On trade-offs between computational complexity and accuracy of electrochemistry-based battery models

2019 IEEE 58th Conference on Decision and Control (CDC)
Palais des Congrès et des Expositions Nice Acropolis
Nice, France, December 11-13, 2019

On Trade-offs Between Computational Complexity and Accuracy of Electrochemistry-based Battery Models

Z. Khalik H.J. Bergveld M.C.F. Donkers

Abstract— **In this paper, we propose several simplifications to the so-called Doyle-Fuller-Newman (DFN) model, which is a popular electrochemistry-based battery model. This simplified DFN (SDFN) model allows for a computationally very efficient implementation. The simplifications are a result of several assumptions, which will be justified for two different parameter sets. Finally, the SDFN model proposed is compared to the DFN model as well as an implementation of the single-particle model, for the two parameter sets. This will show that by making specific assumptions, simplifications can be made that have no significant impact on the model accuracy, while the computation time can be drastically decreased. This leads to a simulation time of over 3600 times faster than real-time.**

I. Introduction

With the emergence of electric vehicles and widespread usage of mobile devices, batteries are taking an increasingly important role. To effectively manage and control batteries, battery models are needed, e.g., for state-of-charge/state-of-health estimation [1], active balancing [2], and fast charging [3]. Equivalent-circuit models (ECM) are often used for these purposes, where the battery is modeled using (passive) circuit elements, e.g., [4]. However, although ECMs are computationally fast, these types of models provide limited information on the internal states of the battery, such as concentrations and potentials. The information of these internal states is crucial for more precise control. Therefore, electrochemistry-based models have generated increasing interest in control applications. A widely-used electrochemistry-based model is the so-called Doyle-Fuller-Newman (DFN) model [5], which is described by a set of partial differential equations (PDEs). While the DFN model can describe internal states, due to its complexity, computation times to simulate the model have made it, so far, unsuitable for control-oriented applications.

There are several ways to reduce the computational burden of the DFN model. One way to reduce complexity is to apply model reduction to the DFN model, e.g. [6], [7]. However, the reduction is mostly in the number of equations, while the reduction in computation time is only marginal, when applied to the DFN model [6]. Another way to reduce complexity is by simplifying the model equations. A popular example is the so-called single-particle model (SPM) [8], and its more advanced versions such as the SPM with electrolyte dynamics (SPMe) [9], where the equations for solid-phase concentration and potentials are simplified. However, generally such simplifications lead to a significant loss in accuracy, and, usually, the simplified models are validated for a single parameter set. While a simplified model could show good accordance with the DFN model for one parameter set, it may not be the case for another parameter set.

Besides addressing the complexity of the DFN model, another way to reduce computation time is to develop algorithms that compute the solution to the DFN model more efficiently, see e.g. [6], [10]. In these papers, the PDEs that describe the DFN model are spatially and temporally discretized, which results in a set of nonlinear algebraic equations (AEs). While there are papers where existing toolboxes such as CasADi are used in order to solve this set of equations [11], the use of such toolboxes can prohibit the understanding of the model structure, which can be used to achieve a faster implementation.

In this paper, we propose to simplify the equations of the DFN model, to arrive at a simplified DFN (SDFN) model. This SDFN model aims to allow for a more computationally efficient implementation, thereby reducing computation time. After spatial and temporal discretization of the SDFN model, we will show that through substitution of equations, the number of equations describing the SDFN model can be considerably reduced to a small set of algebraic equations. We will study the impact of these assumptions on model accuracy for two different sets of parameters, to show the validity of these assumptions. Furthermore, we compare the model accuracy and computation time of the DFN model, the proposed SDFN model, and the SPMe for two different parameter sets. We will show that based on the proposed assumptions, the simplifications made allow for a significant decrease in simulation time, resulting in a simulation speed of over 3600 times faster than real-time.

II. Battery Modeling

In this section, we briefly formulate the Doyle-Fuller-Newman (DFN) model, and introduce several simplifications that, as we will show in Section IV, have no significant impact on the accuracy of the model, both in the input-output behavior as well as the internal states.

This work has received financial support from the Horizon 2020 programme of the European Union under the grants 'Electric Vehicle Enhanced Range, Lifetime And Safety Through INGenious battery management' (EVERLASTING-713771) and 'Advancing fail-aware, fail-safe, and fail-operational electronic components, systems, and architectures for fully automated driving to make future mobility safer, affordable, and end-user acceptable' (AutoDrive-737469).

The authors are with the Depart. of Electrical Eng. of Eindhoven University of Technology, Netherlands. Henk Jan Bergveld is also affiliated with NXP Semiconductors, Eindhoven, Netherlands. E-mail: {z.khalik, h.j.bergveld,m.c.f.donkers}@tue.nl

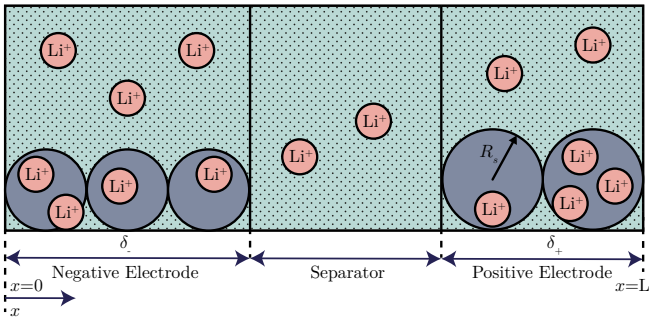

Fig. 1: DFN modeling approach for a Li-ion cell.

A. Doyle-Fuller-Newman Model

The DFN model is a widely used electrochemistry-based model introduced in [5]. Fig. 1 illustrates the modeling approach for a Li-ion cell. In the x dimension, the cell is divided into three regions, namely the negative electrode, the separator, and the positive electrode. In the electrodes, Li-ions exist essentially in two phases. In the solid phase, Li-ions are intercalated into the solid-phase material, which is represented by spheres with radius R_s. In the electrolyte phase, Li-ions exist in a dissolved state in the electrolyte. In the separator, Li-ions exist only in the electrolyte phase. During charging, intercalated Li-ions exit the solid particles in the positive electrode and enter the solid particles in the negative electrode. During discharging, the opposite process happens. We will shortly summarize the governing equations of the DFN model, which will be mostly based on the formulation given in [12]. For compactness of notation, the time and space dependency of the variables given will be left out of the equations. The DFN model is governed by four coupled partial differential equations (PDEs):

1) The Li-ion concentration in the solid phase $c_s(x,r,t)$ for $x \in [0,\delta_-] \cup [L-\delta_+, L]$ is given by Fick's law as

$$\frac{\partial c_s}{\partial t} = \frac{D_s}{r^2}\frac{\partial}{\partial r}\left(r^2 \frac{\partial c_s}{\partial r}\right), \tag{1a}$$

with boundary conditions

$$\frac{\partial c_s}{\partial r}\Big|_{r=0} = 0, \quad -D_s \frac{\partial c_s}{\partial r}\Big|_{r=R_s} = j_n, \tag{1b}$$

with D_s the diffusion coefficient of lithium in the solid phase, and $j_n(x,t)$ is the net molar flux of Li-ions exiting the particle. Furthermore, δ_- and δ_+ are the thickness of the negative and positive electrode, respectively, and L is the cell thickness (see Fig. 1).

2) The Li-ion concentration in the electrolyte phase $c_e(x,t)$ for $x \in [0,L]$ is given by

$$\varepsilon_e \frac{\partial c_e}{\partial t} = \frac{\partial}{\partial x}\left(D_e \frac{\partial c_e}{\partial x}\right) + a_s(1-t_+^0) j_n, \tag{2a}$$

with boundary conditions

$$D_e^{\text{eff}} \frac{\partial c_e}{\partial x}\Big|_{x=0} = D_e^{\text{eff}} \frac{\partial c_e}{\partial x}\Big|_{x=L} = 0, \tag{2b}$$

where $D_e^{\text{eff}} = D_e \varepsilon_e^p$ is the effective Li-ion diffusion coefficient in the electrolyte phase, in which ε_e is the electrolyte phase volume fraction, p is the Bruggeman porosity exponent, and D_e is the diffusion constant of Li-ions in the electrolyte. Furthermore, in (2), $a_s = 3\varepsilon_s/R_s$ is the specific interfacial surface area, in which ε_s is the active material volume fraction, and t_+^0 is the transference number of Li ions.

3) The potential in the solid phase $\phi_s(x,t)$ for $x \in [0,\delta_-] \cup [L-\delta_+, L]$ is given by Ohm's law, i.e.,

$$\frac{\partial}{\partial x}\left(\sigma^{\text{eff}} \frac{\partial \phi_s}{\partial x}\right) = a_s F j_n, \tag{3a}$$

with boundary conditions

$$\sigma^{\text{eff}} \frac{\partial \phi_s}{\partial x}\Big|_{x=0} = \frac{i_{\text{app}}}{A_{\text{surf}}}, \quad \sigma^{\text{eff}} \frac{\partial \phi_s}{\partial x}\Big|_{x=\delta_-} = 0 \tag{3b}$$

$$\sigma^{\text{eff}} \frac{\partial \phi_s}{\partial x}\Big|_{x=L-\delta_+} = 0, \quad \sigma^{\text{eff}} \frac{\partial \phi_s}{\partial x}\Big|_{x=L} = \frac{i_{\text{app}}}{A_{\text{surf}}}, \tag{3c}$$

where F is Faraday's constant, $\sigma^{\text{eff}} = \varepsilon_s \sigma$ is the effective electronic conductivity of a porous electrode, in which σ is the conductivity of the solid material, A_{surf} is the area of the electrode plate, and $i_{\text{app}}(t)$ is the applied current through the battery, with $i_{\text{app}} > 0$ indicating charging.

4) The potential in the electrolyte phase $\phi_e(x,t)$ for $x \in [0,L]$ is given by

$$\frac{\partial}{\partial x}\left(\kappa_{\text{eff}}(c_e)\frac{\partial \phi_e}{\partial x} + \kappa_{\text{eff}}(c_e)\frac{2RT}{F}(t_+^0 - 1)\frac{\partial \ln c_e}{\partial x}\right) = -a_s F j_n, \tag{4a}$$

with boundary conditions

$$\kappa_{\text{eff}}(c_e)\frac{\partial \phi_e}{\partial x}\Big|_{x=0} = \phi_e\Big|_{x=L} = 0, \tag{4b}$$

in which $\kappa_{\text{eff}}(c_e) = \kappa(c_e)\varepsilon_e^p$ is the effective ionic conductivity, where $\kappa(c_e)$ is given by $\kappa(c_e) = 0.00158 c_e \exp(5.63 c_e^{1.4}/10^5)$ in [12], R is the universal gas constant, and T is the absolute temperature.

The above PDEs (1)-(4) are coupled by a Butler-Volmer rate equation, which describes the chemical reaction rate at the solid/electrolyte interface. This rate equation is given by

$$j_n = \frac{i_0}{F}\left(\exp\left(\frac{\alpha_a F}{RT}\eta\right) - \exp\left(-\frac{\alpha_c F}{RT}\eta\right)\right), \tag{5a}$$

which is only defined for $x \in [0,\delta_-] \cup [L-\delta_+, L]$ and assumed zero for $x \in (\delta_-, L-\delta_+)$. In (5a), α_a is the anodic transfer coefficient, α_c is the cathodic transfer coefficient, and the overpotential at the electrodes $\eta(x,t)$ is defined as

$$\eta = \phi_s - \phi_e - U(\bar{c}_s), \tag{5b}$$

in which $U(\bar{c}_s)$ denotes the equilibrium potential of the electrode, which can be given by a pre-defined function typically of the solid-phase concentration at the solid-electrolyte interface $\bar{c}_s(x,t) = c_s(R_s,x,t)$. Furthermore, the exchange current density i_0 in (5a) is given by

$$i_0 = k_0 c_e^{\alpha_a}(c_{s,\max} - \bar{c}_s)^{\alpha_a}(\bar{c}_s)^{\alpha_c}, \tag{5c}$$

where k_0 is the rate constant of the electrochemical reaction, and $c_{s,\max}$ is the maximum concentration in the solid-phase. Finally, the terminal battery voltage is computed with

$$V(t) = \phi_s(L,t) - \phi_s(0,t) + \frac{R_f}{A_{\text{surf}}} i_{\text{app}}(t), \tag{6}$$

in which R_f is an empirical contact resistance.

B. Simplified Doyle-Fuller-Newman Model

In order to reduce computational complexity, some simplifications can be made on the DFN model. We will show using simulation results in Section IV that the simplifications proposed below do not significantly sacrifice the accuracy for two different parameter sets. The assumptions made to arrive at this simplified model are as follows.

- [A1]: The rate equation (5a) can be linearized with respect to the overpotential η around the origin, due to the fact that $\frac{\alpha_a F}{RT} = \frac{\alpha_c F}{RT} \gg \eta$.
- [A2]: The effective ionic conductivity κ_{eff} is constant over x, i.e., $\tilde{\kappa}_{\text{eff}} = \kappa(\tilde{c}_e)\varepsilon_e^p$, where $\tilde{c}_e$ is the average Li-ion concentration in the electrolyte.
- [A3]: The time derivative of the electrolyte concentration $\frac{\partial c_e}{\partial t}$ is sufficiently small so that the term $\ln c_e$ in (4) can be linearized with respect to c_e around a linearization point c_e^*, i.e., $\ln c_e \approx \ln c_e^* + \frac{c_e - c_e^*}{c_e^*}$.

Using [A1], and that $\alpha_a + \alpha_c = 1$, the resulting linearized Butler-Volmer equation (of (5a)-(5b)) can be written as

$$j_n = \tfrac{i_0}{RT}(\phi_s - \phi_e - U). \tag{7}$$

Furthermore using [A2] and [A3], (4) can be simplified to

$$\tfrac{\partial}{\partial x}\big(\tilde{\kappa}_{\text{eff}}\tfrac{\partial \phi_e}{\partial x} + \tilde{\kappa}_{\text{eff}}\tfrac{2RT}{F}(t_+^0 - 1)\tfrac{\partial\left(\ln c_e^* + \frac{c_e - c_e^*}{c_e^*}\right)}{\partial x}\big) = -a_s F j_n, \tag{8}$$

with the boundary conditions given in (4b). Thus, the simplified DFN model is given by (1)-(3), (8), (4b), (7), (5c), and (6). Note that through the simplifications we have eliminated much of the non-linearity of the DFN model. The only non-linearities with respect to the state variables c_s, c_e, ϕ_s, ϕ_e remain in the equation for $i_0(c_s, c_e)$ as well as the equilibrium potential $U(\bar{c}_s)$.

III. Model Implementation

The objective of this paper is to compare the computational complexity and the accuracy of several electrochemistry-based models, including the newly proposed SDFN model. To have low computational complexity of the proposed SDFN model, we propose a computationally efficient implementation of this model. This implementation is warranted by Assumptions [A1]-[A3], and involves several steps. Firstly, fairly standard spatial and temporal discretization is applied to arrive at a set of nonlinear algebraic equations (AEs). Secondly, the set of AEs is reduced to a smaller set of AEs through substitution, after which the resulting set of AEs can be solved using Newton's method. The discretization procedure described here is similar to the procedure presented in [6]. Therefore, in this section, we will shortly summarize this procedure to formulate the set of AEs that arise from the discretization. For further details on the discretization approach, the reader is referred to [6].

A. Discretization

As a first step, spatial discretization is applied on the partial differential equations (PDEs) (1) - (3) and (8). The equation describing the diffusion of the solid-phase concentration (1) is discretized along the radial direction using a finite-difference-method (FDM), to arrive at a set of differential algebraic equations (DAEs). The other equations (2), (3), and (8) are discretized using a finite-volume-method (FVM), and the resulting set of nonlinear DAEs is given as

$$\tfrac{d}{dt}\mathbf{c}_s = A_{c_s}\mathbf{c}_s + B_{c_s}\mathbf{j}_n \tag{9a}$$
$$\tfrac{d}{dt}\mathbf{c}_e = A_{c_e}\mathbf{c}_e + B_{c_e}\mathbf{j}_n \tag{9b}$$
$$0 = A_{\phi_s}\boldsymbol{\phi}_s + B_{\phi_s}\mathbf{j}_n + C_{\phi_s} i_{\text{app}} \tag{9c}$$
$$0 = A_{\phi_e}\boldsymbol{\phi}_e + B_{\phi_e}\mathbf{j}_n + D_{\phi_e}\left(\text{diag}(\mathbf{c}_e^*)^{-1}\mathbf{c}_e + \ln(\mathbf{c}_e^*)\right), \tag{9d}$$

where the bold faced characters refer to their respective vector variables, which are defined as

$$\begin{aligned}
\mathbf{c}_s(t) &= \left[\, c_s(x_1, r_1, t) \;\ldots\; c_s(x_1, r_{n_{r,n}}, t) \;\ldots\; c_s(x_{n_n+n_p}, r_{n_{r,p}}, t) \,\right]^\top, \\
\mathbf{c}_e(t) &= \left[\, c_e(x_1, t) \;\ldots\; c_e(x_{n_n+n_s+n_p}, t) \,\right]^\top, \\
\boldsymbol{\phi}_s(t) &= \left[\, \phi_s(x_1, t) \;\ldots\; \phi_s(x_{n_n}, t) \;\ldots\; \phi_s(x_{n_n+n_p}, t) \,\right]^\top,
\end{aligned} \tag{10}$$

where x_i and r_i are the grid points of the discretization, and $\boldsymbol{\phi}_e$, $\mathbf{j}_n$ are defined similarly to $\mathbf{c}_e$ and $\boldsymbol{\phi}_s$, respectively. Furthermore, in (10), n_n, n_s, n_p are the number of elements of the FVM discretization, and $n_{r,n}, n_{r,p}$ are the number of elements of the FDM discretization. How to construct matrices $A_i, B_i, i \in \{c_s, c_e, \phi_s, \phi_e\}$, C_{ϕ_s}, and D_{ϕ_e} is explained in detail in [6]. Note that in contrast to the set of DAEs obtained after spatial discretization in [6], except for the coupling through $\mathbf{j}_n$, the four sets of equations (9) are linear in the state variables $\mathbf{c}_s, \mathbf{c}_e, \boldsymbol{\phi}_s, \boldsymbol{\phi}_e$. The four sets of DAEs (9) are coupled by the linearized (with respect to $\eta = \phi_s - \phi_e - U$) Butler-Volmer rate equation, written as

$$\mathbf{j}_n = \text{diag}\left(\tfrac{\mathbf{i}_0(\bar{\mathbf{c}}_s, \bar{\mathbf{c}}_e)}{RT}\right)(\boldsymbol{\phi}_s - \bar{\boldsymbol{\phi}}_e - \mathbf{U}(\bar{\mathbf{c}}_s)), \tag{11}$$

in which the $\text{diag}(v)$ denotes a diagonal matrix with the elements of vector v on the main diagonal. Furthermore, in (11), the barred variables $\bar{\mathbf{c}}_s$, $\bar{\mathbf{c}}_e$, and $\bar{\boldsymbol{\phi}}_e$ refer to selected version of their boldfaced counterparts, where $\bar{\mathbf{c}}_s$ denotes the vector of solid-phase surface concentrations, and $\bar{\mathbf{c}}_e$ and $\bar{\boldsymbol{\phi}}_e$ denote the parts of $\mathbf{c}_e$ and $\boldsymbol{\phi}_e$ given in the electrodes, respectively. Mathematically, $\bar{\mathbf{c}}_s, \bar{\mathbf{c}}_e, \bar{\boldsymbol{\phi}}_e$ and $\mathbf{c}_s, \mathbf{c}_e, \boldsymbol{\phi}_e$ are related, respectively, as follows

$$\bar{\mathbf{c}}_s = \bar{A}_{c_s}\mathbf{c}_s, \quad \bar{\mathbf{c}}_e = \bar{A}_{c_e}\mathbf{c}_e, \quad \bar{\boldsymbol{\phi}}_e = \bar{A}_{\phi_e}\boldsymbol{\phi}_e, \tag{12}$$

where

$$\begin{aligned}
\bar{A}_{c_s} &= \text{diag}\left(I_{n_n} \otimes [\mathbf{0}_{1\times n_{r,n}-1}, 1], I_{n_p} \otimes [\mathbf{0}_{1\times n_{r,p}-1}, 1]\right) \\
\bar{A}_{c_e} &= \bar{A}_{\phi_e} = \begin{bmatrix} I_{n_n} & 0 & 0 \\ 0 & 0 & I_{n_p} \end{bmatrix}.
\end{aligned}$$

The differential equations (9a) and (9b) can further be discretized in time with sampling time δ_t using a backward Euler scheme, to arrive at the following set of AEs

$$0 = \hat{A}_{c_s}\mathbf{c}_s(t_k) + \hat{B}_{c_s}\mathbf{j}_n(t_k) + \mathbf{c}_s(t_{k-1}) \tag{13a}$$
$$0 = \hat{A}_{c_e}\mathbf{c}_e(t_k) + \hat{B}_{c_e}\mathbf{j}_n(t_k) + \mathbf{c}_e(t_{k-1}) \tag{13b}$$
$$0 = A_{\phi_s}\boldsymbol{\phi}_s(t_k) + B_{\phi_s}\mathbf{j}_n(t_k) + C_{\phi_s} i_{\text{app}}(t_k) \tag{13c}$$
$$0 = A_{\phi_e}\boldsymbol{\phi}_e(t_k) + B_{\phi_e}\mathbf{j}_n(t_k) + D_{\phi_e}\left(\Xi_{\phi_e}^1\mathbf{c}_e(t_k) + \Xi_{\phi_e}^2\right) \tag{13d}$$

where $t_k \in \{1, \ldots, t_f\}$ is the sample time, in which t_f is the final sample time. Furthermore, $\hat{A}_{c_s} = (\delta_t A_{c_s} - \mathbf{I}_{n_r n_{np}})$, $\hat{B}_{c_s} = \delta_t B_{c_s}$, $\hat{A}_{c_e} = (\delta_t A_{c_e} - \mathbf{I}_{n_x})$, $\hat{B}_{c_e} = \delta_t B_{c_e}$, $\Xi^1_{\phi_e} = \text{diag}(\mathbf{c}_e(t_{k-1}))^{-1}$, and $\Xi^2_{\phi_e} = \ln(\mathbf{c}_e(t_{k-1}))$. Note that we have chosen the linearization point $\mathbf{c}^*_e$ as $\mathbf{c}_e(t_{k-1})$ in accordance with assumption [A3]. The approximation $\ln(\mathbf{c}_e) \approx \Xi^1_{\phi_e}\mathbf{c}_e + \Xi^2_{\phi_e}$ is then valid as long as the difference between $\mathbf{c}_e(t_k)$ and $\mathbf{c}_e(t_{k-1})$ is relatively small.

B. Solution Method

The set of nonlinear AEs (13) obtained after discretization can be solved using any root-finding algorithm, such as Newton's method. However, due to the relatively large number of state variables, the use of such algorithms can be computationally slow. A large part of the computational effort is in the computation of the inverse of the Jacobian of the AEs (13). Therefore, in [6] a method was proposed in which Newton's method was applied sequentially to each set of equations of (13), thereby reducing the computation of the inverse of a single large Jacobian to the computation of the inverses of four smaller Jacobians. However, in doing so, some information of the large Jacobian is lost, which means that a quadratic convergence rate can no longer be achieved. Still, the computation time of the large Jacobian is sufficiently large, that the method proposed in [6] is faster than solving (13) directly using Newton's method.

Rather than sequentially solving (13), we propose a solution method, which retains the full information of the Jacobian of (13), but reduces the number of AEs. This can be done by substitution of equations, such that one state variable remains. In doing so, the full information of the Jacobian of (13) is contained in a smaller Jacobian related to equation of the remaining variable. The derivation involves solving (11) for the chosen state variable, and solving their associated equation given in (13) for j_n. State variable ϕ_s is both linear in j_n and its associated equation in (13) can be explicitly solved for j_n. Hence, the set of AEs (13) will be reduced to one set of AEs such that ϕ_s can be implicitly solved using the Newton's method, from which c_s,c_e,ϕ_e can be obtained.

The derivation of the reduced set of AEs is as follows. First, we solve (11) for ϕ_s, which gives

$$\boldsymbol{\phi}_s(t_k) - \bar{\boldsymbol{\phi}}_e(t_k) - F_{\mathbf{j}_n}(\bar{\mathbf{c}}_s(t_k), \bar{\mathbf{c}}_e(t_k))\mathbf{j}_n(t_k) - U(\bar{\mathbf{c}}_s(t_k)) = 0, \tag{14}$$

where $F_{\mathbf{j}_n}(\bar{\mathbf{c}}_s, \bar{\mathbf{c}}_e) = \text{diag}\left(\frac{2\alpha}{RT}\mathbf{i}_0(\bar{\mathbf{c}}_s, \bar{\mathbf{c}}_e)\right)^{-1}$. Then, by solving (13c) for $\mathbf{j}_n$, $\mathbf{j}_n$ can be given by

$$\mathbf{j}_n(t_k) = -B^{-1}_{\phi_s}\left(A_{\phi_s}\boldsymbol{\phi}_s(t_k) + C_{\phi_s} i_{\text{app}}(t_k)\right) \tag{15}$$

These expressions allow the state variables $\bar{\mathbf{c}}_s, \bar{\mathbf{c}}_e$ to be expressed as a function of $\boldsymbol{\phi}_s$ by first substituting $\mathbf{j}_n$ in (15) into their respective associated equations given in (13), i.e.,

$$\hat{A}_{c_i}\mathbf{c}_i(t_k) - \hat{B}_{c_i}B^{-1}_{\phi_s}(A_{\phi_s}\boldsymbol{\phi}_s(t_k) + C_{\phi_s} i_{\text{app}}(t_k)) + \mathbf{c}_i(t_{k-1}) = 0,$$

for $i \in \{s, e\}$, then solving this for their respective state variables and pre-multiplying by A_{c_s}, A_{c_e}, resulting in

$$\bar{\mathbf{c}}_i(t_k) = \Gamma_{c_i} i_{\text{app}}(t_k) + \Phi_{c_i}\boldsymbol{\phi}_s(t_k) + \Theta_{c_i}, \tag{16a}$$

where $\Gamma_{c_i} = \bar{A}_{c_i}\hat{A}^{-1}_{c_i}\hat{B}_{c_i}B^{-1}_{\phi_s}C_{\phi_s}$, $\Phi_{c_i} = \bar{A}_{c_i}\hat{A}^{-1}_{c_i}\hat{B}_{c_i}B^{-1}_{\phi_s}A_{\phi_s}$, and $\Theta_{c_i} = \bar{A}_{c_i}\hat{A}^{-1}_{c_i}\mathbf{c}_i(t_{k-1})$, for $i \in \{s, e\}$. Note the presence of the full state vectors $\mathbf{c}_s$ and $\mathbf{c}_e$, which can be obtained by (16a), except without the pre-multiplication by $\bar{A}_{c_s}$ and $\bar{A}_{c_e}$, respectively. Similarly, $\bar{\boldsymbol{\phi}}_e$ can be expressed as a function of $\boldsymbol{\phi}_s$ and $\mathbf{c}_e$ by substituting (15) into (13d), i.e.,

$$A_{\phi_e}\boldsymbol{\phi}_e(t_k) - B_{\phi_e}B^{-1}_{\phi_s}\left(A_{\phi_s}\boldsymbol{\phi}_s(t_k) + C_{\phi_s} i_{\text{app}}(t_k)\right) + D_{\phi_e}\left(\Xi^1_{\phi_e}\mathbf{c}_e(t_k) + \Xi^2_{\phi_e}\right) = 0, \tag{16b}$$

then solving (13d) for $\boldsymbol{\phi}_e$ and pre-multiplying by $\bar{A}_{\phi_e}$, giving

$$\bar{\boldsymbol{\phi}}_e(t_k) = \Gamma_{\phi_e} i_{\text{app}}(t_k) + \Phi_{\phi_e}\boldsymbol{\phi}_s(t_k) + \Pi_{\phi_e}\mathbf{c}_e(t_k) + \Theta_{\phi_e} \tag{16c}$$

in which $\Gamma_{\phi_e} = \bar{A}_{\phi_e}A^{-1}_{\phi_e}B_{\phi_e}B^{-1}_{\phi_s}C_{\phi_s}$, $\Phi_{\phi_e} = \bar{A}_{\phi_e}A^{-1}_{\phi_e}B_{\phi_e}B^{-1}_{\phi_s}A_{\phi_s}$, $\Pi_{\phi_e} = -\bar{A}_{\phi_e}A^{-1}_{\phi_e}D_{\phi_e}\Xi^1_{\phi_e}$, and $\Theta_{\phi_e} = -\bar{A}_{\phi_e}A^{-1}_{\phi_e}D_{\phi_e}\Xi^2_{\phi_e}$.

The above steps allow the number of AEs given in (13) to be reduced, by substituting (15), (16a), and (16c) into (14), leading to an expression of the form $F(\boldsymbol{\phi}_s(t_k)) = 0$, which can be solved using Newton's method, i.e.,

$$\boldsymbol{\phi}^{m+1}_s(t_k) = \boldsymbol{\phi}^m_s(t_k) - \gamma J(\boldsymbol{\phi}^m_s(t_k))^{-1}F(\boldsymbol{\phi}^m_s(t_k)), \tag{17}$$

where $m \in \{1, \ldots, M\}$, in which M is the maximum number of iterations, represents the current iteration in Newton's method, J is the Jacobian of F, and γ is a damping coefficient which can be used to warrant convergence. Note that the Jacobian of F has $(n_n + n_p)$ rows and columns, which is considerably smaller than the Jacobian of (13), which would have $(3 + n_{r,p} + n_{r,n})(n_n + n_p) + 2n_s$ rows and columns. Note that since Θ_i, $i \in \{c_s, c_e, \phi_e\}$, and Π_{ϕ_e} change at every time step, these matrices have to be updated at every time step. Further note that matrices Γ_i and $\Phi_i, i \in \{c_s, c_e, \phi_e\}$ can be pre-computed, which saves computation time.

IV. Simulation Study

In this section, we will study the computational performance and accuracy of the simplified DFN (SDFN) model described in Section II.B. Specifically, the impact of Assumptions [A1]-[A3] made in Section II on the accuracy compared to the full DFN model is studied. Furthermore, the SDFN model is compared to the SPMe [9], and DFN model, in terms of model accuracy and computational performance. Note that in our implementation of the SPMe, besides Assumption [A2], which has also been applied in [9], Assumptions [A1] and [A3] have also been applied. However, as we will show below, these assumptions do not have a significant impact on accuracy. The above simulation studies will be done using two different sets of parameters, obtained from [12] and [13]. Interestingly, the parameter set from [12] has been parametrized from a high-power (HP) cell, while the parameter set from [13] has been parameterized from a high-energy (HE) cell. Some striking differences will be observed below.

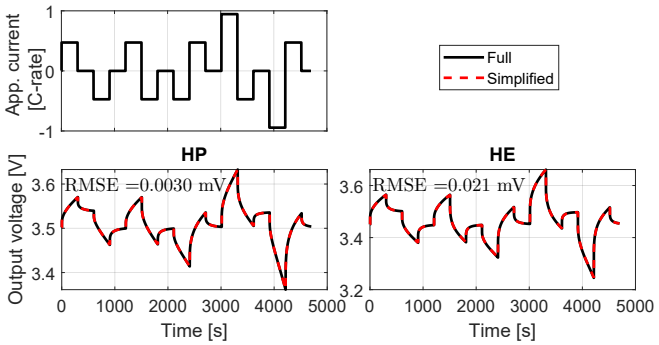

Fig. 2: Comparison of output voltage and root-mean-square error (RMSE) between the full DFN model and the simplified DFN model, with two different parameter sets which represent a high power (HP) battery [12] and a high energy (HE) [13] battery, where the initial state-of-charge is 20 %.

A. Validation of Assumptions [A1]-[A3]

In order to validate Assumptions [A1]-[A3] described in Section II, the full DFN model is compared to the DFN model with varying assumptions at three different initial state-of-charge (SoC) values. The implementation of the full DFN model is based on the implementation described in [6]. The implementation is then adapted for each of the assumptions accordingly, to study the impact Assumptions [A1]-[A3] have on the accuracy of the model.

In Fig. 2, the output voltage of the full and simplified DFN model, and its associated RMS error are shown. We observe that the full and simplified model have no visible difference in output voltage for both parameter sets, which shows that Assumptions [A1]-[A3] do not have a significant effect on accuracy. To further elaborate, in Table I, the root-mean-square error (RMSE) of the output voltage V and normalized electrolyte concentration $c_e/c_{e,0}$ between the full DFN model and the DFN model with varying simplifications is shown for the two different parameters sets. More specifically, each RMSE is computed over three separate simulations, where the initial SoC is selected as 20%, 50%, and 80%. The RMS errors of the normalized electrolyte concentration shows how the assumptions affect the internal states. Note that we have only chosen to use the electrolyte concentration to represent the internal states, as the effect of the assumptions on the other internal states, e.g. c_s, ϕ_e, ϕ_s was similar to the effect on c_e. In both parameter sets, aside from the output voltage RMSE of the HE case, by far the largest RMS error is made with [A2]. It should be noted, however, that even this error is still small relative to the operating values of the output voltage, which is usually between 2.7 V - 4 V, and normalized electrolyte concentration. Remarkably, even though in the HE case the RSME errors resulting from Assumptions [A1] and [A2] are similar in magnitude, they don't add up when used simultaneously, as is the case in the SDFN model. Note that the normalized electrolyte concentration is a ratio, and therefore 0.3×10^{-3} is indeed a very small error. Furthermore, it can be noticed that the RMS errors shown for the HE parameter set are generally larger than the HP parameter set. This can be explained by the fact that the solid-phase diffusion is less limiting, while solid-phase conductivity is higher in the HP parameter set than in the HE parameter set. Therefore, over-potentials are generally higher in the HE case than in the HP case, which in turn leads to larger concentration gradients in the HE case. Higher over-potentials violate [A1], while larger deviations from the average electrolyte concentration violates [A2].

TABLE I: The effect of assumptions [A1]-[A3] introduced in Section II on RMS error (RMSE) for the high-power (HP) [12] and high-energy (HE) [13] parameter set.

RMSE	V [mV]		$c_e/c_{e,0}$ [10^{-3}]	
	HP [12]	HE [13]	HP [12]	HE [13]
[A1]	0.00045	0.017	0.0013	0.0069
[A2]	0.0047	0.021	0.023	0.30
[A3]	0.00064	0.00040	0.0019	0.00016
[A1]-[A3] (SDFN)	0.0047	0.021	0.024	0.30

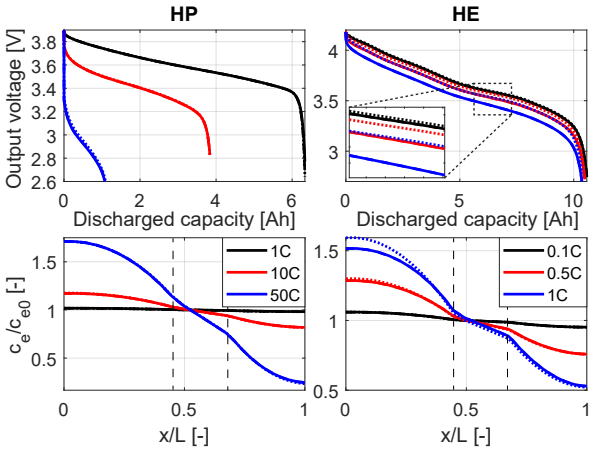

Fig. 3: Discharge curve and normalized electrolyte concentration for the DFN model (solid lines), SDFN model (dashed lines), and SPMe (dotted lines) at several C-rates, simulated for the high-power (HP) [12] and the high-energy (HE) [13] parameter set.

B. Model Comparison

In Fig. 3, discharge curves and normalized electrolyte concentration are shown for both parameter sets at various C-rates for three different models, i.e., DFN model, SDFN model, and SPMe. For both parameter sets, in all cases, there is no visible difference between the DFN and SDFN model, which further shows the validity of [A1]-[A3]. We can further see that for the HP parameter set, there are no visible differences between the three models up to 10 C. At 50 C, the SPMe shows some inaccuracy at the output voltage compared to the DFN model, even though the normalized electrolyte concentration is still close to the DFN model. Interestingly, the opposite is observed with the HE parameter set, where the electrolyte concentration with the SPMe corresponds to the DFN model rather well up to 0.5 C, while the difference in output voltage is substantially worse. Therefore, we observe that an accurate modeling of the output voltage does not

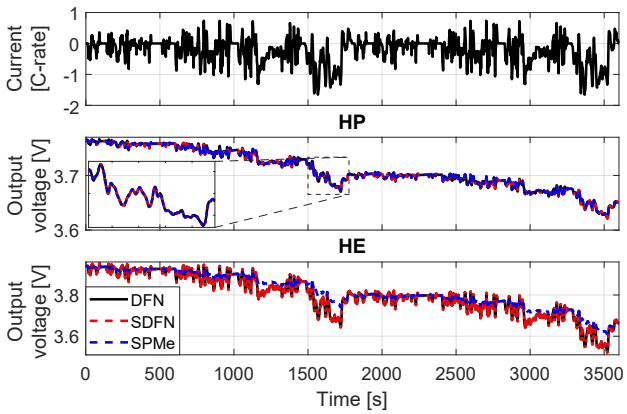

Fig. 4: Output voltage using various models for the high-power (HP) [12] and the high-energy (HE) [13] parameter set. The applied current profile is taken from [14].

necessarily mean an accurate representation of the internal states, and vice versa.

In Fig. 4, the output voltage is shown for the HP and HE battery simulated with a measured current profile of an electric-bike drive cycle [14]. This current profile represents a typical use of a battery, with a relatively fast-changing current, to show how the different battery models perform in such conditions. Consistent with the above analysis, for the HP case, all three models correspond well, while for the HE case the DFN and SDFN are visually identical, and the output voltage of the SPMe is significantly different. In Table II, the RMS errors and computation times corresponding to the simulations shown in Fig 4 are shown. Here, the computation times are an average over 5 simulations. Furthermore, the computation time is measured from the point where the main simulation loop starts to the point where the loop ends. The computation time of pre-defined matrices, outside this loop, is not included, since it is independent of simulation time. The SPMe is the fastest model, as would be expected due to the lower model complexity, although with the HE parameter set, the RMSE is unacceptable. The SDFN model, on the other hand, has very small RMS errors, while the computation time is at least an order of magnitude smaller than the computation time of the DFN model. These results show that by making the Assumptions [A1]-[A3], simplifications can be made that have no significant impact on the model accuracy, while the computation time can be drastically decreased, to achieve a simulation time of over 3600 times faster than real-time.

V. Conclusions

In this paper, we have proposed several simplifications to the so-called Doyle-Fuller-Newman (DFN) model, lending a computationally very efficient implementation. The set of assumptions made to arrive at the SDFN model have been justified by showing that the impact of the assumptions on the accuracy of the DFN model is negligible. This has been done for two different parameter sets, where one represents a high-power (HP) battery and the other a high-energy (HE) battery. Furthermore, the SDFN model has been compared to the DFN model and the SPMe model, for the two parameter sets. While the SPMe is accurate for the HP parameter set, its accuracy is unacceptable for the HE parameter set. The SDFN model, on the other hand, has very small RMS errors, while the computation time is at least an order of magnitude smaller than the computation time of the DFN model. This has shown that by making Assumptions [A1]-[A3], simplifications can be made that have no significant impact on the model accuracy, while the computation time can be drastically decreased, to achieve a simulation time of over 3600 times faster than real-time.

TABLE II: The RMS errors (RMSE) of the output voltage V and computation times corresponding to the simulations shown in Fig. 4.

	RMSE V [mV]		Comp. time [s]	
	HP [12]	HE [13]	HP [12]	HE [13]
DFN	-	-	67.4	22.6
SDFN	0.021	0.046	0.71	0.99
SPMe	0.19	30	0.30	0.39

References

[1] C. Zou, A. G. Kallapur, C. Manzie, and D. Nesic, "PDE battery model simplification for SOC and SOH estimator design," in *Conf. Decision & Control*, 2015.

[2] A. A. Figueiro, A. J. Peixoto, and R. R. Costa, "State of charge estimation and battery balancing control," in *Conf. Decision & Control*, 2015.

[3] R. Klein, N. A. Chaturvedi, J. Christensen, J. Ahmed, R. Findeisen, and A. Kojic, "Optimal charging strategies in lithium-ion battery," in *Amer. Control Conf.*, 2011.

[4] H. E. Perez, X. Hu, S. Dey, and S. J. Moura, "Optimal Charging of Li-Ion Batteries With Coupled Electro-Thermal-Aging Dynamics," *IEEE Trans. Veh. Technol*, 2017.

[5] M. Doyle, T. F. Fuller, and J. Newman, "Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell," *J. Electrochem. Soc.*, 1993.

[6] L. Xia, E. Najafi, Z. Li, H. Bergveld, and M. Donkers, "A computationally efficient implementation of a full and reduced-order electrochemistry-based model for Li-ion batteries," *Applied Energy*, 2017.

[7] G. Fan, K. Pan, and M. Canova, "A comparison of model order reduction techniques for electrochemical characterization of Lithium-ion batteries," in *Conf. Decision & Control*, 2015.

[8] S. Santhanagopalan, Q. Guo, P. Ramadass, and R. E. White, "Review of models for predicting the cycling performance of lithium ion batteries," *J. Power Sources*, 2006.

[9] 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 Trans. Control Syst. Technol.*, 2017.

[10] S. Mazumder and J. Lu, "Faster-Than-Real-Time Simulation of Lithium Ion Batteries with Full Spatial and Temporal Resolution," *Int. J. Electrochem.*, 2013.

[11] M. Torchio, L. Magni, R. B. Gopaluni, R. D. Braatz, and D. M. Raimondo, "LIONSIMBA: A matlab framework based on a finite novolume model suitable for Li-ion battery design, simulation, and control," *J. Electrochem. Soc.*, 2016.

[12] K. A. Smith, C. D. Rahn, and C.-Y. Wang, "Control oriented 1d electrochemical model of lithium ion battery," *Energy Convers. Manag.*, 2007.

[13] N. Jin, D. L. Danilov, P. M. Van den Hof, and M. Donkers, "Parameter estimation of an electrochemistry-based lithium-ion battery model using a two-step procedure and a parameter sensitivity analysis: Parameter estimation of an electrochemistry-based lithium-ion battery model using a two-step procedure and a parameter sensitivity analysis," *Int. J. Energ. Res.*, 2018.

[14] R. Morello, R. D. Rienzo, F. Baronti, R. Roncella, and R. Saletti, "System on chip battery state estimator: E-bike case study," in *IEEE Conf. on Ind. Electron. Soc.*, 2016.