Hybrid Electrochemical Modeling with Recurrent Neural Networks for Li-ion Batteries

Saehong Park, Dong Zhang, Scott Moura

***Abstract*—This paper examines a hybrid battery system modeling framework, where data-oriented recurrent neural network (RNN) and first-principle electrochemical battery model are combined. The data-driven RNN model captures unmodeled dynamics in the electrochemical model. We specifically study a simple RNN model called an Elman network, which has feedback loops in the hidden layer. We analyze and prove convergence of the weight errors for a class of Elman networks and learning update laws. In simulation, we compare our proposed hybrid battery model with reduced electrochemical battery models. The results demonstrate that the proposed hybrid approach outperforms other reduced electrochemical battery models in most scenarios.**

***Keywords*—Single Particle Model, Recurrent Neural Network, Elman Network, Online Learning**

I. Introduction

Lithium-ion batteries have emerged as the primary choice for electrified transportation and energy storage system due to their high energy density, no memory effect, and low self-discharge. In order to ensure battery safety and performance, an advanced battery management system (BMS) is typically deployed with the batteries. Over the past decade, a vast body of literature has developed in BMS research [1], [2]. One may categorize BMS research into model development and algorithm design. In this paper, we focus on control-oriented battery models.

Battery models for BMS applications can be categorized into two groups: equivalent circuit models (ECMs), and electrochemical models. ECMs have relatively simple structures to represent the input-output behavior of batteries using circuit elements, such as resistors and capacitors. However, ECMs do not directly capture the physical phenomena inside the battery, such as lithium transport, solid-electrolyte interphase dynamics, and degradation mechanisms. Electrochemical models directly incorporate diffusion, intercalation, and electrochemical kinetics. Although these models can accurately explain the internal behavior of the battery, their mathematical structure is relatively complex for observer designs, such as state of charge (SoC) and state of health (SoH) estimation. Consequently, reduced order models are generally selected for state estimator design. There are several widely studied reduced order electrochemical models in this field. The simplest is single particle model (SPM), where electrolyte concentration is assumed constant in space and time [3]. Papers by Santhanagopalan et al. [4] and Di Domenico et al. [5] study SPM-based Kalman filters for SOC estimation. More recently, [6] studies nonlinear observers for the SPM. The main limitation of the SPM is that it neglects electrolyte dynamics, which play important role at high charge/discharge C-rates[1]. For this reason, researchers have recently considered a SPM with electrolyte dynamics (SPMe) [7]. A provably convergent partial differential equation (PDE)-based state estimation technique based on the SPMe can be found in [2]. Tanim et al. [8] augment the SPMe with Arrhenius relations for the model parameters to account for temperature variation, abbreviated as SPMeT. A Luenberger state observer based on this SPMeT model is derived in [9].

In parallel, computational learning theory in artificial intelligence has been rapidly applied to real-world engineering applications, such as self-driving cars, pattern recognition, and robotics. Artificial neural networks are a fundamental technique in machine learning for modeling an environment from data. In this paper, we focus on *dynamic* neural networks, which were first proposed for dynamic system identification in the early 1990's [10]. Dynamic neural networks, a.k.a. recurrent neural networks (RNNs), can capture dynamical behavior by learning from input-output data. In battery applications, some published works have studied RNN models. The authors of [11] use an adaptive RNN to predict remaining useful life of lithium-ion batteries. They use the Levenberg-Marquardt algorithm to optimize the network parameters, then estimate the electrolyte resistance and charge transfer resistance in the ECM to infer power fade. In [12], the ECM is converted into an RNN model that captures SoC information and model parameters. The authors conclude that the RNN can accurately describe the charge-discharge phenomena of a lead-acid battery. Similarly, a RNN-based SoH indicator for lithium-ion batteries is proposed in [13]. They also predict variations in capacity and resistance of an ECM reformulated into a RNN. This literature suggests that RNNs can play a useful role in predicting battery dynamics, especially aging mechanisms. However, this set of literature is completely based on the ECM. Moreover, it lacks a theoretical stability analysis. This motivates us to explore the role of RNNs in combination with electrochemical models, and study stability for various classes of RNNs.

In this paper, we propose a hybrid battery model that integrates the SPM and RNN. Note, the SPM is valid for low C-rates, but produces erroneous voltage predictions at high C-rates where electrolyte dynamics play an increased role [14]. This paper focuses on a simple RNN, known as the Elman

Saehong Park, Dong Zhang, and Scott Moura are with the Energy, Controls and Applications Lab (eCAL) at the University of California, Berkeley, CA 94720, USA ({sspark, dongzhr, smoura}@berkeley.edu)

[1]C-rate is a normalized measure of electric current that enables comparisons between different sized batteries. Mathematically, the C-rate is defined as the ratio of current, I, in Amperes [A] to a cell's nominal capacity, in Ampere-hours [Ah]. For example, if a battery has a nominal capacity of 2.5 Ah, then C-rates of 2C, 1C, and C/2 correspond to 5 A, 2.5 A, and 1.25 A, respectively.

network [15], which originally emerged in cognitive science. Contrary to the conventional electrochemical modeling approach of extending models based on first principles, we augment the SPM with Elman networks to improve voltage prediction accuracy. To the best of the authors' knowledge, the combination of electrochemical and data-driven models have not been previously explored. We summarize our key contributions as follows. First, we propose a hybrid modeling framework that incorporates electrochemical and RNN models – namely the SPM and Elman networks. We consider the real-time recurrent learning (RTRL) algorithm for learning Elman network weights online. Second, we suggest a special case of Elman networks that guarantees stability in the sense of Lyapunov. Third, we demonstrate that our hybrid approach often outperforms other reduced electrochemical models that do not apply learning structures.

The paper is organized as follows. Section II describes the Single Particle Model. In Sections III–IV, the proposed Elman network and RTRL algorithm are introduced and analyzed for stability. Section V presents simulation results and quantitative performance metrics to illustrate the efficacy of the hybrid modeling framework. In Section VI, we conclude our work with remarks and future work.

II. Electrochemical Battery Model

In this section we describe the electrochemical model considered in this paper. Electrochemical models provide deep insight on the evolution of internal battery dynamics. These models are generally derived from the Doyle-Fuller-Newman (DFN) model, which is based on intercalation, diffusion, and electrochemical kinetics modeled by partial differential equations (PDEs). Although the DFN model can accurately predict internal states, its mathematical structure is comprised of nonlinear PDEs, ODEs in space, and nonlinear algebraic equations. Due to this intricacy, many researchers have studied model reduction techniques on the DFN model. The most commonly used reduced-order model is the single particle model (SPM), which idealizes each electrode as a single spherical porous particle while neglecting electrolyte dynamics, as shown in Fig. 1. Specifically, the electrolyte concentration is approximated as constant in space and time, and this restriction causes errors which will be discussed later in this section. The intercalation process and mass transport is modeled by a linear diffusion PDE over spherical coordinates:

$$\frac{\partial c_s^-}{\partial t}(r,t) = D_s^-\left[\frac{2}{r}\frac{\partial c_s^-}{\partial r}(r,t) + \frac{\partial^2 c_s^-}{\partial r^2}(r,t)\right], \quad (1)$$

$$\frac{\partial c_s^+}{\partial t}(r,t) = D_s^+\left[\frac{2}{r}\frac{\partial c_s^+}{\partial r}(r,t) + \frac{\partial^2 c_s^+}{\partial r^2}(r,t)\right], \quad (2)$$

with Neumann boundary conditions,

$$\frac{\partial c_s^-}{\partial r}(0,t) = 0, \quad \frac{\partial c_s^-}{\partial r}(R_s^-,t) = \frac{I(t)}{D_s^- Fa^- AL^-}, \quad (3)$$

$$\frac{\partial c_s^+}{\partial r}(0,t) = 0, \quad \frac{\partial c_s^-}{\partial r}(R_s^+,t) = \frac{I(t)}{D_s^+ Fa^+ AL^+}. \quad (4)$$

The Neumann boundary conditions at $r = R_s^-$ and $r = R_s^+$ represent that the molar flux of lithium entering / exiting the electrode is proportional to the input current $I(t)$. For clarity, all parameters related to the SPM are listed in Table I.

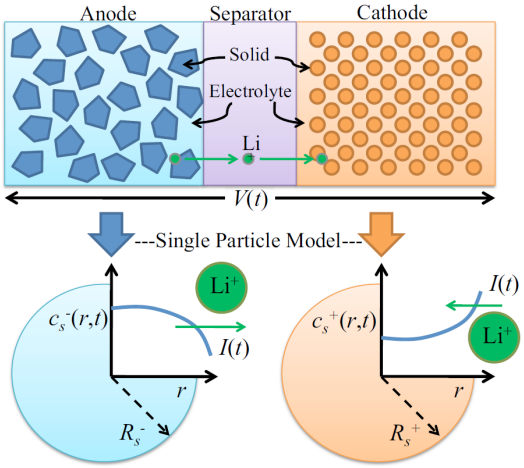

Figure 1: Schematic of single particle model for Li-ion battery.

Symbol	Description	Units
A	Cell cross sectional area	m^2
a^j	Specific interfacial surface area	m^2/m^3
c_s^j	Concentration in solid phase	mol/m^3
c_{ss}^j	Concentration at particle surface	mol/m^3
$c_{s,max}^j$	Max concentration in solid phase	mol/m^3
D_s^j	Diffusion coefficient in solid phase	m^2/m^3
F	Faraday's constant	C/mol
I	Input current	A
i_0^j	Exchange current density	V
L^j	Electrode thickness	m
R	Universal gas constant	J/mol-K
R_f	Lumped current collector resistance	Ω
R_s^j	Particle radius	m
r	Radial coordinate	m
T	Cell temperature	K
t	Time	seconds
U^j	Equilibrium potential	V
V	Output voltage	V
α^j	Anodic/cathodic transfer coefficient	-

Table I: Single Particle Model Parameters.

Note that superscript j in Table I indicates anode, seperator and cathode such as $j \in \{+, \text{sep}, -\}$. The terminal voltage output is governed by a combination of electric overpotential, electrode thermodyanmics, and Butler-Volmer kinetics, yielding:

$$\begin{aligned} V(t) =& \frac{RT}{\alpha^+ F}\sinh^{-1}\left(\frac{-I(t)}{2a^+ AL^+ i_0^+(c_{ss}^+(t))}\right) \\ &+ \frac{RT}{\alpha^- F}\sinh^{-1}\left(\frac{I(t)}{2a^- AL^- i_0^-(c_{ss}^-(t))}\right) \\ &+ U^+(c_{ss}^+(t)) - U^-(c_{ss}^-(t)) + R_f I(t), \end{aligned} \quad (5)$$

where the exchange current density i_0^j and solid-electrolyte surface concentration c_{ss}^j are computed as:

$$i_0^j(c_{ss}^j) = k^j\sqrt{c_e^0 c_{ss}^j(t)(c_{s,\max}^j - c_{ss}^j(t))}, \quad (6)$$

$$c_{ss}^j(t) = c_s^j(R_s^j,t), \quad j \in \{+,-\}. \quad (7)$$

The drawback of using SPM is that predicted voltage accuracy decreases as C-rate increases [2]. This presents a significant challenge for BMS applications, such as electrified vehicles, where high C-rates are commonly experienced. Large voltage output mismatch could lead to misguided control policies.

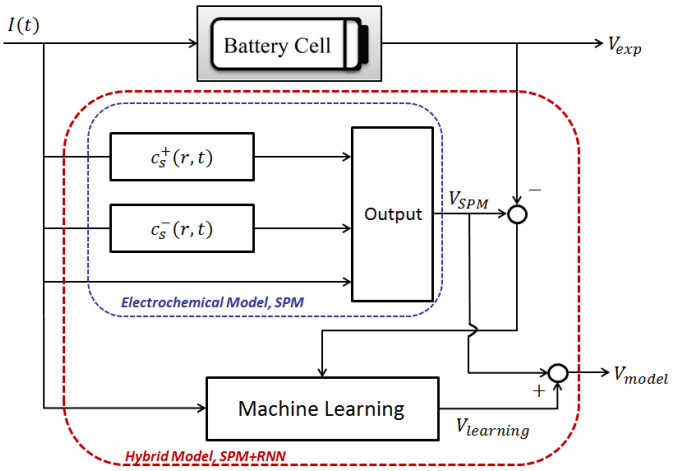

Figure 2: Proposed framework for hybrid modeling approach based on electrochemical model.

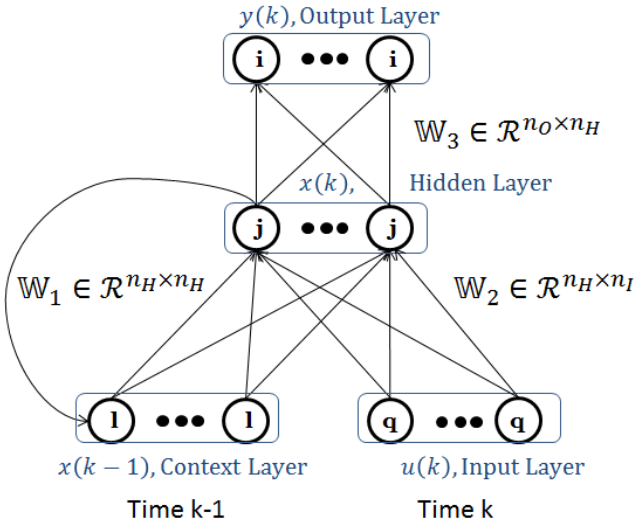

Figure 3: Elman network structure.

III. RNN Structure and Learning Algorithm

In this section we describe the hybrid modeling framework, which applies a machine learning algorithm on top of the electrochemical model as depicted in Fig. 2. We augment the first principals model with an input-output learning model to enhance voltage prediction accuracy.

A. Elman Network Structure and Training Algorithm

Consider the Elman network structure in Fig. 3. In general, there are n_I inputs, n_H hidden units, and n_O outputs in this network. The structure consists of output layer weight matrix $\mathbb{W}_3 \in \mathbf{R}^{n_O \times n_H}$, input layer-to-hidden layer weight matrix $\mathbb{W}_2 \in \mathbf{R}^{n_H \times n_I}$, and hidden layer-to-context layer weight matrix $\mathbb{W}_1 \in \mathbf{R}^{n_H \times n_H}$. Since we have few measurements in battery systems, this Elman network structure can be reduced to a single-input / single-output network, i.e., input current and output voltage. The network dynamic equations can be written as follows:

$$x(k) = f(\mathbb{W}_1 \cdot x(k-1) + \mathbb{W}_2 \cdot u(k)), \tag{8}$$
$$y(k) = g(\mathbb{W}_3 \cdot x(k)). \tag{9}$$

where $x(k)$ is called the "hidden layer" output vector, $y(k)$ is the output from the "output layer", and $u(k)$ represents the input. Functions $f(\cdot)$ and $g(\cdot)$ are "activation functions" for each layer. We take

$$f(z) = \tanh(z), \tag{10}$$

for reasons that emerge in the stability analysis. The function $g(\cdot)$ is often taken as the identity function, that is

$$y(k) = \mathbb{W}_3 \cdot x(k). \tag{11}$$

Note that (8)-(9) is a nonlinear system, with order given by user-selected parameter n_H.

The weight matrices, $\mathbb{W}_1$, $\mathbb{W}_2$, and $\mathbb{W}_3$ are updated to train the network. There are several methods for training RNNs, such as backpropagation through time, real-time recurrent learning (RTRL), and the extended Kalman filter [16]. In this work, we use the RTRL scheme which is a gradient-descent online learning algorithm that calculates the error gradient at every time step. The instantaneous squared error $E(k)$ at time step k is computed as:

$$E(k) = \frac{1}{2}e(k)^2 = \frac{1}{2}(y_d(k) - y(k))^2, \tag{12}$$

where $e(k)$ is the prediction error between desired output $y_d(k)$ and network output $y(k)$ at time step k. In RTRL, we update the weight matrix $\mathbb{W}$ via gradient-descent to minimize the squared error $E(k)$ such as:

$$W_{3,ij}(k+1) = W_{3,ij}(k) - \eta_3 \cdot \frac{\partial E(k)}{\partial W_{3,ij}(k)} \tag{13}$$
$$= W_{3,ij}(k) + \eta_3 \cdot e(k) \cdot x_j(k),$$
$$W_{2,jq}(k+1) = W_{2,jq}(k) - \eta_2 \cdot \frac{\partial E(k)}{\partial W_{2,jq}(k)} \tag{14}$$
$$= W_{2,jq}(k) + \eta_2 \cdot e(k) \cdot W_{3,ij} \cdot \gamma_j(k) \cdot u(k),$$
$$W_{1,jl}(k+1) = W_{1,jl}(k) - \eta_1 \cdot \frac{\partial E(k)}{\partial W_{1,jl}(k)} \tag{15}$$
$$= W_{1,jl}(k) \tag{16}$$
$$+ \eta_1 \cdot e(k) \cdot W_{3,ij} \cdot \gamma_j(k) \cdot x_j(k-1),$$

where $\eta_1, \eta_2, \eta_3 \in \mathbb{R}$ are user-selected learning rates for each weight, and i, j, q, l represent the unit index in each layer. Parameter γ_j is the derivative of activation function:

$$\gamma_j(k) = f_j^{'}(W_{1,jl} \cdot x(k) + W_{2,jq} \cdot u(k)).$$

Compared to typical feedforward neural networks that build approximators of nonlinear functions by using static input-output mappings [17], the Elman network is able to approximate arbitrary dynamical systems with arbitrary precision according to the *universal approximation property* [18]. We consider online training of (8)-(9), i.e. recursive estimation of $\mathbb{W}_1, \mathbb{W}_2, \mathbb{W}_3$, to capture the error between measured voltage and SPM-predicted voltage.

B. Special Case of Elman Network

A long-standing and un-solved challenge is proving the convergence of weights $\mathbb{W}_1, \mathbb{W}_2, \mathbb{W}_3$, or even convergence of $e(k)$ to zero via online learning [19]. The key issue is

nonlinear term $\mathbb{W}_3 x(k)$. In the neural network community, a number of papers have discussed this issue. For instance, in [20], the stability of the dynamic backpropagation algorithm is thoroughly discussed based on a dynamic learning process and RNN suggested by Narendra [10]. The authors propose two learning schemes: a multiplier method and a constrained learning rate algorithm to avoid unstable phenomenon during the learning process. The author in [19] proves that the Elman network's weights converge in the sense of Lyapunov functions under a backpropagation algorithm with adaptive dead zones.In this paper, we examine a special case of Elman networks for which weight convergence can be proven with RTRL update laws.

IV. Stability Analysis

This section analyzes stability for the proposed learning structure. First we derive the weight error dynamics,

$$\tilde{\mathbb{W}}(k) = \mathbb{W}(k) - \mathbb{W}^*, \tag{17}$$

where $\mathbb{W}^*$ is the "true" weight matrix that represents the desired output given input data. The output error at time k is given by:

$$\begin{aligned} e(k) &= y_d(k) - y(k) \\ &= \overline{\mathbb{W}}_3 f(\mathbb{W}_1^* x(k-1) + \mathbb{W}_2^* u(k)) \\ &\quad - \overline{\mathbb{W}}_3 f(\mathbb{W}_1(k) x(k-1) + \mathbb{W}_2(k) u(k)) \\ &= -\alpha \overline{\mathbb{W}}_3 \Big(\tilde{\mathbb{W}}_1(k) x(k-1) + \tilde{\mathbb{W}}_2(k) u(k) \Big), \end{aligned} \tag{18}$$

where the last equality is derived by the Mean Value Theorem, which requires function f to be continuous on a closed interval and differentiable on an open interval, which activation function $f(z) = \tanh(z)$ satisfies. Therefore $\exists\ z \in \mathbb{R}$ such that $\alpha = f'(z)$ and $0 < \alpha \leq 1$. Then, we can re-write (14)-(15) as follows:

$$\begin{aligned} \mathbb{W}_1(k+1) &= \mathbb{W}_1(k) + \eta_1(k) e(k) \gamma(k) x(k-1)^T \\ &= \mathbb{W}_1(k) - \eta_1(k) \alpha \overline{\mathbb{W}}_3 \Big(\tilde{\mathbb{W}}_1(k) \cdot x(k-1) \\ &\quad + \tilde{\mathbb{W}}_2(k) \cdot u(k) \Big) \gamma(k) x(k-1)^T, \end{aligned} \tag{19}$$

$$\begin{aligned} \mathbb{W}_2(k+1) &= \mathbb{W}_2(k) + \eta_2(k) e(k) \gamma(k) u(k) \\ &= \mathbb{W}_2(k) - \eta_2(k) \alpha \overline{\mathbb{W}}_3 \Big(\tilde{\mathbb{W}}_1(k) \cdot x(k-1) \\ &\quad + \tilde{\mathbb{W}}_2(k) \cdot u(k) \Big) \gamma(k) u(k). \end{aligned} \tag{20}$$

Assume the true weight matrices $\mathbb{W}_1^*, \mathbb{W}_2^*$ are constant, and $\eta_1(k) = \eta_2(k) = \eta(k)$. Then the weight error dynamics are:

$$\begin{aligned} \tilde{\mathbb{W}}_1(k+1) &= \tilde{\mathbb{W}}_1(k) - \eta(k) \alpha \overline{\mathbb{W}}_3 \Big(\tilde{\mathbb{W}}_1(k) x(k-1) \\ &\quad + \tilde{\mathbb{W}}_2(k) u(k)) \Big) \gamma(k) x(k-1)^T, \end{aligned} \tag{21}$$

$$\begin{aligned} \tilde{\mathbb{W}}_2(k+1) &= \tilde{\mathbb{W}}_2(k) - \eta(k) \alpha \overline{\mathbb{W}}_3 \Big(\tilde{\mathbb{W}}_1(k) x(k-1) \\ &\quad + \tilde{\mathbb{W}}_2(k) u(k) \Big) \gamma(k) u(k). \end{aligned} \tag{22}$$

The following theorem guarantees the convergence of $\mathbb{W}_1$, $\mathbb{W}_2$ to their true values under update laws (19)-(20).

Theorem 1: The zero equilibrium of weight error dynamics (21), (22) are stable in the sense of Lyapunov functional, $V(k) = ||\tilde{\mathbb{W}}_1(k)||^2 + ||\tilde{\mathbb{W}}_2(k)||^2$ if

$$\eta(k) \leq 2 \left[\alpha \| x(k-1)^T + u(k) \|^2 \overline{\mathbb{W}}_3{}^T \gamma(k)^T \right]^{-1}. \tag{23}$$

Proof: Lyapunov's indirect theorem requires that the difference in the Lyapunov functional denoted by ΔV is negative-definite along the trajectories $\forall\ \tilde{\mathbb{W}}_1, \tilde{\mathbb{W}}_2$, i.e.:

$$||\tilde{\mathbb{W}}_1(k+1)||^2 - ||\tilde{\mathbb{W}}_1(k)||^2 + ||\tilde{\mathbb{W}}_2(k+1)||^2 - ||\tilde{\mathbb{W}}_2(k)||^2 \leq 0.$$

Squaring both sides equations (19) and (20), we can construct ΔV as:

$$\begin{aligned} \Delta V &= -2\eta(k) \alpha \overline{\mathbb{W}}_3 \Big(||\tilde{\mathbb{W}}_1||^2 x(k-1) + \tilde{\mathbb{W}}_1^T \tilde{\mathbb{W}}_2 u(k) \Big) \\ &\quad \times \gamma(k) x(k-1)^T \\ &\quad + \Big|\Big| \eta(k) \alpha \overline{\mathbb{W}}_3 \Big(\tilde{\mathbb{W}}_1 x(k-1) + \tilde{\mathbb{W}}_2 u(k) \Big) \gamma(k) x(k-1)^T \Big|\Big|^2 \\ &\quad - 2\eta(k) \alpha \overline{\mathbb{W}}_3 \Big(\tilde{\mathbb{W}}_2^T \tilde{\mathbb{W}}_1 x(k-1) + ||\tilde{\mathbb{W}}_2|| u(k) \Big) \gamma(k) u(k) \\ &\quad + \Big|\Big| \eta(k) \alpha \overline{\mathbb{W}}_3 \Big(\tilde{\mathbb{W}}_1 x(k-1) + \tilde{\mathbb{W}}_2 u(k) \Big) \gamma(k) u(k) \Big|\Big|^2. \end{aligned} \tag{24}$$

After a few steps for re-arranging ΔV, we can finally obtain

$$\begin{aligned} \Delta V &\leq -2\eta(k) \alpha \overline{\mathbb{W}}_3 \Big|\Big| (\tilde{\mathbb{W}}_1 x(k-1) + \tilde{\mathbb{W}}_2 u(k) \Big|\Big|^2 \gamma(k) \\ &\quad + \eta(k)^2 \alpha^2 \Big|\Big| (\tilde{\mathbb{W}}_1 x(k-1) + \tilde{\mathbb{W}}_2 u(k)) \Big|\Big|^2 \\ &\quad \times \Big|\Big| x(k-1)^T + u(k) \Big|\Big|^2 ||\overline{\mathbb{W}}_3||^2 ||\gamma(k)||^2 \\ &\quad \leq 0, \end{aligned} \tag{25}$$

where the last inequality is derived from the definition of the adaptive learning rate in (23). ■

Remark 1: The adaptive learning rate, $\eta(k)$ defined in (23) can be re-written in terms of the previous desired output $y_d(k-1)$, since we adopt specialized Elman network structure expressed by $y_d(k-1) = \overline{\mathbb{W}}_3 \cdot x(k-1)$. That is:

$$\eta(k) \leq 2 \left[\alpha \Big\| \frac{1}{n_H} \mathbb{1}_{n_H} y_d(k-1) + u(k) \Big\|^2 \overline{\mathbb{W}}_3{}^T \gamma(k)^T \right]^{-1}, \tag{26}$$

where $\mathbb{1}_{n_H}$ is a $n_H \times 1$ vector of ones. Assuming a constant weight matrix $\overline{\mathbb{W}}_3$ plays the critical role for calculating the adaptive learning rates from the previous output data $y_d(k-1)$ and current input data $u(k)$.

Remark 2: Since the activation function is strictly monotonically increasing and has bounded derivative, we can state that when $\alpha = 1$, the following condition

$$\eta(k) \leq 2 \left[\Big\| \frac{1}{n_H} \mathbb{1}_{n_H} y_d(k-1) + u(k) \Big\|^2 \overline{\mathbb{W}}_3{}^T \gamma(k)^T \right]^{-1}, \tag{27}$$

becomes sufficient condition for (26) as $\alpha \in (0, 1]$. Practically, we calculate the learning rates based on (27) with equality as long as the computed value does not exceed 1, since $0 < \eta(k) \leq 1$ is used to regularize the learning rate.

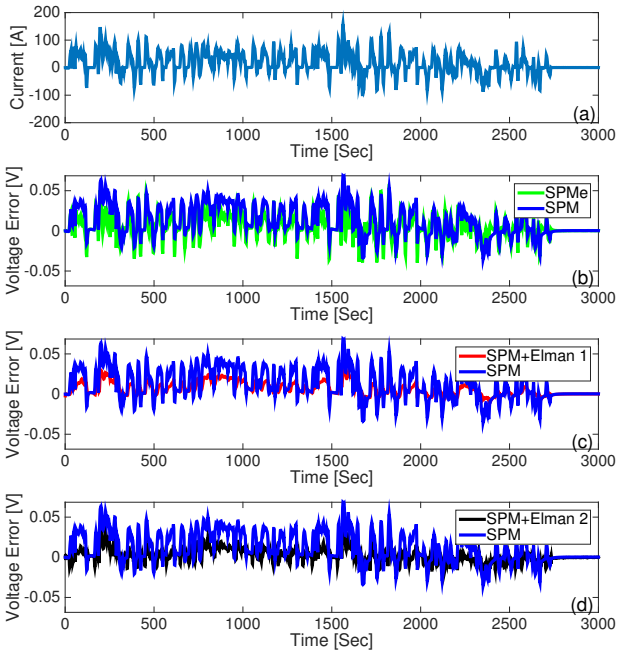

Figure 4: Voltage error comparison for UDDSx2 cycle: (a) Input current. (b) SPMe. (c) The special case of Elman network (d) The original Elman network.

V. Simulation Results

This section presents simulations that demonstrate the hybrid approach performance compared to several electrochemical models. The measured voltage data is generated from the DFN model, which plays the role of "true" voltage. Consequently, the model voltage error arises from model reduction. We notate Elman 1 as the special case of the Elman network and Elman 2 as the original Elman network. We use 4 hidden units for the Elman networks. Elman 1 uses adaptive learning parameter $\eta(k)$ given by (27), and Elman 2 uses fixed learning parameter $\eta = 0.3$ in the simulation. We train the RNN to identify an uncertainty model with various input profiles such as 1C/2C/5C discharge cycles and dynamic current profiles. This enables us to investigate how the identified RNN captures the uncertainty dynamics. In all cases, we require 10 epochs or less for the output error to converge. To evaluate the resulting learned RNN, our primary performance metric is the root mean square error (RMSE) between the DFN voltage and various model voltages.

A. Training Phase

The numerical training results are displayed in Table II, for various input profiles and models that do and do not incorporate learning structures. Indeed, our proposed hybrid approach resolves the limitations of the SPM. To explore this, we also consider the SPMe from [2] as a baseline comparison to quantify the error between pure electrochemical models and hybrid electrochemical models. As shown in Table II, the SPMe – which includes electrolyte dynamics – has about 50 % lower error than SPM in all scenarios. The hybrid modeling approach exhibits even better performance than SPMe in many cases. Interestingly, the SPM+Elman 2 consistently outperforms its simpler SPM+Elman 1 counterpart. This is not surprising, as SPM+Elman 2 has more parameters in $\mathbb{W}_3$ that it may tune to reduce output error.

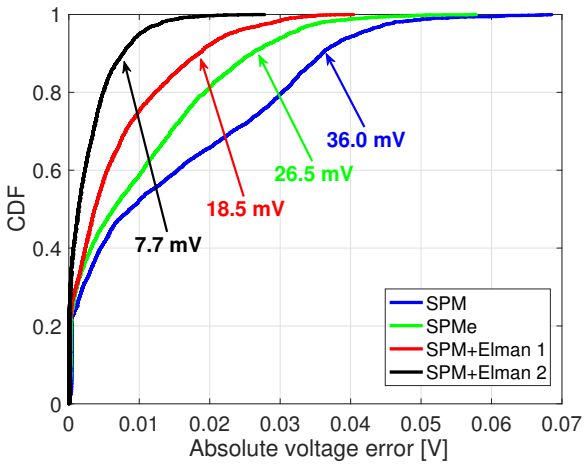

Figure 5: Cumulative distribution function of absolute voltage error on UDDS profile, with 90% quantile errors annotated.

However, as discussed before, analytic weight error convergence guarantees are not available.

Input Profiles (Training Data)	**SPM (60)** (No learning)	**SPMe (90)** (No learning)	**SPM+Elman 1 (64)** (Learning)	**SPM+Elman 2 (64)** (Learning)
1C discharge	20.0mV	10.0mV	1.8mV	1.8mV
2C discharge	39.4mV	19.0mV	3.0mV	5.2mV
5C discharge	107.2mV	54.6mV	67.6mV	38.7mV
Sine wave†	67.6mV	31.0mV	34.7mV	29.3mV
UDDS	20.8mV	14.7mV	10.1mV	7.6mV

†min:1C, max:5C

Table II: RMSE comparison for different models (w/ and w/o learning) and input profiles in the training phase. The number in parentheses is the number of states in the model.

Consider the Urban Dynamometer Driving Schedule (UDDS), which is a standard fuel economy test cycle. The input current data in Fig. 4(a) is generated from two concatenated UDDS cycles. Fig. 4(b) shows the voltage error for the SPMe and SPM relative to DFN, while Fig. 4(c),(d) presents the voltage errors for the proposed hybrid modeling scheme and SPM only, relative to the DFN. Note that we use the fixed weights after learning to verify the training performance. In addition, the cumulative distribution function (CDF) of the absolute voltage error in UDDS is computed for measuring model output precision, as shown in Fig. 5.We comment that applying adaptive learning rate $\eta(k)$ from (27) to the Elman 2 model yields divergent weights for UDDS. This corroborates our analysis that suggests a condition on $\eta(k)$ is not available to guarantee convergent weights for Elman 2, however such a condition is derived for Elman 1. This is a fundamental tradeoff balanced by the increased error reduction potential of the Elman 2 model.

B. Testing Phase

In this phase, we use the fixed weights obtained from the training phase to test the accuracy of hybrid electrochemical model. Here, we simply illustrate the robustness of the hybrid modeling framework. Specifically, we select the RNN models trained on UDDS data and test the performance with different input profiles. The UDDS data is selected for training, since it contains the richest frequency content. The numerical results in Table III illustrate two important observations: (i) The hybrid models do not degrade the voltage prediction accuracy compared to a stand-alone SPM

model. (ii) The hybrid models yield comparable accuracy to the SPMe.

Input Profiles (Testing Data)	SPM (60) (No learning)	SPMe (90) (No learning)	SPM+Elman 1 (64) (Learning)	SPM+Elman 2 (64) (Learning)
1C discharge	20.0mV	10.0mV	10.5mV	9.2mV
2C discharge	39.4mV	19.0mV	20.9mV	18.5mV
5C discharge	107.2mV	54.6mV	73.8mV	66.5mV
Sine wave†	67.6mV	31.0mV	42.7mV	37.9mV

†min:1C, max:5C

Table III: RMSE comparison for different models (w/ and w/o learning) and input profiles in the testing phase, using RNNs trained on UDDS. The number in parentheses is the number of states in the model.

VI. Conclusion

This paper proposes a hybrid electrochemical modeling framework that integrates recurrent neural networks (RNNs) with a reduced electrochemical model. Contrary to conventional approaches of integrating more physics into the electrochemical model, we examine the potential of data-driven models to recover model-measurement mismatch. Specifically, we study a simple RNN, called the Elman network, which mimics dynamical phenomena from observed data. Additionally, we prove convergence of estimated weights to their true values for a special case of the Elman network. This stability analysis yields an adaptive learning rate for the real-time recurrent learning (RTRL) update law. Simulation results demonstrate that our hybrid battery modeling significantly improves the predicted voltage accuracy compared to other reduced electrochemical models. On-going work involves validating our proposed hybrid approach experimentally with different types of Li-ion batteries.

References

[1] N. A. Chaturvedi, R. Klein, J. Christensen, J. Ahmed, and A. Kojic, "Algorithms for advanced battery-management systems", *IEEE Control Systems*, vol. 30, no. 3, pp. 49–68, Jun. 2010.

[2] S. J. Moura, F. B. Argomedo, R. Klein, A. Mirtabatabaei, and M. Krstic, "Battery state estimation for a single particle model with electrolyte dynamics", *IEEE Transactions on Control Systems Technology*, vol. PP, no. 99, pp. 1–16, 2016.

[3] S. Santhanagopalan, Q. Guo, P. Ramadass, and R. E. White, "Review of models for predicting the cycling performance of lithium ion batteries", *Journal of Power Sources*, vol. 156, no. 2, pp. 620–628, 2006.

[4] S. Santhanagopalan and R. E. White, "Online estimation of the state of charge of a lithium ion cell", *Journal of power sources*, vol. 161, no. 2, pp. 1346–1355, 2006.

[5] D. Di Domenico, A. Stefanopoulou, and G. Fiengo, "Lithium-ion battery state of charge and critical surface charge estimation using an electrochemical model-based extended kalman filter", *Journal of dynamic systems, measurement, and control*, vol. 132, no. 6, p. 061 302, 2010.

[6] S. Dey, B. Ayalew, and P. Pisu, "Nonlinear robust observers for state-of-charge estimation of lithium-ion cells based on a reduced electrochemical model", *IEEE Transactions on Control Systems Technology*, vol. 23, no. 5, pp. 1935–1942, 2015.

[7] S. K. Rahimian, S. Rayman, and R. E. White, "Extension of physics-based single particle model for higher charge-discharge rates", *Journal of Power Sources*, vol. 224, no. 0, pp. 180–194, 2013.

[8] T. R. Tanim, C. D. Rahn, and C.-Y. Wang, "A temperature dependent, single particle, lithium ion cell model including electrolyte diffusion", *Journal of Dynamic Systems, Measurement, and Control*, vol. 137, no. 1, p. 011 005, 2015.

[9] T. R. Tanimm, C. D. Rahn, and C.-Y. Wang, "State of charge estimation of a lithium ion cell based on a temperature dependent and electrolyte enhanced single particle model", *Energy*, vol. 80, pp. 731–739, 2015.

[10] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks", *IEEE Transactions on neural networks*, vol. 1, no. 1, pp. 4–27, 1990.

[11] J. Liu, A. Saxena, K. Goebel, B. Saha, and W. Wang, "An adaptive recurrent neural network for remaining useful life prediction of lithium-ion batteries", DTIC Document, Tech. Rep., 2010.

[12] G. Capizzi, F. Bonanno, and G. M. Tina, "Recurrent neural network-based modeling and simulation of lead-acid batteries charge–discharge", *IEEE Transactions on Energy Conversion*, vol. 26, no. 2, pp. 435–443, 2011.

[13] A. Eddahech, O. Briat, E. Woirgard, and J.-M. Vinassa, "Remaining useful life prediction of lithium batteries in calendar ageing for automotive applications", *Microelectronics Reliability*, vol. 52, no. 9, pp. 2438–2442, 2012.

[14] S. J. Moura, M. Krstic, and N. A. Chaturvedi, "Adaptive pde observer for battery soc/soh estimation", in *ASME 2012 5th Annual Dynamic Systems and Control Conference joint with the JSME 2012 11th Motion and Vibration Conference*, American Society of Mechanical Engineers, 2012, pp. 101–110.

[15] J. L. Elman, "Finding structure in time", *Cognitive Science*, vol. 14, no. 2, pp. 179–211, 1990.

[16] H. Jaeger, *Tutorial on training recurrent neural networks, covering bppt, rtrl, ekf and the "echo state network" approach*. GMD-Forschungszentrum Informationstechnik, 2002.

[17] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators", *Neural networks*, vol. 2, no. 5, pp. 359–366, 1989.

[18] T. Chen and H. Chen, "Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems", *IEEE Transactions on Neural Networks*, vol. 6, no. 4, pp. 911–917, 1995.

[19] Q. Song, "On the weight convergence of elman networks", *IEEE Transactions on Neural networks*, vol. 21, no. 3, pp. 463–480, 2010.

[20] L. Jin and M. M. Gupta, "Stable dynamic backpropagation learning in recurrent neural networks", *IEEE Transactions on Neural Networks*, vol. 10, no. 6, pp. 1321–1334, 1999.