Journal of Power Sources 285 (2015) 235–246

Contents lists available at ScienceDirect

Journal of Power Sources

journal homepage: www.elsevier.com/locate/jpowsour

Identification of the battery state-of-health parameter from input–output pairs of time series data

Yue Li, Pritthi Chattopadhyay, Asok Ray*, Christopher D. Rahn

Department of Mechanical & Nuclear Engineering, Pennsylvania State University, University Park, PA 16802, USA

HIGHLIGHTS

- Symbolic Dynamic Filtering (SDF) has been used for low-complexity feature extraction.
- Discrete Wavelet Transform (DWT) has been used for data segmentation.
- Algorithms have been validated on experimental data of pairs of current and voltage data from a lead-acid battery.

ARTICLE INFO

Article history:
Received 21 January 2015
Received in revised form
28 February 2015
Accepted 10 March 2015
Available online 12 March 2015

Keywords:
Battery systems
State of health
Symbolic dynamics
Wavelet-based segmentation
Feature extraction

ABSTRACT

As a paradigm of dynamic data-driven application systems (DDDAS), this paper addresses real-time identification of the State of Health (SOH) parameter over the life span of a battery that is subjected to approximately repeated cycles of discharging/recharging current. In the proposed method, finite-length data of interest are selected via wavelet-based segmentation from the time series of synchronized input–output (i.e., current–voltage) pairs in the respective two-dimensional space. Then, symbol strings are generated by partitioning the selected segments of the input–output time series to construct a special class of probabilistic finite state automata (PFSA), called D-Markov machines. Pertinent features of the statistics of battery dynamics are extracted as the state emission matrices of these PFSA. This real-time method of SOH parameter identification relies on the divergence between extracted features. The underlying concept has been validated on (approximately periodic) experimental data, generated from a commercial-scale lead-acid battery. It is demonstrated by real-time analysis of the acquired current–voltage data on in-situ computational platforms that the proposed method is capable of distinguishing battery current–voltage dynamics at different aging stages, as an alternative to computation-intensive and electrochemistry-dependent analysis via physics-based modeling.

1. Introduction

Batteries are being increasingly used for reusable and safe energy storage in various application systems (e.g., electric & hybrid vehicles, renewable energy, power backup systems, and smart grids). A majority of these applications may require large battery packs that contain hundreds and even thousands of battery cells, to satisfy large and dynamic power demands; examples are plug-in electric vehicles and hybrid locomotives. The State of Health (SOH) parameter is a measure of the battery system's ability to store and deliver electrical energy. Knowledge of the SOH parameter enhances preventive maintenance and reduction of life cycle cost through timely recharging and/or replacement of battery cells. Thus, accurate estimation of the SOH parameter is crucial for efficient operation of battery systems, including allocation of power and energy within and among the cells.

The current state-of-the-art for SOH parameter identification can be divided into two broad categories: (a) model-based analysis, and (b) data-driven analysis, both of which need data bases of experimental measurements for validation. The model-based analysis can be further divided into two sub-categories: (i) empirical modeling; and (ii) physics-based modeling. Empirical modeling methods (e.g., impedance measurements for SOH) [1,2] often employ dedicated hardware and/or software and require battery cells for testing. Although these empirical methods may provide good estimates in specific cases, they are time-consuming and cost-prohibitive for general applications. On the other hand, physics-based modeling methods have been extensively used for identification of battery parameters; examples are reduced-order system

[*] Corresponding author.
E-mail address: axr2@psu.edu (A. Ray).

http://dx.doi.org/10.1016/j.jpowsour.2015.03.068
0378-7753/© 2015 Elsevier B.V. All rights reserved.

identification [3], linear switch models [4], and Kalman filtering [5]. However, physics-based modeling methods require thorough knowledge of the electrochemical characteristics of the battery cells to develop appropriate model structures for parameter identification at different operating points of the (nonlinear) battery dynamics.

Compared to model-based approaches, dynamic data-driven methods do not explicitly rely on dedicated hardware/software and physics-based models of battery dynamics. If comparable and adequate training data are available under different operating conditions, data-driven methods are significantly more efficient than model-based methods in terms of computation execution time and memory requirements. Several data-driven methods have also been reported in literature for battery parameter identification. Nuhic et al. [6] used support vector machines (SVM) to estimate SOH under different environmental and load conditions by processing the training and test data based on load collectives. Lin et al. [7] constructed a probability neural network (PNN) and trained it for SOH identification. He et al. [8] proposed a data-driven method based on dynamic Bayesian networks (DBN), in which a DBN was trained for each class of SOH values in the training data. A forward algorithm was then applied to estimate the SOH in real time. Hu et al. [9] conducted capacity estimation of Lithium-ion batteries by k-nearest neighbor (k-NN) regression [10]. As an alternative to model-based analysis, Li et al. [11] have reported a dynamic data-driven method for identification of SOH by using the time series of the battery output voltage at different aging stages. The underlying concept is built upon the theory of symbolic dynamic filtering (*SDF*) [12,13] that extracts the dynamic information from symbolized time series of signals as probabilistic finite state automata (*PFSA*).

The present paper is an extension of the work reported by Li et al. [11] that solely used the time series of battery output (i.e., voltage) for parameter identification. In contrast to the work of Li et al. [11], the present paper makes use of an ensemble of time series *pairs* of synchronized battery inputs (i.e., charging/discharging current) *and* battery outputs (i.e., voltage) for identification of the SOH parameter. These synchronized input–output pairs of time series are simultaneously analyzed to generate symbol strings that, in turn, are used to construct *PFSA* for information compression and feature extraction. The proposed dynamic data-driven method for the SOH parameter identification has been validated using experimental data of a (commercial-scale) lead-acid battery under varying input–output (e.g., current–voltage) conditions. Furthermore, the underlying software can be executed in real time on in-situ computational platforms and its implementation does not require any detailed knowledge of the battery system's electrochemistry.

Significant contributions of the present paper over the work reported by Li et al. [11] and other investigators include the development of a novel dynamic data-driven method of battery parameter identification, which has the following two major advantages:

- *Characterization of the battery dynamics based on the input–output time series via symbolic dynamic analysis*: This representation of the (possibly nonlinear) input–output characteristics is analogous to a transfer function realization without the need for linearization of the battery system dynamics.
- *Extraction of compressed information from symbolized time series of two-dimensional data*: This is achieved by segmentation of the input–output pairs of data in the time–frequency domain and subsequent analysis in the setting of probabilistic finite state automata (*PFSA*).

The paper is organized into four sections, including the present section, and an appendix. Section 2 briefly describes the underlying concept of symbolic dynamic filtering (SDF) upon which the proposed data-driven tool of battery parameter identification is constructed. Section 3 presents the experimental data collection and results of SOH estimation based on the time series of synchronized input–output pairs. Section 4 summarizes and concludes this paper with recommendations for future research. Appendix A lists three algorithms that are used in the main body of the paper.

2. Data-driven parameter identification

This section presents the underlying theory of symbolic dynamics-based data-driven identification of SOH in battery systems.

2.1. Battery State of Health (SOH) parameter

This subsection introduces definitions of pertinent battery parameters at a given ambient temperature [14]. In this paper, the capacity measurement has been used to calibrate the *SOH* at different stages of battery life.

Definition 2.1. (*Battery capacity*) *The capacity* $C(t)$ *of a battery at time t is the maximum charge* (*in units of ampere-hours*) *that can be discharged from a fully charged condition at a rate* $C(t)/30$ (*in units of amperes*).

Definition 2.2. (*SOH*) *Let a new battery be put into service at time* t_0. *The state of health SOH*(t) *of the* (*possibly used*) *battery at the current time t, where* $t \geq t_0$, *is defined to be the ratio of the battery capacities at time epochs t and* t_0, *i.e.*,

$$SOH(t) = \frac{C(t)}{C(t_0)} \quad \text{for all} \quad t \geq t_0 \tag{1}$$

Remark 2.1. *It is noted that* $SOH \in [0,1]$ *for all time* $t \geq t_0$, *where* t_0 *is the time of putting a new battery into service. In this paper, the battery is assumed to fail to meet the operational requirement, i.e., it needs to be replaced, when the SOH parameter is less than* 80%.

2.2. Wavelet-based time series segmentation

This subsection presents segment extraction of dynamic input–output pairs by conducting wavelet-based segmentation from the database of synchronized system input-outputs. Specifically, selection of the scale range of wavelet basis and the thresholding on wavelet coefficients are addressed. The wavelet-based analysis provides pertinent information of the signal simultaneously in the time domain and the frequency domain [15].

For a given wavelet basis function $\psi(t)$, the scaled and translated child wavelets are defined as [16]:

$$\psi_{\alpha,\tau}(t) = \sqrt{\frac{1}{\alpha}}\psi\left(\frac{t-\tau}{\alpha}\right) \tag{2}$$

where $\alpha \in (0, \infty)$ is the scale of wavelet transform and $\tau \in (-\infty, \infty)$ is the time shift, and $\psi \in \mathbf{L}_2(\mathbb{R})$ is such that $\int_{-\infty}^{\infty} \psi(t)\mathrm{d}t = 0$ and the norm $||\psi|| = 1$.

The continuous wavelet transform (CWT) of a function $x(t)$ at a scale α is represented as

$$\tilde{x}(\alpha, \tau) = \int_{-\infty}^{\infty} \overline{\psi}_{\alpha,\tau}(t)x(t)\mathrm{d}t \tag{3}$$

where $\overline{\psi}$ indicates the complex conjugate of ψ; and the distinction between $\overline{\psi}$ and ψ vanishes for real-valued wavelet basis functions. It is also noted that the time series of a continuous-time signal $x(t)$ is its representation in the discrete-time domain and hence the integral in Eq. (3) is replaced by summation over a (finite) time series $\{x[n]\}$, $n = 1,2,\cdots,N$ to obtain the Discrete Wavelet Transform (DWT) at a scale α_m and time shift τ_ℓ as:

$$\tilde{x}[\alpha_m,\tau_\ell] = \sum_{n=1}^{N} \overline{\psi}_{\alpha_m,\tau_\ell}[n]x[n] \tag{4}$$

Every wavelet basis function can be associated with a purely periodic signal of frequency, called center frequency f_c, which maximizes the Fourier transform of the wavelet modulus [17]. Consequently, the relationship among frequency points φ_m, the associated scales α_m for a given wavelet basis function and a sampling period Δ of the time series is given as:

$$\alpha_m = \frac{f_C}{\varphi_m \Delta} \tag{5}$$

where Δ is the sampling period of the analyzed signal. The wavelet basis function for analysis is selected based on the time–frequency characteristics of analyzed signals.

For a time series $\{x[n]\}$, $n = 1,2,\cdots, N$, the discrete Fourier transform is obtained as:

$$\hat{x}[k] = \sum_{n=1}^{N} e^{-2\pi ikn/N}x[n] \quad \text{for } k = 1,2,\cdots,N \tag{6}$$

Remark 2.2. *In general, the Fourier transform $\hat{x}[k]$ of a signal is a complex number but its power spectral density $|\hat{x}[k]|^2$ is a non-negative real number.*

Following Parseval's theorem, the energy of the time series $x[n]$ can be expressed in the discrete-time and discrete-frequency domains [18] as:

$$\sum_{n=1}^{N} |x[n]|^2 = \sum_{k=1}^{N} |\hat{x}[k]|^2 \tag{7}$$

Since the summand on the right hand side of Eq. (7) can be interpreted as a probability histogram describing the energy distribution of the signal at frequency points k, the energy spectral density of a signal $x[n]$ is defined as:

$$S_x[k] = |\hat{x}[k]|^2 \quad \forall k \in \{1,2,\cdots,N\} \tag{8}$$

The steps of the wavelet-based segmentation procedure, depicted in Fig. 1, are as follows.

- Step 1: *Collection of (finite) time series data $x[n]$, $n = 1,2,\cdots,N$ for a given sampling interval Δ.* It is noted that the length N of the time series is user-selectable.
- Step 2: *Computation of the discrete Fourier transform $\hat{x}[k]$, $k = 1,2,\cdots,N$ and the corresponding PSD $S_x[k]$, $k = 1,2,\cdots,N$.* This step follows Eqs. (6) and (8).
- Step 3: *Identification of the frequency points of interest, φ_m, $m = 1,2,\cdots,M$.* In this step, the set $\{S_x[\varphi_m]\}$ is formed in terms of the M points with highest values of the PSD $S_x[k]$, $k = 1,2,\cdots,N$. [Note: M is usually significantly less than N].
- Step 4: *Computation of the corresponding wavelet scales α_m, $m = 1,2,\ldots, M$.* This step follows Eq. (5) with the corresponding frequency points φ_m and the central frequency f_c of the chosen wavelet basis function.

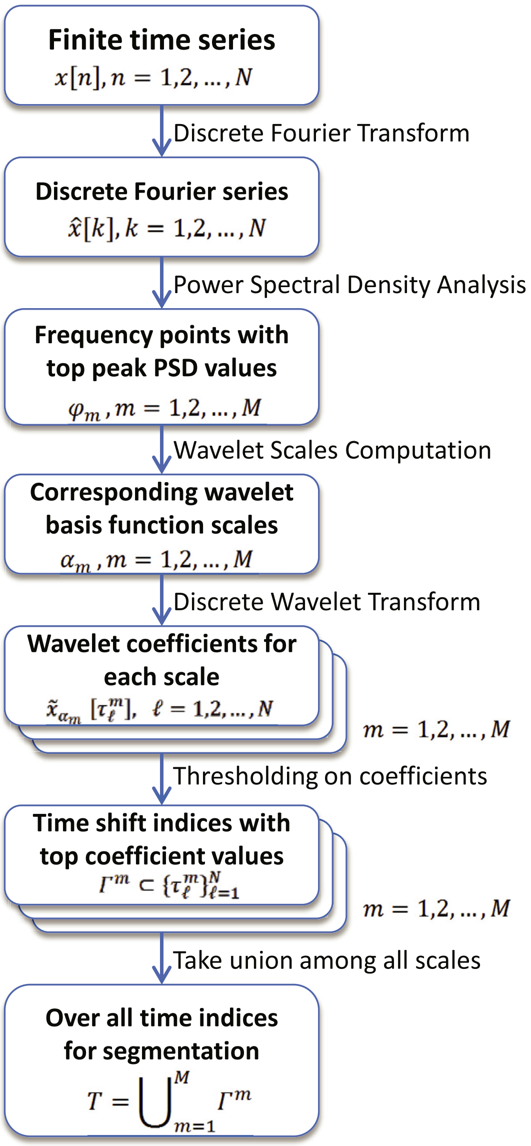

Finite time series
$x[n], n = 1,2,\ldots,N$
Discrete Fourier Transform
Discrete Fourier series
$\hat{x}[k], k = 1,2,\ldots,N$
Power Spectral Density Analysis
Frequency points with top peak PSD values
$\varphi_m, m = 1,2,\ldots,M$
Wavelet Scales Computation
Corresponding wavelet basis function scales
$\alpha_m, m = 1,2,\ldots,M$
Discrete Wavelet Transform
Wavelet coefficients for each scale
$\tilde{x}_{\alpha_m}[\tau_\ell^m],\ \ell = 1,2,\ldots,N$
$m = 1,2,\ldots,M$
Thresholding on coefficients
Time shift indices with top coefficient values
$\Gamma^m \subset \{\tau_\ell^m\}_{\ell=1}^N$
$m = 1,2,\ldots,M$
Take union among all scales
Over all time indices for segmentation
$T = \bigcup_{m=1}^{M} \Gamma^m$

Fig. 1. Procedure for wavelet-based segmentation.

- Step 5: *Identification of the set of segmented time indices Γ^m for each scale α_m.* In this step, the set of time shift indices $\Gamma^m \subset \{\tau_\ell^m\}_{\ell=1}^N$ are selected as the top ξ_T fraction of the values of the wavelet coefficients $\tilde{x}_{\alpha_m}(\tau_\ell^m) = \tilde{x}[\alpha_m,\tau_\ell^m]$, $\ell = 1,2,\cdots,N$ at a scale α_m.
- Step 6: *Identification of the set, T, of all segmented time indices.* This step is completed by taking the union of the sets of $\Gamma^{m\prime}$s among all scales α_m, $m = 1,2,\ldots, M$ as $T = \cup_{m=1}^{M} \Gamma^m$.

The wavelet-based segmentation procedure is summarized as Algorithm 1 in Appendix A.

2.3. Maximum entropy partitioning and symbolization

This subsection addresses the partitioning in the input–output space of single-input single-output (SISO) systems for symbolization of 2-dimensional time series data. Time series of input–output pairs are partitioned into a mutually exclusive and exhaustive set of finitely many segments, where a symbol string is generated by assigning a unique symbol to each segment of the input–output space.

Quasi-stationarity of SISO systems is assumed such that the system behavior is statistically stationary at the fast time scale of the process dynamics, while there exists observable non-stationary behavior evolving at a slow time scale. The notion of two time scales and their significance are discussed in the context of symbolization in a recent publication [13].

In this paper, the Maximum Entropy Partitioning (MEP) [19] of the time-series data has been adopted to construct the symbol alphabet Σ and to generate symbol strings. In this partitioning, the information-rich regions of the data set are partitioned finer and those with sparse information are partitioned coarser to maximize the Shannon entropy of the generated symbol string from the reference data set. The MEP procedure is presented as Algorithm 2 in Appendix A.

In this paper, a pair of time series that represent input and output data is partitioned in the associated two-dimensional space to construct a symbolic string. Four alternative types of partitions have been used in the input–output space:

- Partition Type 1 (Cartesian coordinates): First partition in the input axis (e.g., abscissa), and then partition in the output axis (e.g., ordinate) at individual input segments.
- Partition Type 2 (Cartesian coordinates): First partition in the output axis, and then partition in the input axis at individual output segments.
- Partition Type 3 (Polar coordinates): First partition in the magnitude, and then partition in the phase at individual magnitude segments.
- Partition Type 4 (Polar coordinates): First partition in the phase, and then partition in the magnitude at individual phase segments.

Fig. 2 depicts the underlying concept of symbolization of a 2-dimensional time series, where each segment in the left half plot is labeled by a unique symbol and Σ denotes the alphabet of all these symbols. The segment, visited by the time series plot takes a symbol value from the alphabet Σ. For example, having $\Sigma = \{\alpha, \beta, \gamma, \delta\}$ in Fig. 2, a time-series $x_0x_1x_2\cdots$ generates a string of symbols in the symbol space as: $s_0s_1s_2\cdots$, where each s_i, $i = 0,1,2,\cdots$, takes a symbol value from the alphabet Σ. This mapping is called symbolic dynamics as it attributes a (physically admissible) symbol string to the dynamical system starting from an initial state (for example, see the symbol string at the top right half plot of Fig. 2).

2.4. Symbolic dynamic filtering (SDF)

This subsection briefly describes the underlying concept of symbolic dynamic filtering (SDF) upon which the proposed dynamic-data-driven tool of battery parameter identification is constructed; SDF encodes the behavior of (possibly nonlinear) dynamical systems from the observed time series by symbolization and construction of state machines (i.e., probabilistic finite state automata (PFSA)) [12]. This is followed by computation of the state emission matrices that are representatives of the evolving statistical characteristics of the battery dynamics.

The core assumption in the SDF analysis for construction of probabilistic finite state automata (PFSA) from symbol strings is that the symbolic process under both nominal and off-nominal conditions can be approximated as a Markov chain of order D, called the D-Markov machine, where D is a positive integer. While the details of the D-Markov machine construction are given in Refs. [12,13], the pertinent definitions and their implications are succinctly presented below.

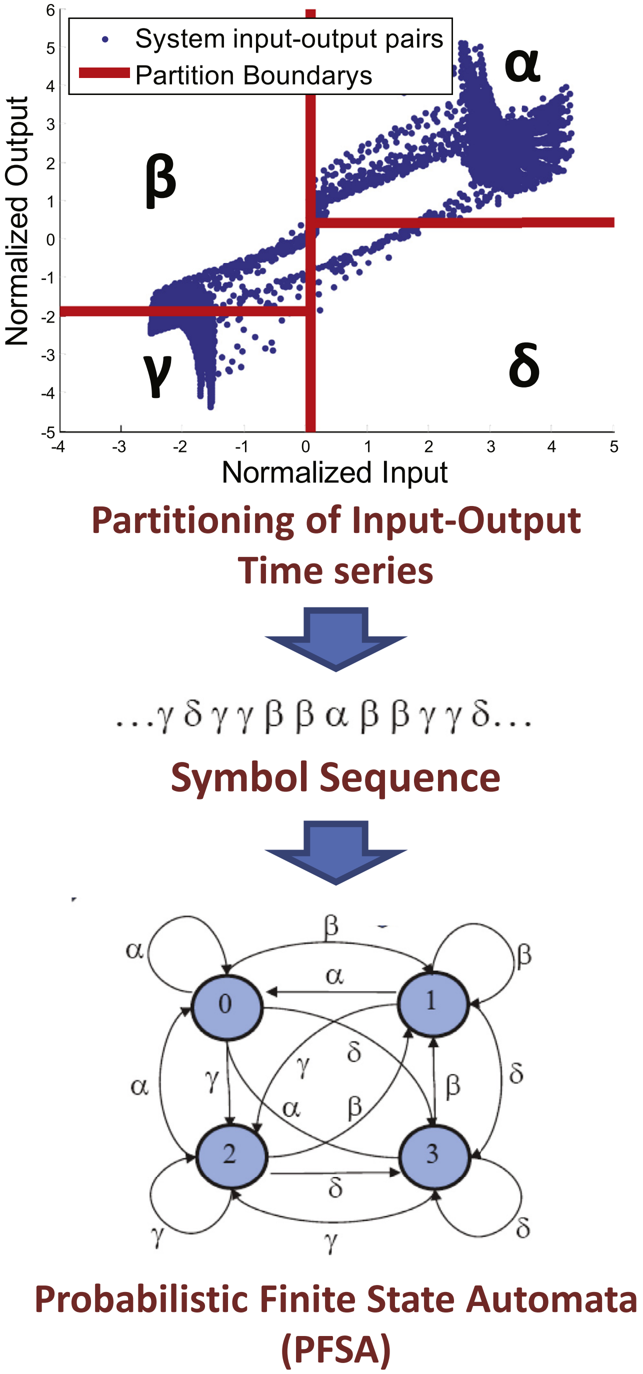

System input-output pairs
Partition Boundarys
Normalized Output
Normalized Input
6
5
4
3
2
1
0
-1
-2
-3
-4
-5
-4
-3
-2
-1
0
1
2
3
4
5
α
β
γ
δ
Partitioning of Input-Output Time series
...γ δ γ γ β β α β β γ γ δ...
Symbol Sequence
0
1
2
3
α
β
γ
δ
Probabilistic Finite State Automata (PFSA)

Fig. 2. Construction of finite state automata (FSA) from time series.

Definition 2.3. *(DFSA) A deterministic finite state automaton (DFSA) is a 3-tuple $G = (\Sigma, Q, \delta)$ where:*

1. *Σ is a non-empty finite set, called the symbol alphabet, with cardinality $1 < |\Sigma| < \infty$;*
2. *Q is a non-empty finite set, called the set of states, with cardinality $1 < |Q| < \infty$;*
3. *$\delta: Q \times \Sigma \rightarrow Q$ is the state transition map;*

and $\Sigma^\star$ is the collection of all finite-length strings with symbols from Σ including the (zero-length) empty string ε, i.e., $|\varepsilon| = 0$.

Remark 2.3. *It is noted that Definition 2.3 does not make use of an initial state, because the purpose here is to work in a statistically stationary setting, where no initial state is required as explained by Adenis et al.* [20].

Definition 2.4. (*PFSA*) *A probabilistic finite state automaton* (*PFSA*) *is constructed upon a DFSA* $G=(\Sigma, Q, \delta)$ *as a pair* $K=(G, \pi)$, *i.e., the PFSA K is a 4-tuple* $K=(\Sigma, Q, \delta, \pi)$, *where*:

1. Σ, Q, *and* δ *are the same as in* Definition 2.3;
2. π:$Q\times\Sigma\rightarrow[0,1]$ *is the probability morph function that satisfies the condition* $\sum_{\sigma\in\Sigma}\pi(\sigma|q)=1\ \forall q\in Q$. *Denoting* π_{ij} *as the probability of occurrence of a symbol* $\sigma_j\in\Sigma$ *at the state* $q_i\in Q$, *the* ($|Q|\times\Sigma|$) *probability morph matrix* (*also known as emission matrix*) *is obtained as* $\Pi=[\pi_{ij}]$.

Definition 2.5. (*D-Markov*) *A D-Markov machine* [12] *is a PFSA in which each state is represented by a* (*nonempty*) *finite string of D symbols where*

- *D, a positive integer, is the depth of the Markov machine*;
- *Q is the finite set of states with cardinality* $|Q|\leq|\Sigma|^D$. *The states are represented by equivalence classes of symbol strings of maximum length D, and each symbol in the sting belongs to the alphabet* Σ;
- δ:$Q\times\Sigma\rightarrow Q$ *is the state transition map that satisfies the following condition if* $|Q|=|\Sigma|^D$: *There exist* $\alpha, \beta\in\Sigma$ *and* $s\in\Sigma^\star$ *such that* $\delta(\alpha s, \beta)=s\beta$ *and* αs, $s\beta\in Q$.

Remark 2.4. *It follows from* Definition 2.5 *that a D-Markov chain is treated as a statistically stationary stochastic process* $S=\cdots s_{-1}s_0s_1\cdots$, *where the probability of occurrence of a new symbol depends only on the last D symbols, i.e.,* $P[s_n|\cdots s_{n-D}\cdots s_{n-1}]=P[s_n|s_{n-D}\cdots s_{n-1}]$.

The construction of a D-Markov machine is based on: (i) state splitting that generates symbol blocks of different lengths according to their relative importance; and (ii) state merging that assimilates histories from symbol blocks leading to the same symbolic behavior. Words of length D on a symbol string are treated as the states of the D-Markov machine before any state-merging is executed. Thus, on an alphabet Σ, the total number of possible states becomes less than or equal to $|\Sigma|^D$; and operations of state merging may significantly reduce the number of states [13].

The PFSA states represent different combinations of blocks of symbols on the symbol string. In the graph of a PFSA, the directional edge (i.e., the emitted event) that interconnects a state (i.e., a node) to another state represents the transition probability between these states. Therefore, the "states" denote all possible symbol blocks (i.e., words) within a window of certain length, and the set of all states is denoted as $Q=\{q_1,q_2,\ldots,q_{|Q|}\}$ and $|Q|$ is the number of (finitely many) states. The procedure for estimation of the emission probabilities is presented next.

Given a (finite-length) symbol string S over a (finite) alphabet Σ, there exist several PFSA construction algorithms to discover the underlying irreducible PFSA model K of S, such as causal-state splitting reconstruction (CSSR) [21], D-Markov [12,19], and Compression via Recursive Identification of Self-Similar Semantics (CRISSiS [22]). All these algorithms start with identifying the structure of the PFSA $K\triangleq(Q,\Sigma,\delta,\pi)$. To estimate the state emission matrix, a $|Q|\times|\Sigma|$ count matrix C is constructed and each element c_{kj} of C is computed as:

$$c_{kj}\triangleq1+N(q_k,\sigma_j) \qquad (9)$$

where $N(q_k, \sigma_j)$ denotes the number of times that a symbol σ_j is generated from the state q_k upon observing the symbol string S. *The maximum a posteriori probability (MAP) estimates of emission probabilities for the PFSA K* are computed by frequency counting as

$$\widehat{\pi}(\sigma_j|q_k)\triangleq\frac{c_{kj}}{\sum_\ell c_{k\ell}}=\frac{1+N(q_k,\sigma_j)}{|\Sigma|+\sum\limits_\ell N(q_k,\sigma_\ell)} \qquad (10)$$

The rationale for initializing each element of the count matrix C to 1 is that if no event is generated at a state $q\in Q$, then there should be no preference to any particular symbol and it is logical to have $\widehat{\pi}(\sigma|q)=(1/|\Sigma|)\forall\sigma\in\Sigma$, i.e., the uniform distribution of event generation at the state q. The above procedure guarantees that the PFSA, constructed from a (finite-length) symbol string, must have an (elementwise) strictly positive morph map Π and that the state transition map δ in Definitions 2.3 and 2.5 *is a total function*.

Having computed the emission probabilities $\widehat{\pi}(\sigma_j|q_k)$ for $j\in\{1,2,\cdots,|\Sigma|\}$ and $k\in\{1,2,\cdots,|Q|\}$, the estimated emission probability matrix of the PFSA is obtained as:

$$\widehat{\Pi}\triangleq\begin{bmatrix}\widehat{\pi}(\sigma_1|q_1) & \ldots & \widehat{\pi}\left(\sigma_{|\Sigma|}\middle|q_1\right)\\ \vdots & \ddots & \vdots\\ \widehat{\pi}\left(\sigma_1\middle|q_{|Q|}\right) & \cdots & \widehat{\pi}\left(\sigma_{|\Sigma|}\middle|q_{|Q|}\right)\end{bmatrix}. \qquad (11)$$

The procedure of symbolic dynamic filtering for feature extraction, which makes use of the estimated emission probability matrix $\widehat{\Pi}$, is presented as Algorithm 3 in Appendix A.

With an appropriate choice of partitioning, it is ensured that the resulting Markov chain model satisfies the ergodicity conditions [23]; in other words, under statistically stationary conditions, the probability of every state being reachable from any other state within finitely many transitions must be strictly positive. The statistics of the time series at an epoch t^k, represented by the estimated emission matrix $\widehat{\Pi}^k$, change to $\widehat{\Pi}^\ell$ at another epoch t^ℓ. Accordingly, the estimated emission matrices $\widehat{\Pi}^k$ and $\widehat{\Pi}^\ell$ are treated as feature vectors that are postulated to have the imbedded information on the dynamical system at the epochs t^k and t^ℓ, respectively. Thus, the (quasi-)stationary emission matrix $\widehat{\Pi}$ serves as the "feature" vector extracted from the time series and is used for pattern classification in the sequel. The evolution of the battery dynamics is captured as the divergence of the feature vector as defined below.

Definition 2.6. (*Feature Divergence*) *Let* $\widehat{\Pi}^0$ *be the feature vector at the epoch* t^0, *which is treated as a reference* (*e.g., fully charged or healthy*) *condition of the battery, and let* $\widehat{\Pi}^k$ *be the feature vector at the current time epoch* t^k. *Then, the* (*scalar*) *feature divergence at an epoch* t^k *is expressed as*:

$$m_k\triangleq d\left(\widehat{\Pi}^k,\widehat{\Pi}^0\right) \qquad (12)$$

where $d(\bullet,\bullet)$ *is an appropriate metric and there are several choices for this metric* (*for example, see* [12]).

If the feature divergence m is intended to be used as an indicator of a battery condition (e.g., a parameter θ related to SOH), then it is highly desirable to have a linear relationship between m and θ. From this perspective, the range of linearity between these two entities is obtained as a statistical model in terms of the coefficient of determination as defined below.

Definition 2.7. (*Coefficient of Determination*) *As a measure of how well the linear least squares fit,* $\widehat{\theta}_k=a+bm_k$, *performs as a predictor of an output* θ *in terms of the input m, where the scalars a and b are the intercept and slope of the linear fit, the coefficient of determination*

(*COD*) *is defined as*:

$$COD = 1 - \frac{\sum_{i=1}^{n}\left(\theta_i - \widehat{\theta}_i\right)^2}{\sum_{i=1}^{n}\left(\theta_i - \overline{\theta}\right)^2} \tag{13}$$

where $\overline{\theta} \triangleq (1/n)\sum_{i=1}^{n}\theta_i$ *is the average of the output data. In this paper,* $\{m_k\}$ *represents a sequence of the feature divergence and* $\{\theta_k\}$ *is the corresponding sequence of* $(1 - SOH)$*. The coefficient of determination COD ranges from* 0 *to* 1 *and* $COD = 1$ *implies a perfect linear fit.*

2.5. Summary of the feature extraction procedure

The flow chart in Fig. 3 summarizes the basic procedures of the proposed information compression and feature extraction method for two-dimensional time series of synchronized input–output data. First, the raw data are normalized such that they have the properties of zero-mean and unit-variance. Second, the dynamic information of interest is extracted from the normalized data via wavelet-based segmentation in the time–frequency domain. Third, the concept of Maximum Entropy Partitioning (MEP) [19] is adopted to partition the segmented data set into multiple states in the 2-dimensional input–output data space. Then, symbol strings are generated from the partitioned data sets and probabilistic finite state automata (PFSA) are constructed from the symbol strings, where a feature is represented as the emission matrix of a PFSA.

3. Experimental validation

This section validates the algorithms of battery parameter identification with an ensemble of experimental data that have been collected from a lead-acid battery. The results of SOH identification are presented.

3.1. Data acquisition and processing

The experiments made use of fresh (12 V AGM VRLA with 56 A h capacity) lead-acid batteries that were charged/discharged according to given input current profiles at room temperature. In this way, an ensemble of synchronized time-series of the input charge/discharge current and output voltage responses has been collected at the sampling frequency of 1 Hz. A typical input current profile for this experiment is shown in Fig. 4. The duration of an input profile is ~150 h, which consists of three capacity measurement cycles followed by 25 duty cycles.

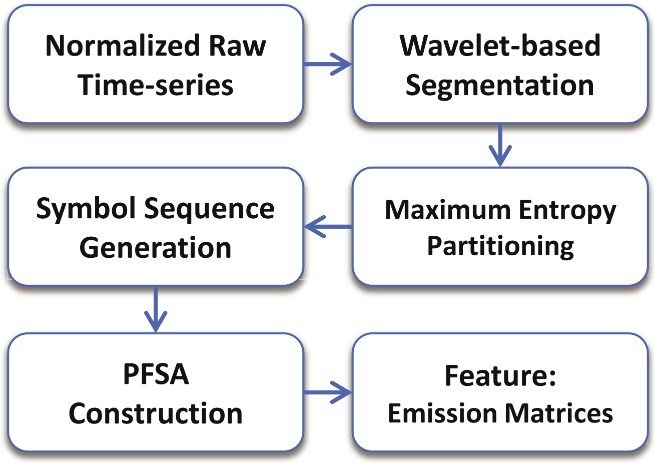
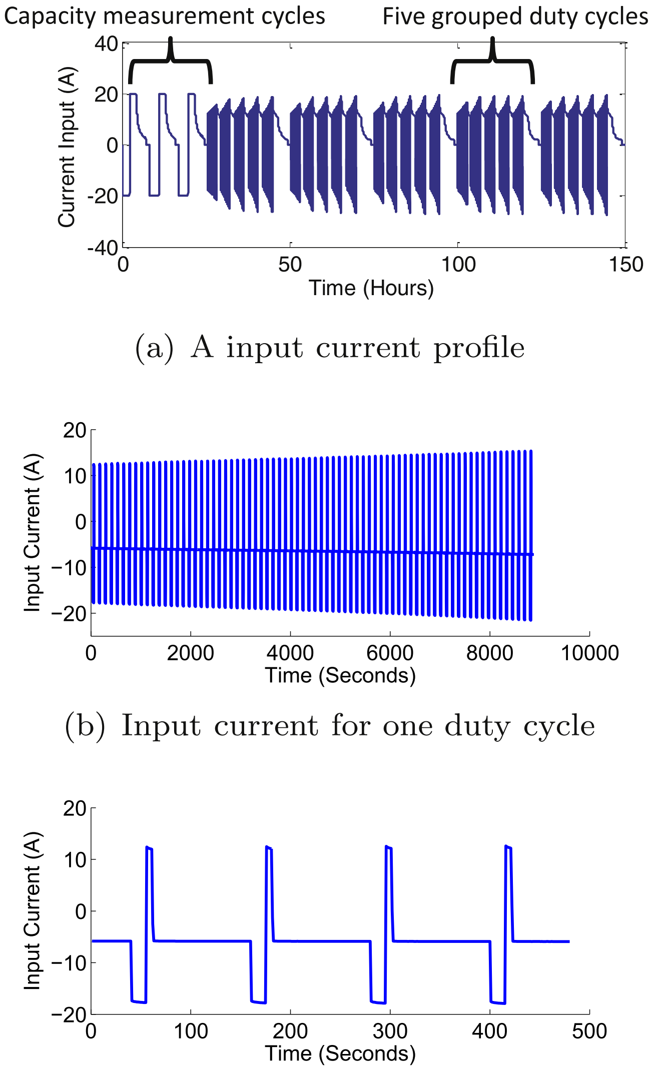

Normalized Raw Time-series
Wavelet-based Segmentation
Symbol Sequence Generation
Maximum Entropy Partitioning
PFSA Construction
Feature: Emission Matrices

Fig. 3. The flow chart for the proposed method of feature extraction.



(a) A input current profile

(b) Input current for one duty cycle

(c) "Hotel-Pulses" cycles

Fig. 4. A typical profile of input current data for the battery experiments.

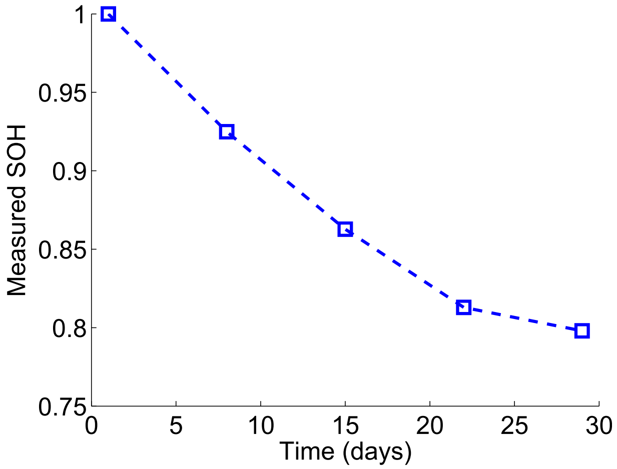


Fig. 5. Degradation of maximum capacity of the battery.

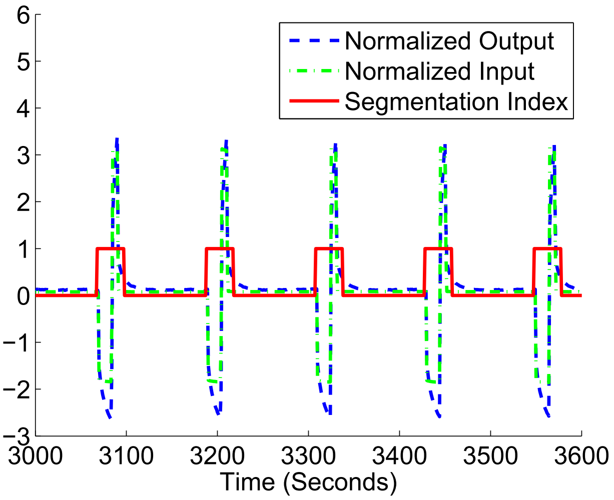


Fig. 6. Wavelet-based segmentation for normalized input–output.

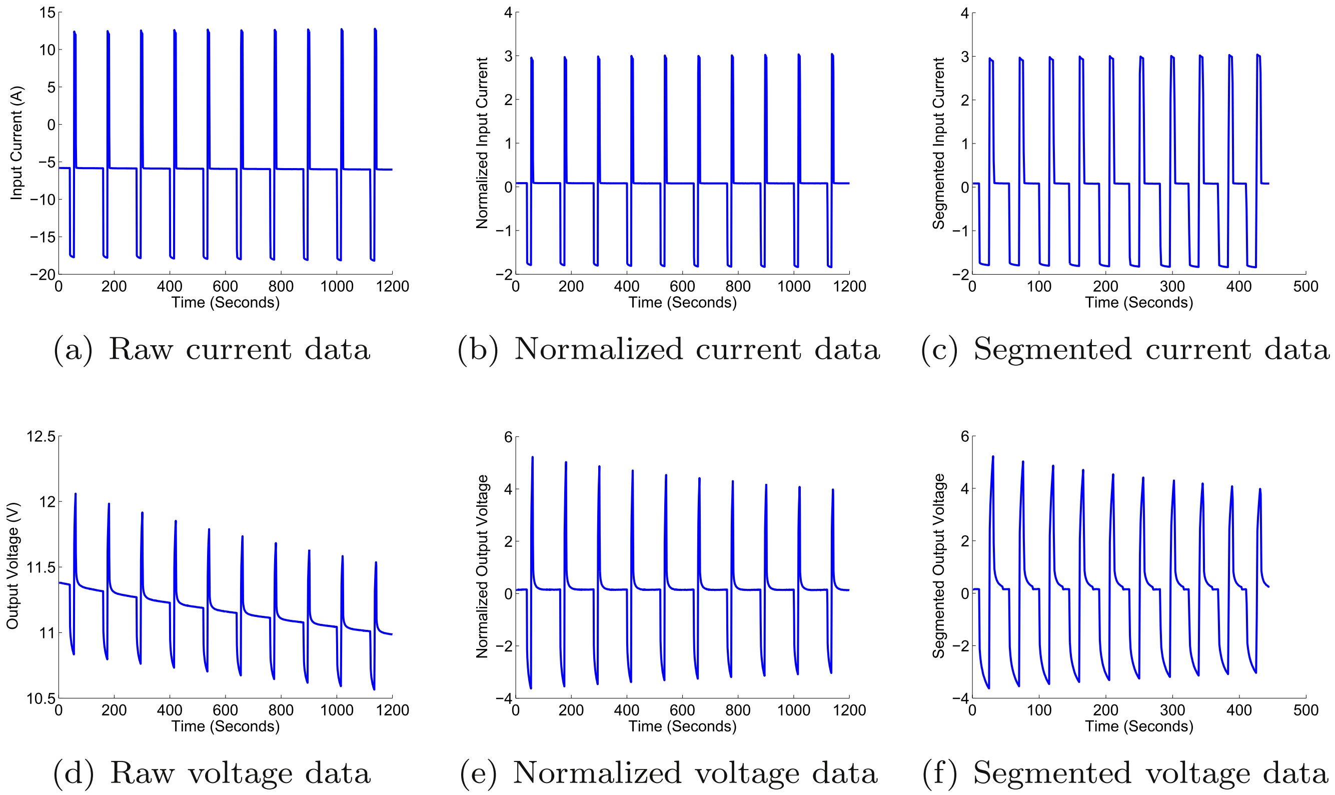
A capacity measurement cycle is a slow, full discharge/charge cycle, where the battery is fully discharged followed by a full charge. In this constant-current constant-voltage (CCCV) operation of the experiments, a fresh battery is initially discharged by a constant current of −20 A for approximately 2 h. The battery is then re-charged with a constant current of 20 A until its output reaches 13.8 V; this voltage is held constant for the next 3 h with gradually decreasing charging current. The maximum capacity at that time is measured by integrating the current during the period of charging. Accordingly, maximum battery capacities are computed from these measurement cycles; the mean value of these battery capacities is considered as the nominal maximum capacity at that particular time. There are, in total, five input profiles of similar pattern, which are applied to the battery during the entire experiment. The degradation profile of battery SOH (see Definition 2.2) is obtained as depicted in Fig. 5.

Total 25 duty cycles are divided into groups of five. The transition time between two consecutive groups is ~6 h for charging to full capacity, while the transition time between two consecutive duty cycles in the same group is ~1.2 h; at this point, the battery is not fully charged. Each duty cycle lasts ~2.7 h, as depicted in Fig. 4(b), which is composed of ~75 “Hotel-Pulse” cycles. Each “Hotel-Pulses” cycle is of duration 120 s and consists of a “hotel” load (i.e., relatively steady discharge due to “hotel” needs like lighting and electrical appliances) and a discharge pulse followed by a charge (i.e., regeneration) pulse, as shown in Fig. 4(c). The amplitudes of the “hotel” load and the discharging & charging pulses are not usually monotonically increasing in a duty cycle, which make each duty cycle slightly different from others. This pattern of input cycles largely simulates a real-time working condition for an electric locomotive.

The raw time series data of (synchronized) input current and output voltage are pre-processed by individual normalization, followed by wavelet-based segmentation of the output-voltage time series. While segmentation extracts the relevant segments of normalized data based on the information of their frequency contents, normalization involves time translation and (down or up)-scaling of the raw data for conversion into zero-mean unit variance time series as:

$$x[n] = \frac{x_{raw}[n] - \mu_N[n]}{\sigma_N[n]} \tag{14}$$

where $\mu_N[n]$ is the mean value and $\sigma_N[n]$ is the standard deviation of the time series over a time span of N data points centered at the time index n. The experimental data of each duty cycle (i.e., both input current and output voltage) are normalized in this moving



(a) Raw current data (b) Normalized current data (c) Segmented current data

(d) Raw voltage data (e) Normalized voltage data (f) Segmented voltage data

Fig. 7. An example of pre-processing for synchronized input current and output voltage data.

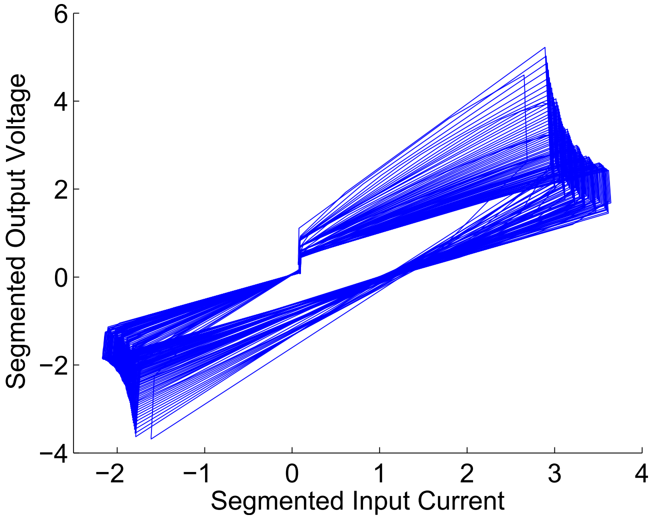


Fig. 8. Normalized input–output pairs of synchronized data for one duty cycle.

average fashion. First, the data set is smoothed by the moving average method with a shift window of 240 s (i.e., 240 consecutive data points). Then, each element of that data set is divided by its own standard deviation. Finally, the normalized data turn out to be a zero-mean and unit-variance time series. Plots (b) and (e) in Fig. 7 are two examples of normalized current and voltage data.

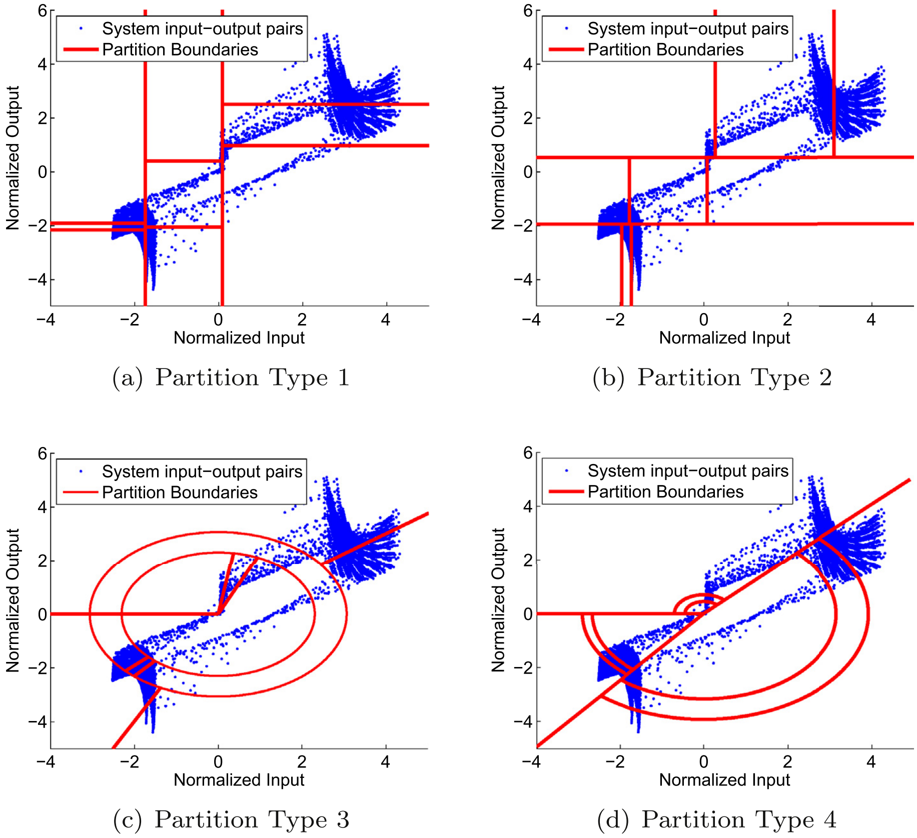
Fig. 6 presents the segmentation process in which relevant information of normalized input and output data is contained over small windows of time. The segmentation index of 1 in Fig. 6 represents the time windows containing relevant data, while the segmentation index of 0 is not of interest because the behavior of the battery system does not convey any useful information in these time windows. Since a "Hotel-Pulses" cycle has duration 120 s (i.e., consisting of 120 consecutive data points), the dynamic characteristics of the tested battery are identified from the segments of discharging and charging pulses for a time span of ~45 s in each "Hotel-Pulses" cycle. The information-relevant segments are extracted from the normalized time series by using the wavelet-based segmentation algorithm (see Section 2.2). Plots (c) and (f) in Fig. 7 present the results of an example of segmentation based on voltage data as applied to both (synchronized) input current and output voltage. The data length after segmentation is much shorter than the original data length.

Fig. 8 exemplifies typical profiles of segmented pairs of normalized input–output data from one duty-cycle, where the charging transitions between two consecutive duty cycles are excluded during data pre-processing.

3.2. Results and discussion

The SOH parameter is identified based on the SDF features extracted from pre-processed input–output (i.e., current–voltage) time-series pairs that were collected during the duty cycles. Particularly, SOH identification relies on the divergence between the extracted features (see Eq. (12)). The parameters for SDF analysis (see Section 2.4) of time series data have been chosen as follows.

- The alphabet size $|\Sigma|$ is chosen to be the same for both dimensions of maximum entropy partitioning in the input–output space. As depicted in Fig. 9, partition types 1 and 2



(a) Partition Type 1

(b) Partition Type 2

(c) Partition Type 3

(d) Partition Type 4

Fig. 9. Maximum Entropy Partitioning (MEP) in the input–output space corresponding to the alphabet size $|\Sigma| = 3$ for both dimensions: The first partition is in one dimension, followed by the partition in second dimension for each segment of the first dimension. (a) First x-axis and then y-axis; (b) first y-axis, then x-axis; (c) first magnitude and then phase, and (d) first phase and then magnitude.

are obtained in Cartesian coordinates and the difference between them lies in the partitioning order of the abscissa (x-axis) and the ordinate (y-axis). Partition types 3 and 4 are obtained in polar coordinates, where magnitude and phase are the used.

- The depth in the *D*-Markov machine is set at $D = 1$, which implies that $|\Sigma| = |\mathscr{Q}|$.
- The distance function $d(\bullet,\bullet)$ for computation of the feature divergence in Eq. (12) for SOH identification is chosen to be the City Block distance (i.e., the absolute sum or 1-norm) for as defined below.

$$d\left(\widehat{\Pi}^k,\widehat{\Pi}^0\right)\triangleq\sum_{j=1}^{|\mathscr{Q}|}\sum_{i=1}^{|\Sigma|}\left|\widehat{\Pi}^k(i,j)-\widehat{\Pi}^0(i,j)\right| \qquad (15)$$

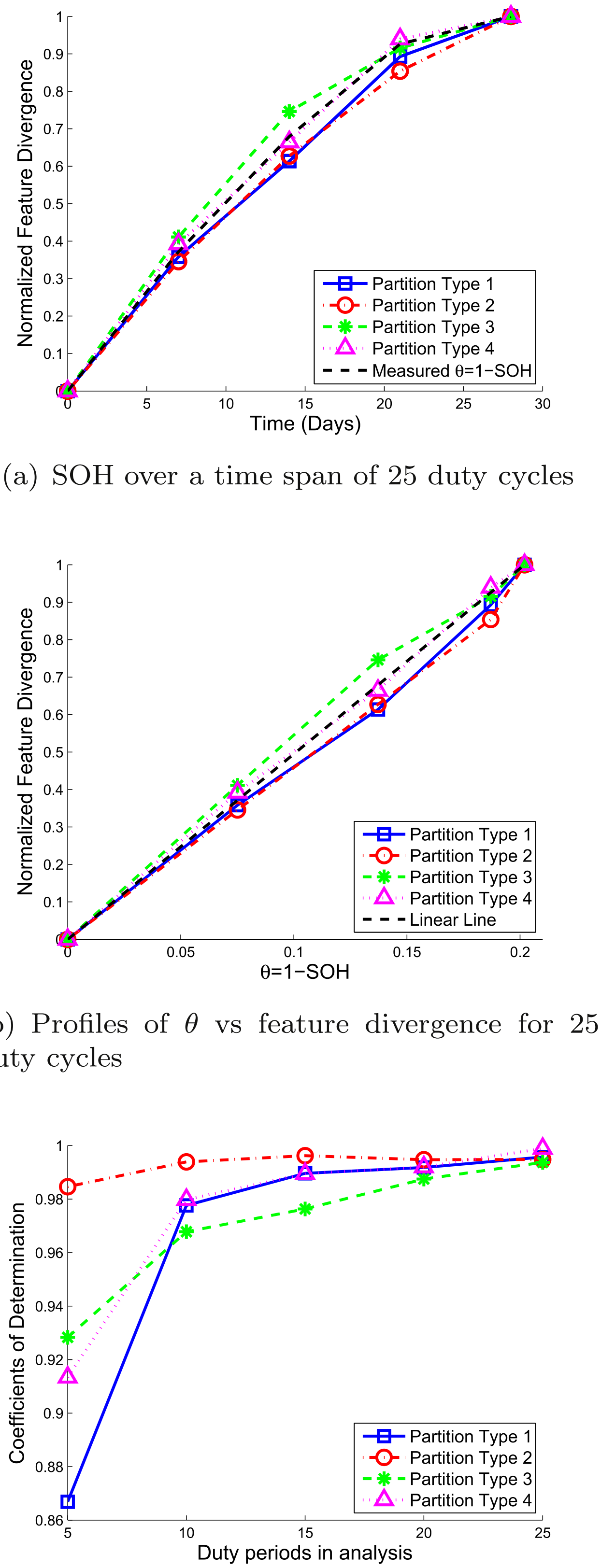
The estimated SOH for real-time data is measured by feature divergence (see Eq. (12) and Eq. (15)) between the reference feature (i.e., extracted from the time series of the new battery) and the feature vector, extracted from the current time series, which has the same length as the reference pattern. Fig. 10(a) presents SOH identification for input–output analysis under different partition types described in previous section; in each case, 25 duty cycles are taken into consideration. The abscissa in the plot of Fig. 10(b) represents the variable $\theta\triangleq 1-SOH$, and the ordinate represents the computed feature divergence. The approximate linear relationship between these two variables are measured in terms of the coefficient of determination (COD) (see Eq. (13) in Definition 2.7), which shows the goodness-of-fit of the data points with a linear structure in the least-squares sense. Closer the COD value is to one, better is the linear relationship between the feature divergence and variable θ that is equivalent to performance of the proposed method for SOH identification. Fig. 10(c) demonstrates the COD at different analyzed data length for the alphabet size $|\Sigma| = 4$ for both input and output. For short data lengths (e.g., 2.7 h or one duty period), the COD values of different partition types tend to be divergent. As the length of analyzed data is increased, the performance of different partition types is improved in the sense that they are more closely clustered. Similar results have been obtained at different choices of alphabet sizes $|\Sigma|$.

Table 1 presents the CODs between normalized feature divergence and the parameter $\theta=1-SOH$ for SOH estimation in the analysis of 25 duty periods. These measurements of linearity show that the feature divergences over the battery life are largely linear (i.e., COD values are very close to one) in terms of the corresponding battery SOH parameter θ for different choices of alphabet size and partition type. Therefore, it is reasonable to conclude that the feature divergence (that is computed in real time) consistently and accurately represents the SOH dynamics during the entire battery life; this representation is valid for different parameter settings of the algorithm.

3.3. Computational cost

This subsection presents a statement of the computational costs (i.e., execution time and memory requirement) of the SOH parameter estimation algorithm. In this paper, all results have been generated on a single core 3.6 Ghz CPU with 12 GB RAM.

Table 2 presents the execution time and memory requirement of the proposed method for SOH estimation for different durations of operation and analysis. In each case, the CPU time and memory requirement have been obtained for the feature extraction process at all stages of battery aging under the assigned duty periods. The variations in the computational cost for different choices of $|\Sigma| = |\Sigma_1| \times |\Sigma_2|$ and partition types are much less compared to those for lengths of time series data. The execution time and



(a) SOH over a time span of 25 duty cycles

(b) Profiles of θ vs feature divergence for 25 duty cycles

(c) Coefficients of determination for different partition types

Fig. 10. Results of SOH estimation with alphabet size $|\Sigma| = 4$ for both input and output.

Table 1
Coefficient of Determination (COD) for SOH estimation at alphabet size $|\Sigma| = |\Sigma_1| \times |\Sigma_2|$.

Alphabet size $\lvert\Sigma_1\rvert \times \lvert\Sigma_2\rvert$	3 × 3	4 × 4	5 × 5	6 × 6
Partition Type 1, COD	0.9770	0.9957	0.9815	0.9928
Partition Type 2, COD	0.9854	0.9948	0.9954	0.9964
Partition Type 3, COD	0.9822	0.9937	0.9917	0.9962
Partition Type 4, COD	0.9637	0.9989	0.9939	0.9894

Table 2
Execution time and required memory for different analyzed data length.

Operating duration in analysis	Execution time (sec)	Memory requirement (MB)
5 duty periods	7 ~ 8	10 ~ 13
10 duty periods	14 ~ 15	15 ~ 18
15 duty periods	20 ~ 22	21 ~ 23
20 duty periods	27 ~ 28	26 ~ 30
25 duty periods	33 ~ 35	33 ~ 36

memory requirement of the proposed software for battery SOH estimation is apparently insignificant even on a small computational platform (e.g., commercially available laptop computers).

As pointed out in the previous subsection, the performance of the proposed algorithm could be improved with longer time series data. Therefore, an appropriate choice of the data length depends on the battery system dynamics that also determine the sampling frequency of data acquisition. In real-life applications, such as hybrid electric vehicles, it is expected that larger data sets would be available for SOH estimation within the same testing period, because the sampling frequency of current/voltage should be larger than 1 Hz, which has been used in the experiments reported this paper.

4. Summary, conclusions, & future work

The proposed SOH identification method is built upon the concept of Symbolic Dynamic Filtering (SDF) [12], where time series of synchronized battery input (current) and output (voltage) data have been analyzed for information compression and feature extraction. The procedure of SOH identification relies on the divergence between the features extracted from long-term observations. Performance of the proposed SDF-based SOH identification method has been validated on experimental data of a commercial-scale lead-acid battery for operations over a long time span. The pertinent conclusions drawn from the work reported in this paper are summarized below:

- Symbolic Dynamic Filtering (SDF), as a low-complexity feature extraction tool, is capable of real-time execution on in-situ computational platforms (e.g., sensor nodes of individual battery systems). It provides a computationally efficient and electrochemistry-independent method of estimating the battery SOH parameter as an alternative to physics-based modeling analysis.
- Extracted SDF features capture the information, embedded in the input–output time series, for SOH identification. The underlying software can be implemented on a sensor network.
- By a synergistic combination of synchronized input current and output voltage, the proposed identification method is seen to be superior to the previous work of output-only analysis [11] in terms of robustness to battery aging and fluctuations in charging/discharging current.

While there are several issues that need to resolved by further theoretical and experimental research, authors suggest the following topics of future research for application of the proposed method of battery SOH identification in practice:

- SDF analysis with $D \geq 1$ in D-Markov machines [13] to accommodate longer memory of synchronized symbolic input–output time series.
- Extension of proposed method to achieve robust performance for changing patterns of the input profiles (e.g., for stochastic nature of the charging and discharging current inputs) that, for example, represent random usage profiles for electric vehicles.
- Investigation of the impact of temperature changes on battery dynamics for SOH identification.
- Validation of the proposed method on other types of batteries (e.g., Li-ion and Ni-MH) as well as for different discharge or charge cycle patterns.

Acknowledgments

The work reported in this paper has been supported in part by the U.S. Air Force Office of Scientific Research under Grant No. FA9550-12-1-0270. Any opinions, findings and conclusions or recommendations expressed in this publication are those of the authors.

Appendix A. Algorithms

This appendix introduces three algorithms that are used in the main body of the paper.

Algorithm 1 Wavelet based Segmentation

```
Require: Perform the following:
    Select the sampling time period $\Delta$.
    Collect finite-length time series $x[n], n = 1, \ldots, N$.
    Select a wavelet basis function $\psi$ with central frequency $f_c$; and the (scalar) threshold parameter
    $\xi_T$ for selection of wavelet coefficients for segmentation.
Ensure: From the collected time series, execute the following:
    Compute power density $S_x(f)$ of the time series $x[n]$ for the frequency window $f \in [0, \frac{1}{2\Delta}]$ at
    $N$ discrete points (see Eqs. (6) and (8)).
    Select the frequency points $\{\varphi_m\}_{m=1}^{M}$ corresponding to the local peaks of the energy density.
for $m = 1$ to $M$ do
    Compute the corresponding wavelet scale $\alpha_m$ by substituting frequency points $\varphi_m$ in Eq. (5).
    Obtain the wavelet coefficients for different translated versions of the wavelet coefficients,
    $\tilde{x}_{\alpha_m}(\tau_\ell^m) = \tilde{x}(\alpha_m, \tau_\ell^m)$, at scales $\alpha_m$ (see Eq. (4)).
    Identify the time indices $\Gamma^m = \{\tau_\ell^m | \tilde{x}_{\alpha_m}(\tau_\ell^m)$ in the top $\xi_T$ percentage of values in
    $\{\tilde{x}_{\alpha_m}(\tau_\ell^m)\}_{\ell=1}^{N}\}$.
end for
The selected time series segment is the union of time indices from all scales $T = \bigcup_{m=1}^{M} \Gamma^m$
```

Algorithm 2 Maximum entropy partition for 2-dimensional data

```
Require: A 2-dimensional string $X[n] = \left[x_1[n]\ x_2[n]\right]$ for $n = 1, 2, 3, \cdots, N$ of synchronized and
    normalized time series data set; Alphabet size $|\Sigma_1|$ for the first coordinate of the 2-dimensional
    data and alphabet size $|\Sigma_2|$ for the second coordinate of the 2-dimensional data.
Ensure: Partition vector $\wp_1 \in \mathbb{R}^{|\Sigma_1|+1}$ for first coordinate data; Partition Matrix $\wp_2 \in$
    $\mathbb{R}^{|\Sigma_1|\times(|\Sigma_2|+1)}$ for the 2-dimensional data set.
Assign $\wp_1(1) = -\infty$, i.e., the minus infinity
Assign $\wp_2(m, 1) = -\infty$, for $m = 1, 2, \cdots, |\Sigma_1|$
Assign $\wp_1(|\Sigma_1| + 1) = \infty$, i.e., the positive infinity
Assign $\wp_2(m, |\Sigma_2| + 1) = \infty$, for $m = 1, 2, \cdots, |\Sigma_1|$
Sort the data string $x_1$ in the ascending order as $x_1^s$
Let $K = length(x_1^s)$
for $i = 1$ to $|\Sigma_1|$ do
    if $i \neq |\Sigma_1|$ then
        $\wp_1(i+1) = x_1^s\left[ceil\left(\frac{i \times K}{|\Sigma_1|}\right)\right]$
    end if
    Define $x_2^i \triangleq \{x_2[n] | \wp_1(i) < x_1[n] < \wp_1(i+1)\}$
    Sort the data string $x_2^i$ in the ascending order as $x_2^{is}$
    Let $L = length(x_2^{is})$
    for $j$=1 to $|\Sigma_2| - 1$ do
        $\wp_2(i, j+1) = x_2^{is}\left[ceil\left(\frac{j \times L}{|\Sigma_2|}\right)\right]$
    end for
end for
```

Algorithm 3 Symbolic dynamic filtering for feature extraction

```
Require: Symbolic strings of length $N$ obtained at $I$ different operating conditions of system under
    analysis: $S_i = \{s_i^1, s_i^2, s_i^3, ..., s_i^N\}, i = 0, 1, \ldots, I-1$; the alphabet size $|\Sigma| = |\Sigma_1| \times |\Sigma_2|$; the
    depth $D$ of Markov machine; and number of states $|Q|$, where $|Q| \leq |\Sigma|^D$.
Ensure: Extracted emission matrices $\widehat{\Pi}_i \in \mathbb{R}^{|\Sigma|^D \times |\Sigma|}, i = 0, 1, \ldots, I-1$ for each symbol string
    and the feature divergences $\{m_i\}_{i=0}^{I-1}$.
Initialize $\Sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_{|\Sigma|}\}$, and $Q = \{q_1, q_2, \ldots, q_{|Q|}\}$.
for $i = 0$ to $I - 1$ do
    for $k = 1$ to $|Q|$ do
        for $j = 1$ to $|\Sigma|$ do
            Count the number of event that symbol $\sigma_j$ occurs after symbol (combination) of $q_k =$
            $\{\sigma_1^k \ldots \sigma_{|D|}^k\}$, denoted as $N(\sigma_j, q_k)$, from the symbol string $S_i$.
        end for
        for $j = 1$ to $|\Sigma|$ do
            compute the estimated emission probability $\hat{\pi}(\sigma_j | q_k)$ (see Eq. (10)).
        end for
    end for
    Construct the estimated emission matrix $\widehat{\Pi}_i$ for symbol string $S_i$ (see Eq. (11)).
    Compute the feature divergence $m_i$ (see Eq. (12)) based on a user-selectable distance func-
    tion.
end for
```

References

[1] H. Blanke, O. Bohlen, S. Buller, R. De Doncker, B. Fricke, A. Hammouche, D. Linzen, M. Thele, D. Sauer, Impedance measurements on lead–acid batteries for state-of-charge, state-of-health and cranking capability prognosis in electric and hybrid electric vehicles, J. Power Sources 144 (2) (2005) 418–425.

[2] T. Okoshi, K. Yamada, T. Hirasawa, A. Emori, Battery condition monitoring (bcm) technologies about lead–acid batteries, J. Power Sources 158 (2) (2006) 874–878.

[3] Z. Shen, Control-oriented Modeling, State-of-charge, State-of-health, and Parameter Estimation of Batteries (PhD thesis), The Pennsylvania State University, 2013.

[4] Z. Shen, C.D. Rahn, Model-based state-of-charge estimation for a valve-regulated lead-acid battery, in: ASME 2012 5th Annual Dynamic Systems and Control Conference Joint with the JSME 2012 11th Motion and Vibration Conference, 2012, pp. 279–286. American Society of Mechanical Engineers.

[5] C. Sankavaram, B. Pattipati, A. Kodali, K. Pattipati, M. Azam, S. Kumar, M. Pecht, Model-based and data-driven prognosis of automotive and electronic systems, in: Automation Science and Engineering, 2009. CASE 2009. IEEE International Conference on, Pp. 96–101, IEEE, 2009.

[6] A. Nuhic, T. Terzimehic, T. Soczka-Guth, M. Buchholz, K. Dietmayer, Health diagnosis and remaining useful life prognostics of lithium-ion batteries using data-driven methods, J. Power Sources 239 (0) (2013) 680–688.

[7] H.-T. Lin, T.-J. Liang, S.-M. Chen, Estimation of battery state of health using probabilistic neural network, IEEE Trans. Ind. Inf. 9 (2) (2013) 679–685.

[8] Z. He, M. Gao, G. Ma, Y. Liu, S. Chen, Online state-of-health estimation of lithium-ion batteries using dynamic bayesian networks, J. Power Sources 267 (0) (2014) 576–583.

[9] C. Hu, G. Jain, P. Zhang, C. Schmidt, P. Gomadam, T. Gorka, Data-driven method based on particle swarm optimization and k-nearest neighbor regression for estimating capacity of lithium-ion battery, Appl. Energy 129 (0) (2014) 49–55.

[10] C. Bishop, Pattern Recognition, Springer, New York, NY, USA, 2006.

[11] Y. Li, Z. Shen, A. Ray, C.D. Rahn, Real-time estimation of lead-acid battery parameters: A dynamic data-driven approach, J. Power Sources 268 (2014) 758–764.

[12] A. Ray, Symbolic dynamic analysis of complex systems for anomaly detection, Signal Process. 84 (July 2004) 1115–1130.

[13] K. Mukherjee, A. Ray, State splitting and merging in probabilistic finite state automata for signal representation and analysis, Signal Process. 104 (November 2014) 105–119.

[14] C.D. Rahn, C.-Y. Wang, Battery Systems Engineering, John Wiley, 2012.

[15] G. Kaiser, A Friendly Guide to Wavelets, Springer, 2010.

[16] G. Kaiser, A Friendly Guide to Wavelets, Springer, 2010.

[17] M. Misiti, Y. Misiti, G. Oppenheim, J.-M. Poggi, Wavelet Toolbox, The

MathWorks Inc., Natick, MA, 1996.

[18] M. Vetterli, J. Kovačević, Wavelets and Subband Coding, vol. 87, Prentice Hall PTR Englewood Cliffs, New Jersey, 1995.

[19] V. Rajagopalan, A. Ray, Symbolic time series analysis via wavelet-based partitioning, Signal Process. 11 (November 2006) 3309–3320.

[20] P. Adenis, Y. Wen, A. Ray, An inner product space on irreducible and synchronizable probabilistic finite state automata, Math. Control Signals Syst. 23 (1) (January 2012) 281–310.

[21] C. Shalizi, K. Shalizi, Blind construction of optimal nonlinear recursive predictors for discrete sequences, in: AUAI'04: Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, AUAI Press, Arlington, VA, USA, 2004, pp. 504–511.

[22] I. Chattopadhyay, Y. Wen, A. Ray, S. Phoha, Unsupervised inductive learning in symbolic sequences via recursive identification of self-similar semantics, in: Preprints Proceedings of American Control Conference, San Francisco, CA, USA, June–July 2011, pp. 125–130.

[23] A. Berman, R. Plemmons, Nonnegative Matrices in the Mathematical Sciences, SIAM Publications, Philadelphia, PA, USA, 1994.