on-board [2][3]. Thus, in the last decade, many efforts have been made to develop methods for improving the reliability and the availability of Li-Ion batteries, thus benefiting the users' satisfaction.

In this context, a major role is played by prognostic and health management (PHM) methods, which aim at automatically monitoring the degradation state of the batteries, on the basis of different kinds of available information, such as sensor's signals, physics-based dynamic models, past usage data of similar components, boundary conditions (environmental or operational), etc., and at predicting their remaining useful lifetime, possibly to support prediction-based maintenance actions. More specifically, as underlined in [4], PHM methods address the crucial problems of whether the batteries are capable of supplying an adequate power level during any discharge cycle, what kind of degradation mechanisms the batteries are undergoing and how many more charge-recharge cycles the batteries can withstand without reaching unacceptable levels of degradation of its performances. Eventually, attempting to answer these questions amounts to searching for possible strategies for the estimation of several battery parameters and/or health metrics, such as the state-of-health (SOH), state-of-charge (SOC), state-of-life (SOL), the end-of-discharge (EOD) time and the remaining-useful-life (RUL) [4]. The SOH actually collects the measures/estimates of different internal battery physical parameters, such as its resistance, its impedance, its capacity, etc., internal operating

conditions, such as current, terminal voltage, temperature, etc., as well as external operating conditions influencing the behavior and the degradation rate of the battery. The SOC, on the other hand, estimates the amount of charge (in [Ah] or in [%]) still available at the terminals as the battery is being used, i.e., in other words, during its discharge cycle. Note that the SOC is a very important diagnostic and prognostic metric for applications in Electric (or Hybrid) Vehicles (EV), since it is a fundamental input of modern battery management systems (BMSs) for performing voltage-limit-based control and it can be assumed as the equivalent of the fuel gauge in fossil fuel-based vehicles, thus providing an indication of the remaining usage of the vehicle before the batteries need recharging [5]. Alternatively, in some applications, the terminal voltage variation during a discharge (in [V]), which is strictly related to the current and the SOC, is directly used as a diagnostic/prognostic indicator during the battery discharge phase, as for example done in [1] and in many other works of literature based on the NASA Ames database [6]. The SOL estimates the remaining capacity of the battery (usually in [Ah]) as it fades through the charge-discharge cycles due to different degradation processes. The EOD is usually defined as the time at which either the SOC or the terminal voltage reach some predefined thresholds, so that the battery needs to be charged, whereas the RUL is the remaining time at the end of which the battery capacity after recharge (SOL) will fade below a threshold, indicating that the battery must be replaced [4][6][8].

Many methods have been proposed in the recent years for the diagnosis and the prognosis of Li-Ion batteries, which consist either of (i) the direct measurements of the quantities of interest, such as, for example, electrochemical impedance spectroscopy (EIS) methods for impedance estimation or current pulses-based techniques for DC resistance estimation, or (ii) the use of some indirect, but more accessible observations, such as the current, the terminal voltage, etc., within different modeling/algorithmic schemes to infer the desired diagnostic/prognostic indicators [9]. While the former methods can hardly be used, in general, for the real-time operation typically required by PHM applications, due to the expensive and complex instrumentation required, the latter are, on the other hand, more suitable for performing on-line diagnosis and prognosis.

The methods based on indirect measurements can be, in turn, classified in three families: model-based, data-driven or hybrid methods [10]. Model-based methods aim at relating observable quantities to the indicators of interest by building either electro-chemical detailed model of the degradation processes affecting the battery life, or equivalent electrical circuits of the battery. Data-driven methods, on the other hand, do not require the explicit derivation of a physics-based model, since they aim at mapping the relationships between the accessible observations and the hidden indicators by some approximating, general model adaptively built on the basis of available data, such as, for example, those relying on neural networks (NN), Gaussian process functional

regressions, support vector regressions, fuzzy inference engines, etc. For thorough reviews of the existing model-based and data-driven methods, the interested reader may refer to the many recent works of literature, such as for example [6], [8], [9], or [11]. Both families of approaches have turned out to offer significant advantages in different practical situations. In this regard, the work in [12] provides useful guidelines for the identification of the best strategy to adopt, depending on the amount of the information available, although in a context different from that of the PHM of Li-Ion batteries. Indeed, the successful application of model-based methods heavily depends on the physics-based model adopted, whose accuracy is, in general, the result of a trade-off between the reliability of the outcomes and the computational efforts required to obtain them; at the same time, the effectiveness of data-driven methods strongly relies on the amount and the quality of the data available. In this context, hybrid methods aim at combining model-based and data-driven methods, when possible, in an attempt to overcome the limitations of the individual methods and, thus, improve diagnostic and prognostic accuracies by better exploiting all the available information. A promising hybrid strategy seems, in particular, that of resorting to Bayesian filtering-based algorithms, where the required analytical models representing either the dynamic behavior of the system or the measurement equation are actually suitable data-driven surrogate models. For example, in [13] an extended Kalman filter (EKF) is coupled to a NN trained off-line to approximate the relationship between the observable battery terminal voltage at the present temporal step and the voltage, the current and the SOC at previous steps, in order to be able to estimate the present battery SOC; a similar approach is propounded also in [14] for the prognosis of gas turbine engines, where the EKF is substituted by the a more flexible particle filter (PF). The major drawback of these approaches is that the NNs have to be trained off-line, so that, if the battery operates in conditions different from those under which the training data are obtained, their performances may severely deteriorate. In [15], this problem is partially overcome by combining a PF with an exponential model of the battery SOL identified from data, but whose parameters are updated on line, thus guaranteeing a certain degree of adaptability. However, the functional form of the SOL model seems rather rigid, so that any sudden, unexpected changes in the SOL dynamics may hamper the filter performances.

In this work, similarly to the work in [1], we tackle the problem of the real-time prognosis of the EOD time of a Li-Ion battery on the basis of terminal voltage observations. Differently from [1], where a PF was coupled to a classical electrical model of the battery discharge dynamics, with lumped parameters to be identified on line, here we aim at providing a fully adaptive scheme based on a combination of a PF with a NN. More specifically, we propose to resort here to the radial basis functions (RBF) networks, which were proven to be suitable for SOC estimation in some works of

literature [16][17][18]. The algorithm represents an advancement with respect to the past works using particle filters (e.g. [1] or [15]) or NNs ([16], [17] or [18]) alone, or even their combination (e.g. [13] or [14]): in fact, the RBF training is performed by adapting the algorithm proposed in [19] for training general feed-forward NNs with particle filters. More specifically, the RBF's parameters (centers, weights and biases) can be identified on-line on the basis of the input-output patterns as they are collected by the measurement system. The resulting algorithm is capable of adaptively calibrating itself, while performing its diagnostic and prognostic tasks, virtually with no need for off-line training. The use of a general NN (RBF) instead of other models whose functional form has been previously identified by regression on available data (as in [15]), provides more flexibility to the algorithm and, consequently, reduces its dependence on the laboratory data typically used for model (parameters and/or functional form) identification purposes. These data are usually obtained by means of experiments conducted in highly controlled conditions, hardly met in the realistic operation of most Li-Ion batteries. On the other hand, the larger flexibility of the proposed algorithm allows to effectively deal with unexpectedly accelerated aging scenarios, due, for example, to improper recharge or storage procedures. The algorithm is demonstrated with respect to the NASA Ames Research Center database available in the prognostics data repository [6].

The paper is structured as follows. Section 2 summarizes the methodology and its tailoring to deal with the specificity of the terminal voltage monitoring and prediction tasks. Section 3 discusses the performance of the method against experimental data available in [6]. The algorithm capabilities in estimating of the terminal voltage and in forecasting the EOD time are demonstrated in case of normal aging and unexpected, accelerating aging scenarios. Section 4 draws some conclusions on the proposed method and its further developments.

2. METHODOLOGICAL APPROACH

This Section first briefly reviews the basics of the RBF networks functioning and illustrates our adaptation of the method proposed by [19], where a multi-layer perceptron NN model was sequentially trained by means of a sequential importance resampling (SIR)-based PF algorithm. The RBF-PF approach is then customized in order to predict the discharge dynamics of the terminal voltage of Li-Ion batteries under a constant current mode, until it reaches the predefined threshold V_{min}. The interested reader is referred to [20] and [21], and to [22] and [23] for thorough descriptions of the functioning of RBFs and particle filters, respectively, and to [19] for further details on particle filtering-based NN training.

2.1. RBF neural networks

Compared to other types of artificial NNs, such as the multi-layer perceptrons [20], RBF networks possess the property of best approximation, can be trained with substantially faster procedures and are characterized by simpler structures, which make them attractive for time series prediction [20][24]. In fact, focusing on their structures, RBF networks are typically built with one hidden layer with non-linear activation functions and a linear output layer, in order to maintain a fast training procedure [21]. A schematic view of the RBF network is shown in Figure 1, specifically tailored for this application, where the RBF model aims at approximating the decreasing terminal voltage as a function of time. The input node represents the time (t) at which the terminal voltage is measured, whereas the terminal voltage measure (z) is associated to the output node. The output node collects the non-linear outputs from a generic number M of hidden nodes, each one weighted by a factor w, and the bias w_0. In a RBF network, each hidden node consists of a basis function Φ, which provides a non-linear activation dependent on the distance between the input (t) and a prototype vector, hereafter referred to as the basis function center. The output of the RBF network in Figure 1 is calculated as:

$$z = \sum_{j=1}^{M} w_j \, \Phi_j\left(t, \xi_j\right) + w_0 \tag{1}$$

The vector ξ_j contains the parameters of the j-th basis function Φ_j. Here we choose a RBF network structure with a basis function $\Phi = r^4 \log r$ [21], written as a function of the radial distance of the input from the basis function center (c_j), so that ξ_j only contains c_j and $r = \left\| t - c_j \right\|$. Equation (1) can thus be written as:

$$z = \sum_{j=1}^{M} w_j \left\| t - c_j \right\|^4 \log\left(\left\| t - c_j \right\|\right) + w_0 \tag{2}$$

As a general principle, M should be sized according to the complexity of the relationship that has to be learnt. The terminal voltage dynamics are, in general, well represented by three different behaviors as a function of time: a sudden drop at the beginning of the discharge, a slower decrease in the central part of the process and another sudden drop at the end of the discharge curve, with a curvature opposite to that of the first drop (see Figure 2 (a) for some examples). Intuitively, the larger the number of hidden neurons, the higher the capability of the algorithm in approximating complex relationships [20]. However, a large number of hidden neurons may give rise to data

overfitting issues, thus strongly limiting the generalization capability of the NN on new data, especially for prediction outside the training domain. Furthermore, increasing M also induces a larger number of unknown parameters to be optimized during training, which may significantly increase the computational burden of the updating process described later. In this application, the number of hidden neurons has been empirically set to five based on a trial and error procedure. According to a qualitative sensitivity analysis performed by the authors, and not shown here for brevity's sake, a slightly smaller or larger hidden layer (e.g., 4-7 neurons) still provides satisfactory results.

With this choice of the RBF network structure, we can collect the $n_x = 11$ network parameters in a vector $\boldsymbol{x} = [x_1, x_2, \dots, x_{11}]^T$, where the elements $\boldsymbol{x}_{1:5}$ are the five centers of the basis function ($\boldsymbol{c}_{1:5}$), the $\boldsymbol{x}_{6:10}$ are the five RBF weights ($\boldsymbol{w}_{1:5}$) and x_{11} is the RBF network bias (w_0). Note that the weights in $\boldsymbol{x}_{6:10}$ are different from the importance weights of the PF algorithm described later on.

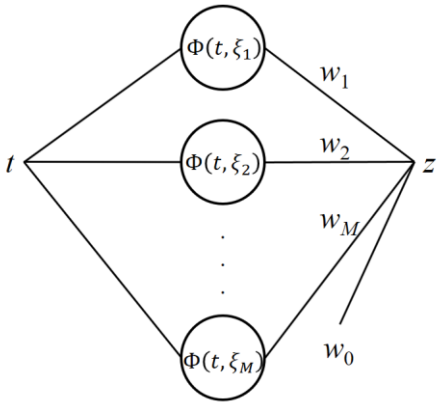

Figure 1. Example of RBF neural network, with one input (t), M hidden nodes and one output (z).

2.2. The RBF-PF algorithm

The Bayesian framework for the sequential training of the RBF model grounds on a state-space representation of the time evolution of the n_x RBF model parameters, which are collected in the state vector $\boldsymbol{x} \in \mathbb{R}^{n_x \times 1}$. The set of the NN parameters $\boldsymbol{x}$ will be called RBF model parameter vector or system state vector henceforth. The filtering problem presented here differs from most of the Bayesian filtering problems analyzed in the prognostic literature, where the objective was the estimation of physical quantities collected in the system state vector [25]. In this application, the state vector collects the parameters of the RBF NN model in (2), so that the filtering process aims at training the NN sequentially. Note that the "time evolution" of the RBF parameters refers to the

updating procedure driven by the particle filter, as it will be described later. This evolution is then governed by the time instants at which the process to be approximated is observed. Under the common operative assumption that the process observations are available at discrete times, the state space representation then reads:

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{x}_{k-1} + \boldsymbol{\omega}_{k-1} \\ z_k &= g(\boldsymbol{x}_k, t_k) + \eta_k \end{aligned} \tag{3}$$

where the subscript k is the index of the discrete time step, t_k and z_k are the input and the output of the non-linear RBF mapping $g(\cdot)$ in (2), i.e. the time instant and the corresponding value of the terminal voltage, respectively, the random process $\boldsymbol{\omega}_k \in \mathbb{R}^{n_x \times 1}$ is the stochastic component of the RBF model parameter evolution and $\eta_k \in \mathbb{R}$ is the measurement noise [22]. Thus, according to [19] and to many other works of literature (e.g. [15]) on fixed parameter estimation by particle filtering, the evolution of the network weights during their updating process is modeled as a discrete Markov chain driven by a process noise $\boldsymbol{\omega}_k$ which allows to properly "explore" the parameters' space in order to identify a satisfactory set of values and, at the same time, guarantees a certain degree of flexibility in order to be able to possibly adapt to any, possibly unexpected, changes in the process dynamics.

Note that this procedure bears some similarities with that presented in [15], the major difference lying on the use of a more general RBF network instead of a previously identified, but less flexible, double exponential model.

The final objective of this Bayesian framework is the recursive estimation of the posterior probability density function (pdf) of the RBF model weights $\boldsymbol{x}_k$ given the set of observations $\boldsymbol{z}_{0:k}$ up to the current time step k, i.e., the posterior pdf of the state vector $p(\boldsymbol{x}_k|\boldsymbol{z}_{0:k})$. The optimal solution of this problem requires is the well-known optimal Bayesian recursion, which involves a prediction and an update stage, defined by the Chapman-Kolmogorov equation and the Bayes' rule, respectively [22][23]:

$$\begin{aligned} p(\boldsymbol{x}_k|\boldsymbol{z}_{0:k-1}) &= \int p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) p(\boldsymbol{x}_{k-1}|\boldsymbol{z}_{0:k-1}) \mathrm{d}\boldsymbol{x}_{k-1} \\ p(\boldsymbol{x}_k|\boldsymbol{z}_{0:k}) &= \frac{p(\boldsymbol{z}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{z}_{0:k-1})}{p(\boldsymbol{z}_k|\boldsymbol{z}_{0:k-1})} \end{aligned} \tag{4}$$

Since the perturbation $\boldsymbol{\omega}$ is the only source of uncertainty in the process equation and it is assumed independent of the state vector, the transition pdf can be written as:

$$p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = \int \delta(\boldsymbol{x}_k - \boldsymbol{\omega}_{k-1} - \boldsymbol{x}_{k-1})\, p(\boldsymbol{\omega}_{k-1})\, \mathrm{d}\boldsymbol{\omega}_{k-1} \quad (5)$$

where $\delta(\cdot)$ denotes the delta Dirac function. Similarly, the likelihood function and the evidence become [22]:

$$p(z_k|\boldsymbol{x}_k) = \int \delta(z_k - g(\boldsymbol{x}_k, t_k) - \boldsymbol{\eta}_k)\, p(\boldsymbol{\eta}_k)\, \mathrm{d}\boldsymbol{\eta}_k \quad (6)$$

$$p(z_k|\boldsymbol{z}_{0:k-1}) = \int p(z_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{z}_{0:k-1})\, \mathrm{d}\boldsymbol{x}_k \quad (7)$$

As extensively discussed in many works of literature, the nonlinearity of the dynamic state-space model in (3) and/or any non-Gaussianity of the process and/or measurement noises are such that analytical solutions of the optimal prediction-update recursion in (4) cannot be obtained. Thus, the estimation of a sub-optimal solution for the posterior pdf $p(\boldsymbol{x}_k|\boldsymbol{z}_{0:k})$ is typically achieved by means of numerical methods, such as direct numerical integration, Gaussian approximations and Monte Carlo (or particle filters methods). Here, we resort to the solution based on the sampling importance resampling (SIR) PF algorithm proposed in [22].

A Monte Carlo estimator of the RBF weights posterior pdf can then be obtained as:

$$\hat{p}(\boldsymbol{x}_k|\boldsymbol{z}_{0:k}) \approx \sum_{i=1}^{N_s} w_k^{(i)} \delta\left(\boldsymbol{x}_k - \boldsymbol{x}_k^{(i)}\right) \quad (8)$$

where the importance samples $\boldsymbol{x}_k^{(i)},\ i = 1, \ldots, N_s$ are N_s independent and identically distributed realizations of the system state vector, drawn from the importance pdf $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$. The terms $w_k^{(i)},\ i = 1, \ldots, N_s$ are the normalized importance weights and $\delta(\cdot)$ is the Kronecker delta.

As proven in [23], the normalized importance weights can be recursively computed as

$$w_k^{(i)} = \frac{\tilde{w}_k^{(i)}}{\sum_{i=1}^{N_s} \tilde{w}_k^{(i)}} \quad (9)$$

where the non-normalized weights $\tilde{w}_k^{(i)}$ are given by

$$\tilde{w}_k^{(i)} = w_{k-1}^{(i)} p\left(z_k|\boldsymbol{x}_k^{(i)}\right) \quad (10)$$

The function $p\left(z_k|\boldsymbol{x}_k^{(i)}\right)$ is the likelihood of the observation z_k, i.e., the probability of observing z_k given the underlying non-linear mapping $g(\boldsymbol{x}_k, t_k)$ is approximated by the i-th RBF model defined by the parameters $\boldsymbol{x}_k^{(i)}$.

Finally, in order to avoid the sample impoverishment problem [22][23], which dramatically reduces the number of samples with non-negligible weights, thus hampering the posterior pdf representation, a resampling scheme is implemented after the weight normalization. Here, a systematic resampling approach is adopted, which generates a new pool of N_s samples by picking (with replacement) the i-th sample with probability $\Pr\left\{\boldsymbol{x}_k^{(j)} = \boldsymbol{x}_k^{(i)}\right\} = w_k^{(i)}, \forall\, j = 1, \ldots, N_s$ from the original set.

The estimation of the posterior distribution $p(\boldsymbol{x}_k|\boldsymbol{z}_{0:k})$ is carried out each time k a new observation z_k becomes available. The set of samples and associated weights $\left\{\boldsymbol{x}_k^{(i)}, w_k^{(i)}\right\}$ defines a set of N_s RBF network models, which can be used also to predict the future terminal voltage behavior over time and, consequently, the EOD, as it shall be discussed later.

Operatively, we assume that the random walk in (3) is driven by uncorrelated, zero-mean Gaussian noises, i.e., $\boldsymbol{\omega}_k \sim \mathcal{N}\left(0, \Sigma_{\boldsymbol{\omega}_k}\right)$, where the covariance matrix $\Sigma_{\boldsymbol{\omega}_k}$ is diagonal. In general, the choice of the noise variances is not an easy task: too small values may hamper a proper (and reasonably fast) exploration of the state-space, whereas too large values do not guarantee a satisfactory state estimation, as discussed by many authors in literature. A common approach for achieving a satisfactory trade-off is that of letting the variances decrease from an initial value as the estimation process progresses, so as to guarantee the convergence of the algorithm [25][27]. In this work, we propose the following expression for the process noise covariance matrix as a function of the discrete time step index k:

$$\Sigma_{\boldsymbol{\omega}_k} = \left(\sigma_0\, e^{-\frac{k}{\sigma_1}} + \sigma_2\right) \cdot \mathrm{I} \tag{11}$$

where all the variances are taken to be the same. The parameter σ_0 is such that $\Sigma_{\boldsymbol{\omega}_0} = (\sigma_0 + \sigma_2)$ is the initial value of the variances, σ_1 regulates their rate of decrease and σ_2 is their asymptotic value; I is the $n_x \times n_x$ identity matrix.

We further assume that the measurement noise η_k is also zero-mean, Gaussian with variance σ_η^2 and independent from k, i.e., $\eta_k \sim \mathcal{N}\left(0, \sigma_\eta^2\right)$. The likelihood function in (10) then becomes [19]:

$$p\left(z_k|\boldsymbol{x}_k^{(i)}\right) = \left(2\pi\sigma_\eta^2\right)^{-0.5} \exp\left\{-\frac{1}{2\sigma_\eta^2}\left(z_k - g\left(\boldsymbol{x}_k^{(i)}, t_k\right)\right)\right\} \quad (12)$$

where z_k represents the observation of the terminal voltage in correspondence of the discrete time steps t_k and $g\left(\boldsymbol{x}_k^{(i)}, t_k\right)$ is the correspondent prediction by the *i-th* RBF network with parameters $\boldsymbol{x}_k^{(i)}$. Operatively, we have observed that the adoption of a systematic resampling strategy combined with a relatively high number of parameters in the state vector $\boldsymbol{x}$ and a limited number of samples N_s hampers the convergence of the filter if only the last observation is included in the likelihood function. In practice, the procedure for the sequential updating of the system state vector gives rise to a "loss of memory", which, in turn, makes the filtered RBF network parameter unstable and, consequently, the prediction of the system future evolution more unreliable.

Thus, we modify the weight updating equation in (10) in order for the particles weights to account for the whole sequence of terminal voltage observations up to the current step *k*. By doing so, in fact, it is possible to obtain a more robust estimation of the RBF network parameters and, as a consequence, a more robust prediction of the battery discharge dynamics, as verified by the authors. To this aim, we introduce a $k \times k$ diagonal covariance matrix Σ_η, whose diagonal elements are all assumed equal to σ_η^2 and we define the modified likelihood function in (10) as follows:

$$p\left(z_{0:k}|\boldsymbol{x}_k^{(i)}\right) = \left((2\pi)^{k+1}|\Sigma_\eta|\right)^{-0.5} \exp\left\{-\frac{1}{2}\left(z_{0:k} - g\left(\boldsymbol{x}_k^{(i)}, t_{0:k}\right)\right)^T \Sigma_\eta^{-1}\left(z_{0:k} - g\left(\boldsymbol{x}_k^{(i)}, t_{0:k}\right)\right)\right\} \quad (13)$$

where the term $z_{0:k}$ represents the sequence of terminal voltage observations in correspondence of the discrete time steps $t_{0:k}$ and the terms $g\left(\boldsymbol{x}_k^{(i)}, t_{0:k}\right)$ are the correspondent predictions of the *i-th* RBF network with parameters $\boldsymbol{x}_k^{(i)}$.

As a result of the filtering process, N_s samples of the RBF parameter vector $\boldsymbol{x}_k$ and their N_s associated importance weights $\boldsymbol{w}_k$ are available at the current time step k.

2.3. Particle projection in the future

After the identification of the RBF model parameters (included in the state vector $\boldsymbol{x}_k$) at the current time step k in Section 2.2, now we discuss how they can be used to predict the future evolution of the battery discharge dynamics. Following the approach proposed in [28], at the current time step k the posterior pdf of the end of discharge time (i.e. EOD_k) is estimated by projecting in the future the

predictions of the N_s RBF networks associated to the N_s parameter samples (particles) $\boldsymbol{x}_k^{(i)},\ i = 1, \dots, N_s$. Operatively, a set of discrete future times $[t_{k+1}, \dots, t_{k+p}]$ is fed to each of the N_s RBF networks; the corresponding outputs, i.e., the terminal voltage predictions at the different time instants, are used to build the sample posterior pdf of the terminal voltage at future times. At the same time, the $\mathrm{EOD}_k^{(i)},\ i = 1, \dots, N_s$ predicted by each RBF network are used to build the sample posterior pdf of the EOD_k. The $\mathrm{EOD}_k^{(i)}$ are found as the time $t_{k_{EOD}}^{(i)}$ at which the predicted terminal voltages $g\left(\boldsymbol{x}_k^{(i)}, t_{k_{EOD}}^{(i)}\right)$ become lower than the threshold V_{min} for the first time. Thus, the index p is chosen so as to be reasonably sure that $t_{k_{EOD}}^{(i)} < t_{k+p},\ \forall\ i = 1, \dots, N_s$.

The estimates of the posterior pdfs of the future terminal voltages $\tilde{z}_{k+l},\ l = 1, 2, \dots$, and of the EOD are operatively obtained as:

$$\hat{p}(\tilde{z}_{k+l}|\boldsymbol{z}_{0:k}) \approx \sum_{i=1}^{N_s} w_k^{(i)} \delta\left(\tilde{z}_{k+l} - g\left(\boldsymbol{x}_k^{(i)}, t_{k+l}\right)\right) \tag{14}$$

$$\hat{p}(\mathrm{EOD}_k|\boldsymbol{z}_{0:k}) \approx \sum_{i=1}^{N_s} w_k^{(i)} \delta\left(\mathrm{EOD}_k - \mathrm{EOD}_k^{(i)}\right) \tag{15}$$

Consequently, if we define the remaining time to discharge (RTD) at the current time step k as $\mathrm{RTD}_k = (\mathrm{EOD}_k - t_k)$, its estimated posterior pdf is the same as $\hat{p}(\mathrm{EOD}_k|\boldsymbol{z}_{0:k})$, but shifted by t_k.

3. RESULTS

In this Section, we demonstrate the prediction capabilities of the proposed RBF-PF algorithm using run-to-failure data of Li-ion batteries, generated during discharge-charge experiments in a laboratory controlled environment. The dataset is available in the prognostic data repository provided by the Prognostics Center of Excellence at NASA Ames Research Center [6]. The progressive discharge of the batteries during the discharge-charge experimental cycles was monitored by recording terminal voltage, capacity and impedance over time. The terminal voltage measurements are used to monitor and predict the EOD during several discharge-charge cycles. In particular, the dataset *battery 5* (B005.mat) of the prognostics data repository is here analyzed. The experiments were carried out under the following operational assumptions:

- The discharge was performed at a constant current level of $I = 2\ A$ and at room temperature.
- The discharges were stopped when the battery terminal voltage reached approximately $2.7\ V$.

- The experiments were stopped when the batteries met the end-of-life criteria, i.e. a 30% fade in rated capacity (from 2Ahr to 1.4Ahr) [6].

Repeated discharge and charge cycles thus resulted in a progressive aging of the batteries, i.e. in a degradation of their performance (rated capacities). Further details on the experimental setup are out of the scope of this work: the interested reader may refer to [6] for additional information.

The *battery 5* dataset contains 168 discharge transients monitored through measures of the terminal voltage V. After 168 discharge-charge cycles, the test ended. Figure 2 (a) shows all the discharge curves, where the longest curve refers to the pristine battery and the shortest one refers to the last discharge before the end of the test. The terminal voltage measures are characterized by very little noise due to the extreme level of control of the experimental procedure.

However, in real applications, in general, terminal voltage (or other battery health indicators) observations are much noisier, as it shall be shown in our application, so that real-time health monitoring becomes more challenging. For instance, impedance measures using electromechanical impedance spectroscopy have proven to be particularly noisy in practical applications [1], and the work in [29] shows noisy experimental terminal voltage measures used to predict the RTD of the battery. As a consequence, discharge prognosis may request the filtering of noisy observations of health indicators [30][31] to guarantee high performance and reliable predictions.

In order to test the robustness of our algorithm, we simulate a more realistic, real-time scenario by corrupting the original terminal voltage measurements of Figure 2 (a) with an artificial noise, as shown in Figure 2 (b). In correspondence of each curve, a new set of noisy terminal voltage data is then generated. The artificial noise is modeled as a Gaussian random variable with zero-mean and variance proportional to the slope of the discharge curve.

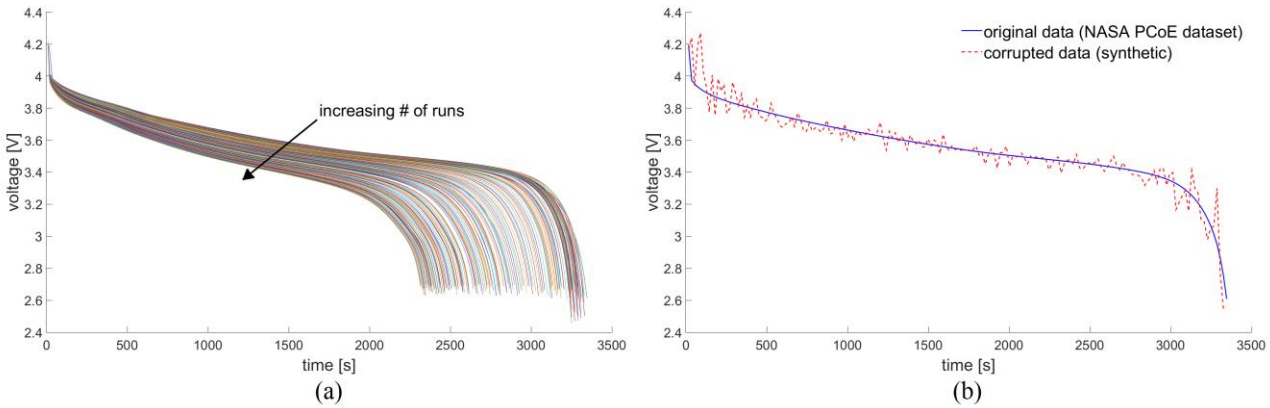

Figure 2. (a) Discharge curves from the battery 5 dataset of the prognostics data repository, [6] and (b) example of original and artificially corrupted terminal voltage data.

3.1. Algorithm initialization and run

The proposed algorithm is run with $N_s = 4000$, $\sigma_0 = 10^{-5}$, $\sigma_1 = 500$ and $\sigma_2 = 10^{-6}$, respectively. The measurement noise variance is assumed to be equal to $\sigma_\eta^2 = 0.05$. These choices were driven by authors' experience and by the order of magnitudes of the quantities involved in the process, but they might vary on a case-basis.

The initialization of the state vector $\boldsymbol{x}_0$ (the RBF's parameters) is crucial for the convergence of the RBF-PF algorithm during operation. As verified by the authors, but not shown here for brevity's sake, the RBF-PF algorithm does not always reach convergence if the RBF model features $\boldsymbol{x}_0$ are randomly initialized. Therefore, the system state vector is initialized by resorting to the classical two-step method for RBF network training provided by the *Netlab* tool [21]. The initial centers, weights and biases of the RBF model are tuned on a single discharge curve corresponding to the first discharge curve of the dataset (i.e., the longest curve in Figure 2 (a)). The RBF model features identified by the two-step training method are then modified by adding a zero-mean Gaussian noise with variance in (11) to generate a pool of possible features $\boldsymbol{x}_0^{(i)}, \forall\ i = 1, \dots, N_s$. This *a priori* tuning of the RBF network parameters aims at concentrating the samples in a potentially more useful region in the 11-dimensional state-space, thus favoring the convergence of the filter. Then, the discharge curve used for the initialization is discarded and the prediction capabilities of the algorithm are assessed starting from the second discharge curve available in the dataset. In principle, one could avoid the initial tuning of the parameters and let the RBF-PF adjusts its state vector starting from random values, but this procedure would require a much larger set of particles in order to be able to properly explore the parameters' space, thus incurring in heavier computational expenses.

After the algorithm is initialized, it updates the RBF model features according to the sequential information gathered on the terminal voltage over time. The EOD posterior distribution is also updated when new observations become available, and the algorithm stops as the last terminal voltage observation is collected. Before the start of the next discharge cycle, the system state vector is initialized using the last sample obtained in the previous run, i.e., $\boldsymbol{x}_0^{(i)} = \boldsymbol{x}_K^{(i)}$ (where K is the last time-step of the previous discharge cycle) and the initial covariance matrix of the perturbation $\Sigma_{\boldsymbol{\omega}_0}$. The pseudo-code in Table 1 summarizes the algorithm operative procedure for the prediction of the RTD.

Figure 3 visually represents the algorithm functioning during a single discharge cycle. The reference trajectory, i.e., the terminal voltage discharge curve from the dataset [6], is shown as a grey line (note that these data are not used in any step of the algorithm). The observations

artificially generated from the reference trajectory are represented by the blue dots. The limit threshold at which the battery is considered exhausted is set to 70% of the initial terminal voltage $V_{in} = 4.2\,V$, i.e., $V_{min} = 2.95\,V$. Two examples of decreasing terminal voltage trajectories predicted by two different RBFs of the RBF-PF pool are shown as dashed, gray lines in Figure 3.

1. Collect a discharge curve $V(t)$ representative of the operative condition of the battery (e.g., first discharge curve).
2. Initialize the RBF features $\overline{\boldsymbol{x}}_0$ using the two-stage training method (*Netlab*)
3. Initialize the state vector $\boldsymbol{x}_0 = \overline{\boldsymbol{x}}_0 + \mathcal{N}\left(0, \Sigma_{\omega_0}\right)$
4. SOC prognosis:
    4.1. For every time step *k* from 0 to *K*
        4.1.1. Perform the transition $\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \mathcal{N}\left(0, \Sigma_{\boldsymbol{\omega}_{k-1}}\right)$
        4.1.2. Update the covariance matrix $\Sigma_{\boldsymbol{\omega}_k}$ using (11)
        4.1.3. Evaluate the likelihood using (13), the weights and the posterior distribution of the system state
        4.1.4. Predict the evolution of the terminal voltage using (14)
        4.1.5. Estimate the remaining time before that the terminal voltage drops below V_{min} using (15) for all the samples
        4.1.6. Compute the sample RTD posterior pdf using the pairs $\left\{RTD_k^{(i)}, w_k^{(i)}\right\}$
        4.1.7. Resample the system state vector using the systematic resampling method [23]
    4.2. When the algorithm processes the last observation,
        4.2.1. Wait for the next charge
        4.2.2. Initialize the RBF model features for the next discharge using the last estimated RBF model features
5. Repeat step 4 for every discharge cycle

Table 1. RBF-PF algorithm pseudo-code.

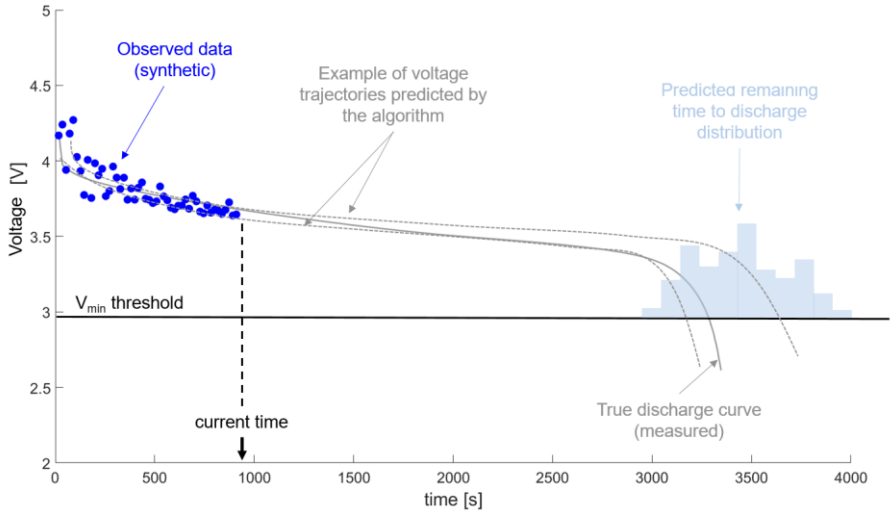

Figure 3. Visual description of the input and output quantities of the RBF-PF algorithm.

3.2. Prediction results

Figure 4 (a-c) show an example of discharge predictions at different time steps, i.e. ~800 s (a), ~1800 s (b) and ~3000 s (c). They refer to the first run, so the first discharge curve that has to be monitored (which is the curve #2 of the dataset, since the first has been used to initialize the algorithm). The state estimation, i.e., the *a posteriori* expected value of the terminal voltage at the times ~800 s, ~1800 s and ~3000 s (solid red lines with triangles), along with their confidence intervals (dashed, red lines), are compared to the synthetic terminal voltage measurements (blue dots) gathered up to the prediction time. Some samples of terminal voltage trajectories predicted by the RBFs in the pool are also shown (dashed, light gray lines; only a limited number of predicted trajectories is shown for the readability of the figure). The gray histogram at the end of the discharge transients is the estimate of the posterior EOD pdf.

In order to emphasize the advantage of the proposed method against more traditional usage of NN models, the prognostic of the terminal voltage curve using the two-step training method of RBF networks has been plot on the same Figures (a-c). It has been called *trivial prediction*, since it does not involve any stochastic or probabilistic approach to predict the future terminal voltage dynamics. The standard RBF network is built by using the parameters of the RBF-PF initialization, and then updates such features with the sequential terminal voltage observations (blue dots) to perform the forecast (thick, dashed and green line). The poor prediction performance of the RBF model trained with the two-stage method are evident in this comparison; the predictions are never able to project reasonable trajectories outside the training domain (the latter stopping at the last, *k-th* observation).

Figure 4 (d) shows the RTD prediction (i.e., the RTD sample posterior PDF mean) of the first algorithm run as a function of the prediction time (dashed and thick red lines) along with its 95% confidence intervals (dashed and thin red lines) with respect to the actual RTD (blue line).

The prediction slightly underestimates the true RTD, the largest error occurring in correspondence of the plateau of the transient. After ~2000 s, the predictions drastically improve and the confidence intervals shrink around the target RTD. When the first discharge ends, the battery is recharged and the new prediction starts as soon as the algorithm receives new terminal voltage observations.

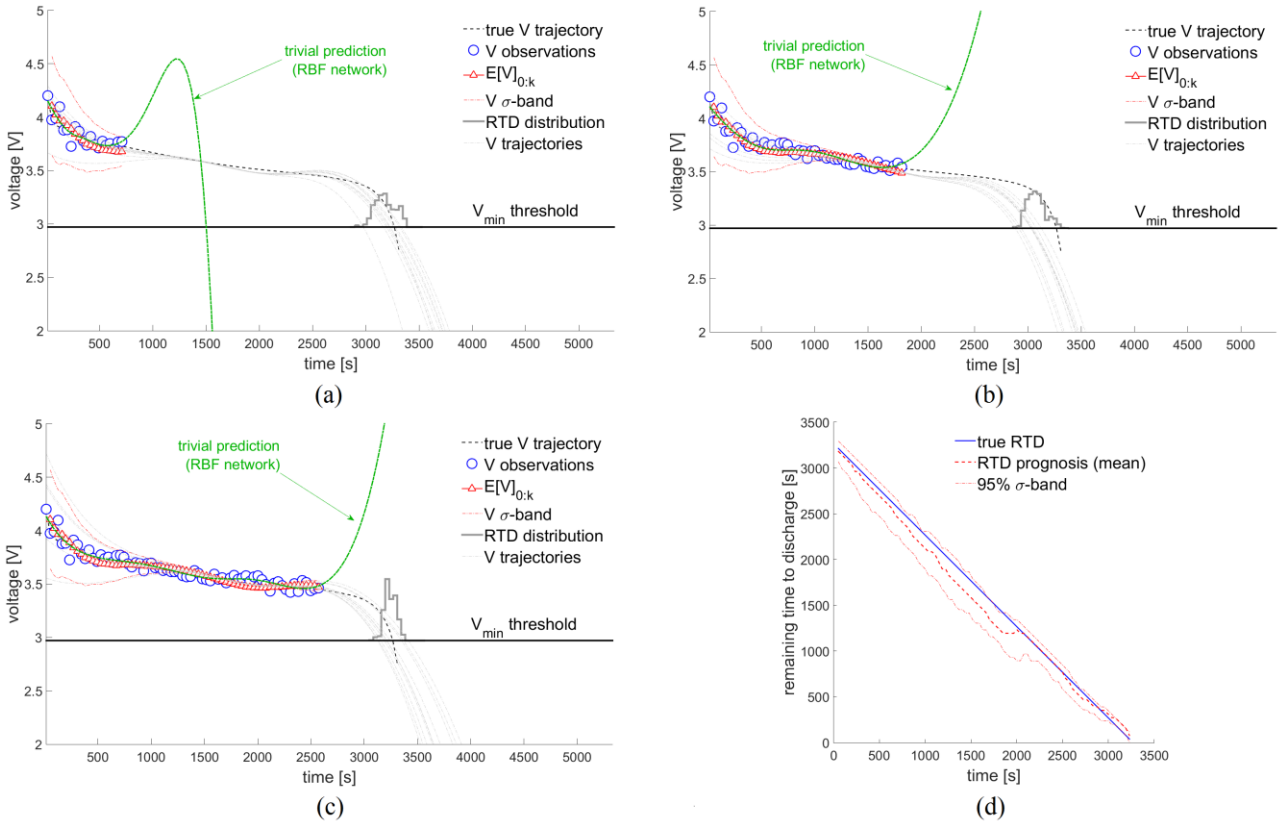

Figure 4. Filtered terminal voltage after ~800 (a), ~1800 (b) and ~3000 (c) seconds, and RTD prediction (d).

3.3. Prediction under unexpected accelerated aging

We now demonstrate the RBF-PF capabilities to deal with unexpected changes in the discharge behavior possibly occurring during the functioning of the battery. In order to do so, we simulate an accelerated discharge of the battery as follows. We initially predict the first discharge transient, which refers to the pristine battery, as done before. Then, instead of running the RBF-PF on the subsequent discharge curve, we select the curve #101 of the dataset as the new observation sequence. Indeed, the transient #101 is characterized by a much faster discharge rate than the first

one, due to the fact that, at this stage, the battery has already undergone 100 discharge-charge cycles, thus significantly degrading its performances. By doing so, we force the RBF-PF to operate in an unexpectedly accelerated aging scenario, due, for example, to improper recharge, storage procedures or temperature-induced intensified degradation. It is to be kept in mind, in fact, that real discharge-charge cycles are characterized by behaviors different from those observed in highly controlled laboratory experiments. Figure 5 (a) shows the two transients used to simulate the accelerated degradation scenario.

Since the first discharge transient is the same as before, the first terminal voltage prediction performance is similar to the previous one (see Figure 4 (a-c)). The predictions in correspondence of the accelerated degradation scenario are shown in Figure 5 (b-d). At the beginning of the second discharge, the algorithm seems not to be capable of identifying the unexpected behavior (Figure 5 (b)): this is due to both the similarity of the two observation curves at the initial discharge stages and to the fact that the RBF model parameters at the beginning of the prediction are the same of those at the end of the previous cycle, as illustrated above. As soon as the RBF-PF algorithm receives more observations, the prediction moves towards the new curve, thus quickly adapting to the new, unexpected discharge dynamics (Figure 5 (c)). Figure 5 (d) shows the predicted RTD (red, dashed line) and its confidence intervals (red, light dotted lines) with respect to the true RTD (blue, solid line). As expected, initially the prediction overestimates the reference RTD, but, eventually, it converges, due to the adaptation capabilities of the RBF-PF.

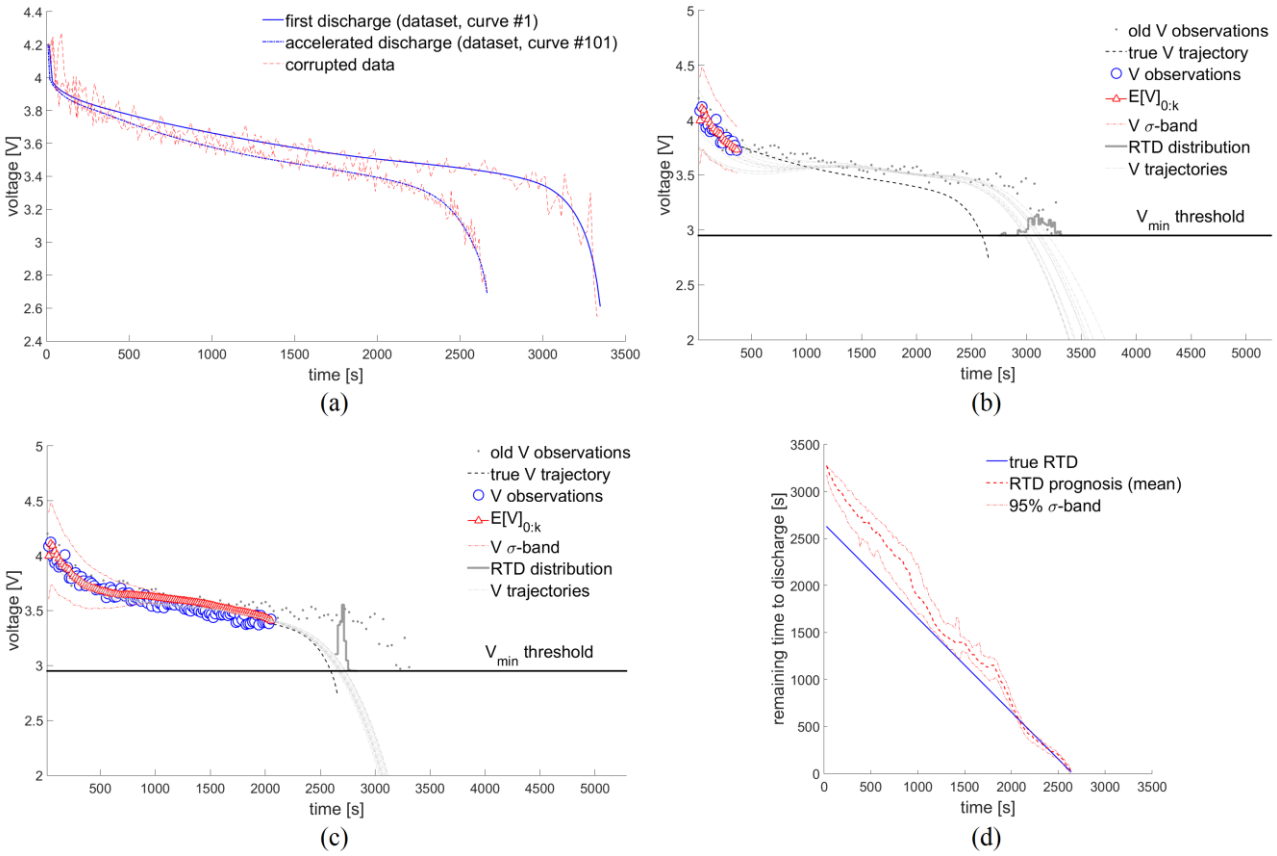

Figure 5. (a) Two discharge curves used to simulate an accelerated aging scenario, the first (longer) referring to the pristine battery, and the second (accelerated discharge) corresponding to the curve #101 of the dataset [6]. (b) Filtered terminal voltage and EOD predictions under accelerated aging after ~500 s and (c) after ~2200 s. (d) RTD prediction under unexpected accelerated aging.

3.4. Analysis of RTD prognostic performance during repeated discharges

We now test the RBF-PF algorithm performances when sequentially applied to a series of repeated discharges of the battery. In order to do so, we refer again to the two scenarios of normal and accelerated aging previously described. First, the normal aging scenario is addressed by sequentially applying the RBF-PF to all the 167 discharge curves available in the dataset. Then, an accelerated aging scenario is simulated by sequentially feeding the RBF-PF algorithm with only the discharge curves #1, 51, 101 and 151, thus assuming that the depletion of the battery occurs after only 4 discharges (EOL). As in the previous Section, the reference transients are corrupted by artificial noise.

Figure 6 (b-d) compares the RTD predictions obtained in the nominal scenario with those obtained in the accelerated aging scenario, in correspondence of the same discharge transients, i.e., curves

#51, #101 and #151 (shown in Figure 6 (a)). The RTD prediction obtained for the transient #1 is obviously very similar to that already shown in Figure 4 (d).

It can be qualitatively seen that the performances in the normal and the accelerated aging scenarios during all the transients considered are very similar, even when reaching the battery EOL. However, in order to quantitatively compare the RTD predictions during the entire battery life cycle, we compute, at each time step *k*, the relative prediction error [32] as:

$$err_k(RTD_k) = \frac{|\widehat{RTD}_k - RTD_k|}{RTD_k}\ 100 \tag{16}$$

where RTD_k is the target remaining time to discharge and $\widehat{RTD}_k$ its prediction. By averaging $err_k(RTD_k)$ over all the time steps of a single transient, we then compute the average relative prediction error, $err(RTD_k) = \sum_{k=0}^{K} err_k(RTD_k)/K$, which is used as a performance metric for comparing the performances over different transients. Figure 7 shows the average relative prediction errors for the normal aging scenario and for the accelerated aging scenario as functions of all the sequential 167 available discharge curves and of the selected 4 discharge curves, respectively. As expected from the qualitative considerations, the two relative errors are comparable. The $err_k(RTD_k)$ in the normal aging scenario appears to increase as the discharge transients predictions progress, up to approximately the transients #80-100. Then, it stabilizes, as also shown by its moving average computed over 15 discharge transients. This increase of the $err_k(RTD_k)$ may be physiologically due to the progressive reduction of the EOD time (and, consequently, of the RTD), which renders relative errors more fluctuating, since the RTD appears at the denominator of (16). The $err(RTD)$ in the accelerated aging scenario shows a similar trend.

Note that the similarity of behaviors in the two scenarios analyzed suggests a satisfactory performance of the RBF-PF algorithm in terms of its capability to adapt to unforeseen situations.

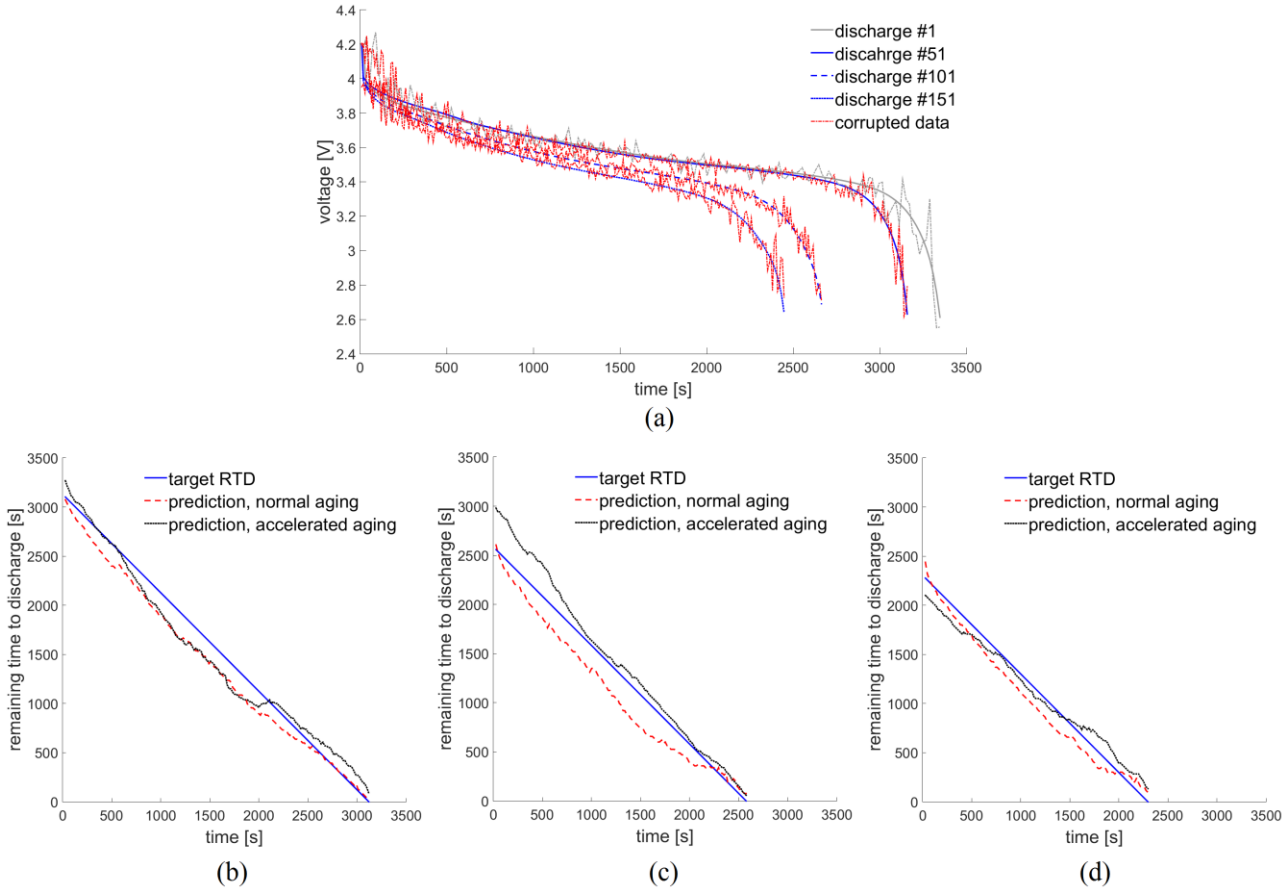

Figure 6. Discharge curves simulating the accelerated case scenario (a), and RTD predictions in case of normal and accelerated aging at different life stages: discharge curve #51 (b), #101 (c) and #151 (d).

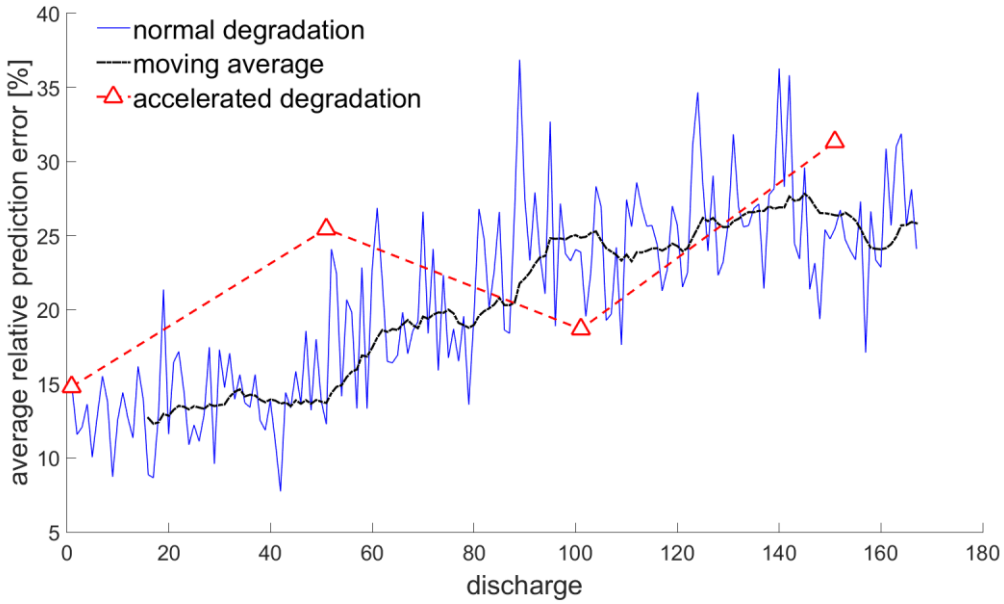

Figure 7. Average relative prediction error (%) of the normal (blue, solid and black, dashed lines) and accelerated aging (red, dashed line and triangles) scenarios.

4. CONCLUSIONS

In this work, we have proposed a novel approach to Lithium-Ion batteries prognosis, which relies on the combination of radial basis function neural networks and particle filtering (RBF-PF algorithm). More specifically, the method has been used to predict the future behavior of the batteries terminal voltage and, consequently, to estimate the end of discharge time and the remaining time to discharge over several discharge cycles. The basic idea was that of exploiting the capabilities offered by PFs of performing real-time identification, estimation and prediction and, at the same time, those offered by RBFs of approximating a wide variety of different dynamics, thus avoiding the use of pre-identified physics-based or empirical models, which, albeit possibly more accurate, lack of the flexibility needed when unforeseen behaviors occur. In fact, the major interest in the application of the RBF-PF algorithm lies in its large on-line adaptation capability, which is a fundamental property whenever a prognostic tool must work outside the limits of its "comfort zone", or, in other words, outside its training region, such as, for example, in presence of sudden, unforeseen changes in the battery dynamic behavior due to failures, errors, or extreme boundary conditions.

The RBF-PF algorithm has been demonstrated on a publicly available dataset containing measurements of the voltage at the terminals of a Li-Ion battery, collected during a charge-discharge laboratory tests conducted at constant current. Before its use, the dataset has been significantly corrupted by adding artificial, Gaussian, but not i.i.d. noise. Nonetheless, the RBF-PF algorithm has shown to be capable of providing satisfactory estimates of the RTD over all the discharge cycles considered, thus being able to capture also the normal aging dynamics, i.e. the slow, but progressive, decrease in the terminal voltage eventually leading to the battery end of life.

Moreover, the adaptation capability of the algorithm has been proven by applying the RBF-PF to an artificial scenario simulating sudden, unforeseen changes in the discharge behavior of the Li-Ion battery, due, for example, to some accelerated aging process. The performance of the algorithm has been satisfactory, with the predicted RTDs being comparable to that shown in the previous normal aging scenario.

These promising results suggest us continuing the investigation of these approach along several lines. First, we are currently studying the possibility of applying the RBF-PF algorithm to the prediction of the remaining useful life, or, alternatively, the end of life time, of Li-Ion batteries. The aim is that of comparing the performances of the RBF-PF algorithm with those of other PF-based

approaches, where physics-based are used; we expect slightly worse prediction results in terms of accuracy, due to the larger approximations inherent the data-driven RBF models, but larger capabilities of adapting to changed dynamics and, possibly, of performing change detection. Then, the RBF-PF algorithm should be also demonstrated for predicting the EOD in Li-Ion batteries discharging at varying, mode-dependent levels of current; possibly, the algorithm should be used to estimate in real-time the SOC and the EOD (especially useful for diagnosis and prognosis in electrical vehicles), on the basis of the available current and terminal voltage observations, which, in general, depend on the battery operating mode. Finally, in order to further increase the adaptation capabilities of the proposed algorithm, we are investigating the possibilities of combining the PF module with other metamodels, such as for example Gaussian process regressions, support vector machines, etc. and/or of adapting on-line also the architecture (e.g., the number of hidden nodes) of the metamodel on the basis of the information gathered.

It is worth noticing that the proposed method is very flexible and, as such, bears the potential to be applied to many other different engineering prognostic problems.

Acknowledgments

The authors would like to thank the Prognostics Center of Excellence at NASA Ames Research Center for the availability of the dataset on battery degradation.

5. REFERENCES

[1]. B. Saha, K. Goebel, 2009. Modeling Li-ion Battery Capacity Depletion in a Particle Filtering Framework. Annual Conference of the PHM society, San Diego, CA.

[2]. K. Goebel, B. Saha, A. Saxena, J.R. Celaya, J.P. Christophersen, 2008. Prognostics in battery health management, IEEE Instrumentation & Measurement Magazine 11(4);33.

[3]. W. He, N. Williard, M. Osterman, M. Pecht, 2011. Prognostics of lithium-ion batteries based on Dempster-Shafer theory and the Bayesian Monte Carlo method. Journal of Power Sources 196(23);10314:10321.

[4]. B. Saha, K. Goebel, S. Poll, J. P. Christophersen, 2009. Prognostics methods for battery health monitoring using a Bayesian Framework. IEEE Transactions on Instrumentation and Measurement 58(2);291:296.

[5]. P. Tagade, K. S. Hariharan, P. Gambhire, S. M. Kolake, T. Song, D. Oh, T. Yeo, S. Doo, 2016. Recursive Bayesian filtering framework for lithium-ion cell state estimation. Journal of Power Sources 306;274:288.

[6]. B. Saha, K. Goebel, 2007. Battery Data Set. NASA Ames Prognostics Data Repository. http://ti.arc.nasa.gov/project/prognostic-data-repository, NASA Ames Research Center, Moffett Field, CA.

[7]. J. Zhang, J. Lee, 2011. A review on prognostics and health monitoring of Li-Ion battery. Journal of Power Sources 196;6007:6014.

[8]. Y. Zou, X. Hu, H. Ma, S. Eben Li. Combined state of charge and state of health estimation over lithium-ion battery cell cycle lifespan for electric vehicle. Journal of Power Sources 273;793-803.

[9]. M. Berecibar, I. Gandiaga, I. Villareal, N. Omar, J. Van Mierlo, P. Van den Bosscher, 2016. Critical review of state of health estimation methods of Li-ion batteries for real applications. Renewable and Sustainable Energy Reviews 56;572:587.

[10]. J. Guo, Z. Li, M. Pecht, 2015. A Bayesian approach for Li-ion battery capacity fade modeling and cycles to failure prognostics. Journal of Power Sources 281;173:184.

[11]. L. Lu, X. Han, J. Li, J. Hua, M. Ouyang, 2013. A review on the key issues for lithium-ion battery management in electric vehicles. Journal of Power Sources 226;272:288.

[12]. P. Baraldi, F. Cadini, F. Mangili, E. Zio, 2013. Model-based and data-driven prognostics under different available information. Probabilistic Engineering Mechanics 32;66-79.

[13]. M. Charkhgard, M. Farrokhi, 2010. State-of-charge estimation for Lithium-Ion Batteries using neural networks and EKF. IEEE Transactions on Industrial Electronics 57(12);4178-4187.

[14]. N. Daroogheh, A. Baniemerian, N. Meskin, K. Khorasani, 2015. A hybrid prognosis and health monitoring strategy by integrating particle filters and neural networks for gas turbine engines. IEEE Conference on Prognostics and Health Management, June 22-25, Austin, TX.

[15]. W. He, N. Williard, M. Osterman, M. Pecht, 2011. Prognostics of lithium-ion batteries based on Dempster-Shafer theory and the Bayesian Monte Carlo method. Journal of Power Soruces 196;10314:10321.

[16]. M. Sauradip, S.K. Sinha, K. Muthukumar, 2001. Estimation of State of Charge of Lead Acid Battery using Radial Basis Function. In: IECON'O1: The 27th Annual Conference of the IEEE Industrial Electronics Society.

[17]. B. Jun, S. Sai, G. Wei, W. Lu, 2012. State of charge estimation of Li-ion batteries in an electric vehicle based on a radial-basis-function neural network. Chin. Phys. B 21(11), 118801.

[18]. H. Gholizade-Narm, M. Charkhgard, 2013. Lithium-ion battery state of charge estimation based on square-root unscented Kalman filter. IET Power Electronics 6(9), 1833-1841.

[19]. J F G de Freitas, M. Niranjan, A. H. Gee, A. Doucet, 2001. Sequential Monte Carlo Methods to Train Neural Network Models. Neural Computation 12, 995-993.

[20]. C. M. Bishop, Neural Networks for Pattern Recognition. Ed. Oxford University Press, 1995.

[21]. I. T. Nabney, NETLAB Algorithms for Pattern Recognition. Ed. London: Springer-Verlag, 2004.

[22]. A. Doucet, S. Godsill, C. Andrieu, 2000. On sequential Monte Carlo sampling methods for Bayesian filtering. Statistics and Computing 10;197:208.

[23]. M. S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, 2002. A Tutorial on Particle filters for Online Nonlinear/non-Gaussian Bayesian Tracking. IEEE Transactions on Signal Processing 50(2);174-189.

[24]. H. Du, N. Zhang, Time series prediction using evolving radial basis function networks with new encoding scheme, Neurocomputing, 71(7–9), 2008, Pag. 1388-1400.

[25]. S. Schwunk, N. Armbruster, S. Straub, J. Kehl, M. Vetter, 2013. Particle filter for the state of charge and state of health estimation of lithium-iron phosphate batteries. Journal of Power Sources 239;705:710.

[26]. A. J. Haug, 2005. A tutorial on Bayesian Estimation and Tracking Techniques Applicable to Nonlinear and Non-Gaussian Processes. MITRE Technical Report MTR 05W0000004.

[27]. Gordon NJ, Salmond DJ, Smith AFM. Novel approach to non-linear/non-Gaussian Bayesian state estimation. IEEE Proceedings F (Radar and Signal Processing) 1993; 140(2):107-13.

[28]. F. Cadini, D. Avram, E. Zio, 2009. Monte Carlo-based filtering for fatigue crack growth estimation. Probabilistic Engineering Mechanics 24, 367-373

[29]. M. E. Orchard, P. Hevia-Koch, B. Zhang, L. Tang, 2013. Risk measures particle filtering-based state-of-charge prognosis in Lithium-Ion batteries. IEEE Transactions on Industrial Electronics 60(11), 5260-5269.

[30]. He et al. Microelectronics Reliability 2013

[31]. M. Orchard, A. Cerda, B.E. Olivares, J.F. Silva, 2012. Sequential Monte Carlo methods for Discharge Time Prognosis in Lithium-Ion Batteries. International Journal of Prognostics and Health Management.

[32]. Saxena A, Celaya J, Saha B, Saha S, Goebel K. Metrics for offline evaluation of prognostic performance. International Journal of Prognostics and Health Management, ISSN 2153-2648, 2010 001.