A Novel Numerical Implementation of Electrochemical-Thermal Battery Model for Electrified Powertrains with Conserved Spherical Diffusion and High Efficiency

Hao Yuan[a], Sipeng Zhu[b], Sam Akehurst[a], Liye Wang[c,*], Lifang Wang[c]

[a]*Department of Mechanical Engineering, University of Bath, Bath BA2 7AY, United Kingdom*
[b]*School of Energy and Power Engineering, Shandong University, Jinan 250061, China*
[c]*Institute of Electrical Engineering, Chinese Academy of Sciences, Beijing 100190, China*

Abstract

The performance of batteries in electrified powertrain systems is highly influenced by mass diffusion and electrochemistry which are often ignored in the simulation of these systems due to the lack of a conserved, efficient, and integrable battery model. Therefore, this work numerically implements an electrochemical-thermal battery model with conserved numerical schemes and efficient numerical methods which include Jacobian-based and Jacobian-Free Newton Krylov (JFNK) solvers. The performance of the developed model is evaluated by simulating measurements of a LiFePO$_4$ battery under constant discharge rates and Urban Dynamometer Driving Schedule (UDDS), as well as by a detailed comparison with existing battery models. The comparison highlights two features of our model: (a) negligible mass imbalances in the spherical diffusion modelling, which are five orders of magnitude smaller than those from a recent battery model in the literature; (b) efficient modelling of real-world driving cycles with the computational time two orders of magnitude shorter than that of the literature model. These advanced features indicate that our model can be applied in both fundamental electrochemical-thermal studies of lithium-ion battery and detailed simulations of electrified powertrains as an accurate and efficient sub-model.

Keywords: Lithium-ion battery, Electrochemical-thermal model, Conserved spherical diffusion, Jacobian-based method, Jacobian-Free Newton-Krylov method

---

[*]Corresponding author
  Email address: `yuanhaoars@gmail.com`(Hao Yuan), `wangliye@mail.iee.ac.cn` (Liye Wang)

1. Introduction

Electrifying transportation system with de-carbonized electricity is considered as critical in mitigating greenhouse gas emissions [1–4]. According to the European Commission, electrified vehicles are promising technologies towards the future electrified transportation for the following reasons: the electrified powertrain systems are more efficient than those based on fossil fuels, electricity can be generated from renewable energy sources, and batteries of the electrified powertrain systems could stabilize the power grid and thus enable the integration of renewable sources [5]. Lithium-ion batteries are commonly used in the electrified powertrain systems, which exhibited great potentials for their applications in transportations almost a decade ago [6]. Tremendous advances have been made in lithium-ion battery technology so far to improve energy density and cycle life and reduce costs [7], which bring both climate and economic benefits based on a recent multi-regional case study [8]. However, further improvements of battery technologies are still required to roll out electrified vehicles across the world [7], and the fundamental understanding of the behaviours of lithium-ion batteries is crucial in this process.

The behaviours of lithium-ion batteries are mainly controlled by the diffusion of reacting species, as well as electrochemical reactions [9]. The physics-based battery model predicts the spatially distributed molar concentrations of lithium-ions and potentials in both electrolyte and electrodes [10–13], which is known as the pseudo two-dimensional (P2D) model. In the battery model, the most critical and computationally intensive process is the electrochemistry driven spherical diffusion of lithium-ions in solid particles with variable diffusivity, which requires careful numerical implementation. Ford Versypt and Braatz [14] compared several finite difference schemes in discretising the spherical diffusion equation with variable diffusivity and found that the accuracies of the numerical solutions are dependent on the scheme applied. Besides, Torchio et al. [15] applied a high-order finite difference scheme to discretise the spherical diffusion equation in their battery model. Recently, Liu et al. [16] discretised the spherical diffusion equation for gas transport in coal with the finite difference method to estimate variable diffusivity via matching measurements. A known issue with the finite difference method is mass conservation, which can be resolved by the finite volume method. In the modelling of spherical flames, Chen and his coworkers [17–19] solved unsteady Navier-Stokes equations in a one-dimensional spherical coordinate with the finite volume method. However, the cell averaged finite volume method cannot directly be used to resolve states on the spherical surface where electrochemical reactions

2

occur in the battery model. Therefore, the control volume method [20] was implemented by Zeng et al. [21] and Chayambuka et al. [22] in the spherical diffusion modelling to obtain accurate mole concentration of lithium-ion on the spherical surface. The numerical formulation by Zeng et al. [21] approximates the mass matrix of the time derivative to achieve mass conservation, while Chayambuka et al. [22] applied mass conservation law to each control volume which automatically conserves mass. Apart from the numerical scheme, iterative methods are required to handle variable diffusivity dependent on mole concentrations of lithium-ion in the battery model. Considering the wide applications of the spherical diffusion in different research areas, it is critical to perform a rigorous numerical modelling of the spherical diffusion with variable diffusivity to better understand the fundamental mass transfer process.

Both the mass diffusion and electrochemical reactions in lithium-ion batteries are dependent on temperatures and thus a thermal model is normally coupled with the battery model to formulate an electrochemical-thermal model. This model has been applied by numerous studies to investigate thermal behaviours of lithium-ion batteries [23–29] under constant charge/discharge rates using commercial models. However, real-world driving cycles of electrified powertrains, e.g., Urban Dynamometer Driving Schedule (UDDS), are much more complicated than constant-rate discharges, which significantly affect the performance of batteries as well. In addition, fast-changing, time-dependent current profiles from real-world driving cycles may challenge the convergence of most electrochemical-thermal battery models. Therefore, it is essential to have a battery model which simulates real-world driving cycles accurately and efficiently, as well as bridges the gap between electrochemical-thermal processes and the performance of electrified powertrains.

This work first applies the control volume method to discretise the spherical diffusion equation and the finite volume method for the rest equations whose solutions at boundaries are not as critical as those of the spherical diffusion equation in the battery modelling. Then, the resulting coupled and non-linear equations are solved by two numerical methods with and without Jacobian dependence to achieve fast convergence in simulating real-world driving cycles. For the Jacobian-based method, the Symbolic Math Toolbox in MATLAB is used to formulate the Jacobian matrix which is taken as an input of MATLAB's non-linear system solver. In terms of the Jacobian free approach, a preconditioned Newton-Krylov method [30] is used and the generalized minimum residual method is

applied to approximate the Newton correction based on the Krylov subspace. With the conservative schemes and efficient numerical methods, this study builds a Lithium-ion battery model for Powertrain research (LionPower), which is available at https://github.com/hyuanmech/LionPower. A comprehensive comparison is also carried out among LionPower, an open-source battery model termed LIONSIMBA [15], as well as a commercial battery model called GT-AutoLion, which highlights the advanced features of our model.

2. Numerical Schemes for the Battery Model

The battery model is comprised of three domains: the negative electrode (anode), the separator, and the positive electrode (cathode) in the horizontal x-axis, and the transport of lithium-ions across these domains represents the charge/discharge process, as shown in Fig. 1. In both electrodes, lithium-ions are stored in the solid phase in lattice structures which are emulated by several spherical particles and electrochemical reactions occur on the particles' surfaces which drive the diffusion of lithium-ions in the radial r-axis. Therefore, the transports of lithium-ion in both solid and liquid phases, as well as the corresponding charge distributions formulate the physical picture of the battery model. Considering these processes are temperature-dependent, the thermal characteristics of the battery need to be considered in the modelling, which leads to an electrochemical-thermal model of lithium-ion batteries. Conservative numerical schemes for discretising the governing equations formulating the battery model are discussed in the following sections.

2.1. Spherical Diffusion in Particle

The diffusion of lithium-ions in the solid spherical particle is expressed in Eq. 1 based on Fick's law. The boundary conditions are given by Eq. 2 at the centre and surface of the particle, and the molar flux j is the driving force for the diffusion. The governing equation is solved at all grid points in the x-axis allocated to the electrodes, which is the most computationally intensive part in the battery modelling. Therefore, an efficient and conservative numerical scheme is critical for solving the spherical diffusion problem.

$$r^2 \frac{\partial c_s}{\partial t} = \frac{\partial}{\partial r} \left(D_s r^2 \frac{\partial c_s}{\partial r} \right) \tag{1}$$
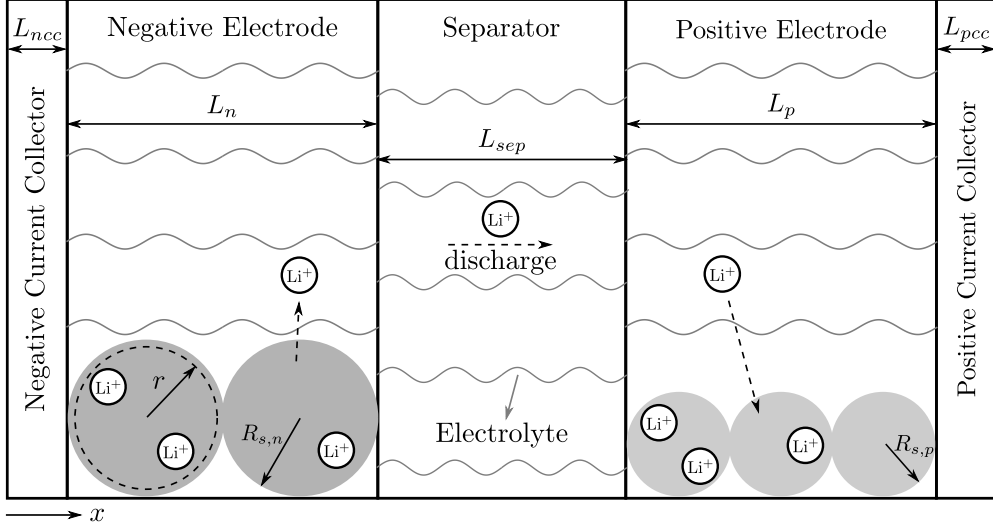
Figure 1: Schematic of the lithium-ion battery during discharge.

$$\left.\frac{\partial c_s}{\partial r}\right|_{r=0} = 0, \quad \left.\frac{\partial c_s}{\partial r}\right|_{r=R_s} = -\frac{j}{D_s} \tag{2}$$

Before the discretisation, the variables in both the governing equation and the boundary conditions are normalised using these correlations: $\hat{c}_s = c_s/c_{s,max}$, $\hat{r} = r/R_s$, and $\hat{j} = j/\bar{j}$, and the resulting equations are expressed in Eqs. 3 and 4. It is worth emphasizing that the normalisation is a crucial step in solving a system of partial differential equations, which leads to a well-scaled Jacobian matrix that can be readily handled by solvers. In this work, the normalisation is applied to all governing equations and boundary conditions prior to the discretisation.

$$\hat{r}^2 \frac{\partial \hat{c}_s}{\partial t} = \frac{\partial}{\partial \hat{r}} \left(\hat{r}^2 \frac{D_s}{R_s^2} \frac{\partial \hat{c}_s}{\partial \hat{r}} \right) \tag{3}$$

$$\left.\frac{\partial \hat{c}_s}{\partial \hat{r}}\right|_{\hat{r}=0} = 0, \quad \left.\frac{\partial \hat{c}_s}{\partial \hat{r}}\right|_{\hat{r}=1} = -\bar{j}\frac{\hat{j}}{D_s}\frac{R_s}{c_{s,max}} \tag{4}$$

The schematic of the spherical diffusion model and its spatial discretisation are shown in Fig. 2, which were previously applied by Zeng et al. [21] and Chayambuka et al. [22]. Furthermore, both studies [21, 22] recommended variable spacing of grid points following Eq. 5, and fine meshes are used in the region near the spherical surface

5

where the gradient of mole concentration changes sharply. This local refinement improves the modelling accuracy of the surface concentration which is critical for computing the rate of progress of the electrochemical reaction.

$$\hat{r} = \frac{H - H^{\beta}}{H - 1}, \quad \beta = \frac{N_s - i}{N_s - 1}, \quad 1 \leqslant i \leqslant N_s \tag{5}$$

This work applies both the control volume method and the variable spacing in the spatial discretisation of the normalised equation. Note that the uniform spacing of grid points in Fig. 2 is for the simplicity of illustration.
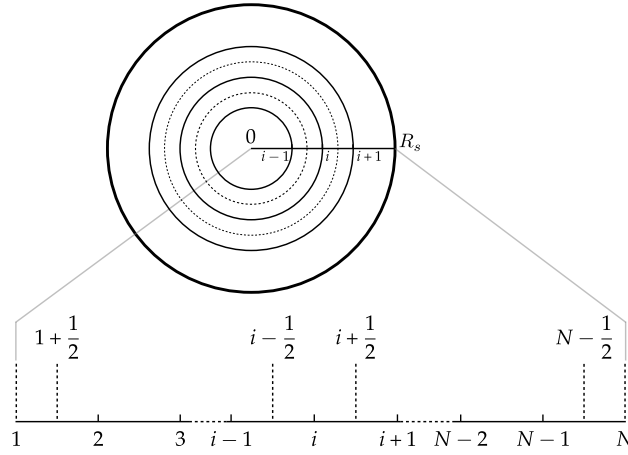
Figure 2: Schematic of the spherical diffusion model and discretisation based on the control volume method. Solid circles represent locations where grid points locate, while dashed circles and dashed vertical lines denote surfaces of control volume elements.

Integrating both sides of Eq. 3 from $i - 1/2$ to $i + 1/2$ gives

$$\frac{1}{3} \left(\hat{r}_{i+\frac{1}{2}}^3 - \hat{r}_{i-\frac{1}{2}}^3 \right) \frac{\partial \hat{c}_{s,i}}{\partial t} = \frac{D_{s,i+\frac{1}{2}}}{R_s^2} \hat{r}_{i+\frac{1}{2}}^2 \frac{\partial \hat{c}_{s,i+\frac{1}{2}}}{\partial \hat{r}} - \frac{D_{s,i-\frac{1}{2}}}{R_s^2} \hat{r}_{i-\frac{1}{2}}^2 \frac{\partial \hat{c}_{s,i-\frac{1}{2}}}{\partial \hat{r}} \tag{6}$$

The left-hand side (LHS) of Eq. 6 represents the change of normalised mole concentration in a spherical shell i, which equals to the net-flux via the left and right interfaces of the control volume. After approximating the interface fluxes with the central difference method, we obtain

$$\frac{1}{3} \left(\hat{r}_{i+\frac{1}{2}}^3 - \hat{r}_{i-\frac{1}{2}}^3 \right) \frac{\partial \hat{c}_{s,i}}{\partial t} = \frac{D_{s,i+\frac{1}{2}}}{R_s^2} \hat{r}_{i+\frac{1}{2}}^2 \frac{\hat{c}_{s,i+1} - \hat{c}_{s,i}}{\hat{r}_{i+1} - \hat{r}_i} - \frac{D_{s,i-\frac{1}{2}}}{R_s^2} \hat{r}_{i-\frac{1}{2}}^2 \frac{\hat{c}_{s,i} - \hat{c}_{s,i-1}}{\hat{r}_i - \hat{r}_{i-1}} \tag{7}$$

The discretised equations for the centre and the boundary volumes are expressed in Eqs. 8 and 9, respectively.

6

$$\frac{(\hat{r}_2 - \hat{r}_1)^3}{24} \frac{\partial \hat{c}_{s,1}}{\partial t} = \frac{D_{s,1+\frac{1}{2}}}{R_s^2} \hat{r}_{1+\frac{1}{2}}^2 \frac{\hat{c}_{s,2} - \hat{c}_{s,1}}{\hat{r}_2 - \hat{r}_1} \tag{8}$$

$$\frac{1}{3} \left[\hat{r}_N^3 - \left(\frac{\hat{r}_N + \hat{r}_{N-1}}{2} \right)^3 \right] \frac{\partial \hat{c}_{s,N}}{\partial t} = -\frac{\bar{j} \hat{r}_N^2}{R_s c_{s,max}} \hat{j} - \frac{D_{s,N-\frac{1}{2}}}{R_s^2} \hat{r}_{N-\frac{1}{2}}^2 \frac{\hat{c}_{s,N} - \hat{c}_{s,N-1}}{\hat{r}_N - \hat{r}_{N-1}} \tag{9}$$

Combining Eq. 7, Eq. 8, and Eq. 9 gives the matrix form of the spherical diffusion equation in Eq. 10. The conservative spatial discretisation of the spherical diffusion equation in this work is inspired by a very recent numerical study [22]. However, the temporal discretisation in [22] assumed that the variable diffusion coefficient relies on the mole concentrations of lithium-ion from the previous time point instead of the current one, which will result in noticeable errors with either a large time step or fast-changing current inputs from powertrain systems. Therefore, this work applies the Crank-Nicolson scheme for the time derivative in Eq. 10 and makes no assumptions in computing the variable diffusion coefficient, which achieves accurate modelling of the spherical diffusion. The discretised algebraic equation based on the Crank-Nicolson scheme is expressed in Eq. 11.

$$\mathbf{A_s} \frac{\partial \hat{\mathbf{c}}_{\mathbf{s}}}{\partial t} = \mathbf{f}(\hat{\mathbf{c}}_{\mathbf{s}}) \tag{10}$$

$$\mathbf{A_s} \frac{\hat{\mathbf{c}}_{\mathbf{s}}^{\mathbf{n+1}} - \hat{\mathbf{c}}_{\mathbf{s}}^{\mathbf{n}}}{\Delta t} = \frac{1}{2} \left[\mathbf{f}(\hat{\mathbf{c}}_{\mathbf{s}}^{\mathbf{n}}) + \mathbf{f}(\hat{\mathbf{c}}_{\mathbf{s}}^{\mathbf{n+1}}) \right] \tag{11}$$

2.2. Transport in Electrolyte

The transport of lithium-ions in the electrolyte is a typical multilayer diffusion problem. The governing equation and the boundary conditions are shown in Eqs. 12 and 13 where $a_s = 3\epsilon_s / R_s$, and the requirement of matching interface fluxes are expressed in Eq. 14 where D_e^{eff} represents the effective diffusion coefficient considering the volume fraction of the electrolyte in each domain and is given by $D_e^{eff} = D_e \epsilon_e^\gamma$. Efforts have been made by Hickson et al. [31] and March and Carr [32] to solve the multilayer diffusion problem with finite difference schemes and control volume schemes, respectively. Hickson et al. [31] matched the diffusivities by approximating the diffusion coefficient on the interface using adjacent values. In comparison, March and Carr [32] split the control volume element on the

7

interface into two and then applied the control volume schemes to these small elements, which, however, has not been verified for pure Neumann boundary conditions, e.g., Eq. 13. In this case, we follow the idea of Hickson et al. [31] and apply it with the finite volume scheme.

$$\epsilon_e \frac{\partial c_e}{\partial t} = \frac{\partial}{\partial x} \left(D_e^{eff} \frac{\partial c_e}{\partial x} \right) + a_s (1 - t_+) j \tag{12}$$

$$\left. \frac{\partial c_e}{\partial x} \right|_{x=0} = \left. \frac{\partial c_e}{\partial x} \right|_{x=L} = 0 \tag{13}$$

$$D_{e,n}^{eff} \left. \frac{\partial c_e}{\partial x} \right|_{x=L_n} = D_{e,sep}^{eff} \left. \frac{\partial c_e}{\partial x} \right|_{x=0_{sep}}, \quad D_{e,sep}^{eff} \left. \frac{\partial c_e}{\partial x} \right|_{x=L_{sep}} = D_{e,p}^{eff} \left. \frac{\partial c_e}{\partial x} \right|_{x=0_p} \tag{14}$$

To normalise the governing equation and boundary conditions, the following correlations: $\hat{c}_e = c_e/c_{e0}$, $\hat{x} = x/L$, $\hat{j} = j/\bar{j}$ are used, which result in

$$\epsilon_e \frac{\partial \hat{c}_e}{\partial t} = \frac{\partial}{\partial \hat{x}} \left(\frac{D_e^{eff}}{L^2} \frac{\partial \hat{c}_e}{\partial \hat{x}} \right) + a_s (1 - t_+) \frac{\bar{j}}{c_{e0}} \hat{j} \tag{15}$$

$$\left. \frac{\partial \hat{c}_e}{\partial \hat{x}} \right|_{\hat{x}=0} = \left. \frac{\partial \hat{c}_e}{\partial \hat{x}} \right|_{\hat{x}=1} = 0 \tag{16}$$

The mesh of the negative electrode, the separator, and the positive electrode is shown in Fig. 3 and a uniform spacing is used in each domain. The effective diffusion coefficient on the interface is interpolated from the two adjacent values. Based on the example shown in Fig. 3, the effective diffusion coefficient on the interface of the negative electrode and the separator is shown in Eq. 17.

$$D_{e,ns}^{eff} = \frac{D_{e,N_n}^{eff} \Delta x_{sep} + D_{e,1_{sep}}^{eff} \Delta x_n}{\Delta x_n + \Delta x_{sep}} \tag{17}$$

Integrating both sides of Eq. 15 on a control volume element i of a random domain from $i - 1/2$ to $i + 1/2$ and applying the central difference scheme to approximate the fluxes on the surfaces of the control volume element, we
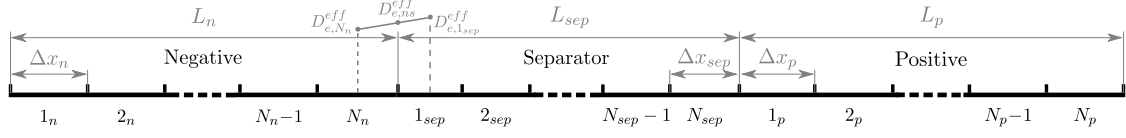
8

Figure 3: Grid points of the negative electrode, the separator, and the positive electrode with the thickness of L_n, L_{sep}, and L_p, respectively. The grid points are meshed uniformly in each domain, and Δx_n, Δx_{sep}, and Δx_p denote the grid sizes. The effective diffusion coefficient on the inner interface is interpolated from the values of the two adjacent volume elements.

obtain

$$\frac{\partial \hat{c}_{e,i}}{\partial t} = \frac{D_{e,i}^{eff} + D_{e,i+1}^{eff}}{2\epsilon_e L^2} \frac{\hat{c}_{e,i+1} - \hat{c}_{e,i}}{\Delta \hat{x}} - \frac{D_{e,i-1}^{eff} + D_{e,i}^{eff}}{2\epsilon_e L^2} \frac{\hat{c}_{e,i} - \hat{c}_{e,i-1}}{\Delta \hat{x}} + \frac{a_s (1 - t_+)}{\epsilon_e} \frac{\bar{j}}{c_{e0}} \hat{j}_i \Delta \hat{x} \tag{18}$$

Note that if the node i belongs to the separator, $\hat{j}_i$ becomes zero as the electrochemical reactions only occur in the negative and positive electrodes. Incorporating the boundary conditions in Eq. 18, the discretised equations for the two boundary volume elements are described by Eqs. 19 and 20.

$$\frac{\partial \hat{c}_{e,1_n}}{\partial t} = \frac{D_{e,1_n}^{eff} + D_{e,2_n}^{eff}}{2\epsilon_e^n L^2} \frac{\hat{c}_{e,2_n} - \hat{c}_{e,1_n}}{\Delta \hat{x}_n} + \frac{a_{s,n} (1 - t_+)}{\epsilon_e^n} \frac{\bar{j}}{c_{e0}} \hat{j}_{1_n} \Delta \hat{x}_n \tag{19}$$

$$\frac{\partial \hat{c}_{e,N_p}}{\partial t} = \frac{D_{e,N_p-1}^{eff} + D_{e,N_p}^{eff}}{2\epsilon_e^p L^2} \frac{\hat{c}_{e,N_p-1} - \hat{c}_{e,N_p}}{\Delta \hat{x}_p} + \frac{a_{s,p} (1 - t_+)}{\epsilon_e^p} \frac{\bar{j}}{c_{e0}} \hat{j}_{N_p} \Delta \hat{x}_p \tag{20}$$

With the interpolated effective diffusion coefficient for the interface between the negative electrode and the separator in Eq. 17, the discretised equation for the control volume element N_n can be derived, which is given by

$$\frac{\partial \hat{c}_{e,N_n}}{\partial t} = \frac{D_{e,ns}^{eff}}{\epsilon_e L^2} \frac{\hat{c}_{e,1_{sep}} - \hat{c}_{e,N_n}}{\left(\Delta \hat{x}_n + \Delta \hat{x}_{sep}\right)/2} - \frac{D_{e,N_n-1}^{eff} + D_{e,N_n}^{eff}}{2\epsilon_e L^2} \frac{\hat{c}_{e,N_n} - \hat{c}_{e,N_n-1}}{\Delta \hat{x}_n} + \frac{a_s (1 - t_+)}{\epsilon_e} \frac{\bar{j}}{c_{e0}} \hat{j}_{N_n} \Delta \hat{x}_n \tag{21}$$

Following the same approach, the governing equations for the control volume elements 1_{sep}, N_{sep}, and 1_p can be discretised to match the interface diffusivities.

The discretised equations above can be written in the matrix form, as shown in Eq. 22 with the source term $\mathbf{S}_{c_e}^{j}$ representing the driving force for the diffusion. After approximating the time derivative with the Crank-Nicolson scheme, the fully discretised algebraic equation is shown in Eq. 23.

$$\frac{\partial \hat{\mathbf{c}}_{\mathbf{e}}}{\partial t} = \mathbf{g}\left(\hat{\mathbf{c}}_{\mathbf{e}}\right) + \mathbf{S}_{\mathbf{c}_{\mathbf{e}}}^{\mathbf{j}}\left(\hat{\mathbf{j}}\right) \tag{22}$$

$$\frac{\hat{\mathbf{c}}_{\mathbf{e}}^{\mathbf{n+1}} - \hat{\mathbf{c}}_{\mathbf{e}}^{\mathbf{n}}}{\Delta t} = \frac{1}{2}\left[\mathbf{g}\left(\hat{\mathbf{c}}_{\mathbf{e}}^{\mathbf{n}}\right) + \mathbf{g}\left(\hat{\mathbf{c}}_{\mathbf{e}}^{\mathbf{n+1}}\right)\right] + \mathbf{S}_{\mathbf{c}_{\mathbf{e}}}^{\mathbf{j}}\left(\hat{\mathbf{j}}\right) \tag{23}$$

2.3. Potential in Electrolyte

The potentials in the liquid phase are affected by the molar concentrations of lithium-ion in the electrolyte, as well as the molar fluxes from the electrochemical reactions, as described in Eq. 24 where κ^{eff} denotes the effective conductivity and is given by $\kappa^{ref} = \kappa \epsilon_e^{\gamma}$. Zero potential flux is assigned to both boundaries in Eq. 25. Note that Poisson's equation (Eq. 24) with pure Neumann boundary conditions (Eq. 25) has infinite number of solutions. A Dirichlet boundary condition, $\phi_{e,0}^{ref} = 0$, is used to anchor the numerical solution since we are only interested in differences of the potentials.

$$\frac{\partial}{\partial x}\left(\kappa^{eff}\frac{\partial \phi_e}{\partial x}\right) + \frac{\partial}{\partial x}\left[\frac{2RT\kappa^{eff}}{F}\left(t_+ - 1\right)\left(1 + \frac{dlnf_{\pm}}{dlnc_e}\right)\frac{\partial lnc_e}{\partial x}\right] + a_s F j = 0 \tag{24}$$

$$\left.\frac{\partial \phi_e}{\partial x}\right|_{x=0} = \left.\frac{\partial \phi_e}{\partial x}\right|_{x=L} = 0 \tag{25}$$

The normalised governing equation and boundary conditions are shown in Eqs. 26 and 27, respectively. It is assumed that $f_{\pm}$ is a constant to simplify the following numerical implementation.

$$\frac{\partial}{\partial \hat{x}}\left(\kappa^{eff}\frac{\partial \phi_e}{\partial \hat{x}}\right) + \frac{2R\hat{T}T_{ref}\kappa^{eff}}{F}\frac{\partial}{\partial \hat{x}}\left(\frac{\kappa^{eff}}{\hat{c}_e}\frac{\partial \hat{c}_e}{\partial \hat{x}}\right) + a_s F \bar{j} L^2 \hat{j} = 0 \tag{26}$$

$$\left.\frac{\partial \phi_e}{\partial \hat{x}}\right|_{\hat{x}=0} = \left.\frac{\partial \phi_e}{\partial \hat{x}}\right|_{\hat{x}=1} = 0 \tag{27}$$

Integrating Eq. 26 on a control volume element i of a random domain from $i - 1/2$ to $i + 1/2$ and applying the central difference scheme to approximate the fluxes on the surfaces of the control volume element, we obtain

10

$$\frac{\kappa_i^{eff} + \kappa_{i+1}^{eff}}{2} \frac{\phi_{e,i+1} - \phi_{e,i}}{\Delta \hat{x}} - \frac{\kappa_{i-1}^{eff} + \kappa_i^{eff}}{2} \frac{\phi_{e,i} - \phi_{e,i-1}}{\Delta \hat{x}} + \frac{2R\hat{T}T_{ref}(t_+ - 1)}{F} \left(\frac{\kappa_i^{eff} + \kappa_{i+1}^{eff}}{\hat{c}_{e,i} + \hat{c}_{e,i+1}} \frac{\hat{c}_{e,i+1} - \hat{c}_{e,i}}{\Delta \hat{x}} - \right.$$

$$\left. \frac{\kappa_{i-1}^{eff} + \kappa_i^{eff}}{\hat{c}_{e,i-1} + \hat{c}_{e,i}} \frac{\hat{c}_{e,i} - \hat{c}_{e,i-1}}{\Delta \hat{x}} \right) + a_s F \bar{j} L^2 \hat{j}_i \Delta \hat{x} = 0 \tag{28}$$

With the Dirichlet boundary condition at $\hat{x} = 0$, the flux at the same position is approximated based on the Taylor expansion of $\phi_{e,1_n}$ and $\phi_{e,2_n}$ with respect to $\phi_{e,0}^{ref}$. Besides, the conductivity at $\hat{x} = 0$ is extrapolated from $\kappa_{1_n}^{eff}$ and $\kappa_{2_n}^{eff}$. The discretised equation with the Dirichlet boundary condition is shown in Eq. 29, while the one with the Neumann boundary condition can be derived straightforwardly, as given by Eq. 30.

$$\frac{\kappa_{1_n}^{eff} + \kappa_{2_n}^{eff}}{2} \frac{\phi_{e,2_n} - \phi_{e,1_n}}{\Delta \hat{x}_n} - \frac{3\kappa_{1_n}^{eff} - \kappa_{2_n}^{eff}}{2} \frac{9\phi_{e,1_n} - \phi_{e,2_n} - 8\phi_{e,0}^{ref}}{3\Delta \hat{x}_n} + \frac{2R\hat{T}T_{ref}(t_+ - 1)}{F} \left(\frac{\kappa_{1_n}^{eff} + \kappa_{2_n}^{eff}}{\hat{c}_{e,1_n} + \hat{c}_{e,2_n}} \frac{\hat{c}_{e,2_n} - \hat{c}_{e,1_n}}{\Delta \hat{x}_n} \right) + a_s F \bar{j} L^2 \hat{j}_{1_n} \Delta \hat{x}_n = 0$$

$$\tag{29}$$

$$\frac{\kappa_{N_p-1}^{eff} + \kappa_{N_p}^{eff}}{2} \frac{\phi_{e,N_p-1} - \phi_{e,N_p}}{\Delta \hat{x}_p} + \frac{2R\hat{T}T_{ref}(t_+ - 1)}{F} \left(\frac{\kappa_{N_p-1}^{eff} + \kappa_{N_p}^{eff}}{\hat{c}_{e,N_p-1} + \hat{c}_{e,N_p}} \frac{\hat{c}_{e,N_p-1} - \hat{c}_{e,N_p}}{\Delta \hat{x}_p} \right) + a_s F \bar{j} L^2 \hat{j}_{N_p} \Delta \hat{x}_p = 0 \tag{30}$$

Similar to the transport problem in the electrolyte, the discretised equations for the four control volume elements adjacent to the two internal interfaces are derived with the interpolated physical quantity (electrolyte conductivity) on the interfaces.

The matrix form describing the potential distribution in the electrolyte is expressed in Eq. 31, which is not time-dependent as the change of potential is considered to occur instantaneously, especially compared with the diffusion.

$$\mathbf{h}(\boldsymbol{\Phi_e}) + \mathbf{g_1}(\hat{\mathbf{c}}_e) + \mathbf{S}_{\boldsymbol{\Phi_e}}^{\mathbf{j}}(\hat{\mathbf{j}}) = \mathbf{0} \tag{31}$$

2.4. Potential in Electrode

The governing equation and boundary conditions for the potentials in the solid phase are given by Eqs. 32 and 33 where $\sigma^{eff} = \sigma \epsilon_s$ is the effective conductivity. The normalised equations are shown in Eqs. 34 and 35.

$$\frac{\partial}{\partial x}\left(\sigma^{eff}\frac{\partial \phi_s}{\partial x}\right) = a_s F j \tag{32}$$

$$-\sigma_n^{eff}\left.\frac{\partial \phi_s}{\partial x}\right|_{x=0} = -\sigma_p^{eff}\left.\frac{\partial \phi_s}{\partial x}\right|_{x=L} = I, \quad \left.\frac{\partial \phi_s}{\partial x}\right|_{x=L_n} = \left.\frac{\partial \phi_s}{\partial x}\right|_{x=L_n+L_{sep}} = 0 \tag{33}$$

$$\frac{\partial}{\partial \hat{x}}\left(\sigma^{eff}\frac{\partial \phi_s}{\partial \hat{x}}\right) = a_s F \bar{j} L^2 \hat{j} \tag{34}$$

$$-\sigma_n^{eff}\left.\frac{\partial \phi_s}{\partial \hat{x}}\right|_{\hat{x}=0} = -\sigma_p^{eff}\left.\frac{\partial \phi_s}{\partial \hat{x}}\right|_{\hat{x}=1} = LI, \quad \left.\frac{\partial \phi_s}{\partial \hat{x}}\right|_{\hat{x}=L_n/L} = \left.\frac{\partial \phi_s}{\partial \hat{x}}\right|_{\hat{x}=(L_n+L_{sep})/L} = 0 \tag{35}$$

The potentials in the negative and positive electrodes are handled separately. The spatial discretisation is only carried out for the negative electrode in this work, which can be similarly extended to the positive electrode. With the finite volume method, the discretised governing equation and boundary conditions are expressed in Eqs. 36, 37, and 38, which can be written in the matrix form in Eq. 39.

$$\sigma^{eff}\left(\frac{\phi_{s,i}-\phi_{s,i-1}}{\Delta \hat{x}_n} - \frac{\phi_{s,i+1}-\phi_{s,i}}{\Delta \hat{x}_n}\right) = a_{s,n} F \bar{j} L^2 \hat{j}_i \Delta \hat{x}_n \tag{36}$$

$$\sigma^{eff}\frac{\phi_{s,2_n}-\phi_{s,1_n}}{\Delta \hat{x}_n} + LI = a_{s,n} F \bar{j} L^2 \hat{j}_{1_n} \Delta \hat{x}_n \tag{37}$$

$$\sigma^{eff}\frac{\phi_{s,N_n-1}-\phi_{s,N_n}}{\Delta \hat{x}_n} = a_{s,n} F \bar{j} L^2 \hat{j}_{N_n} \Delta \hat{x}_n \tag{38}$$

$$\mathbf{w}(\mathbf{\Phi_s}) - \mathbf{S}_{\mathbf{\Phi_s}}^{\mathbf{j}}\left(\hat{\mathbf{j}}\right) = \mathbf{0} \tag{39}$$

It is clear that the potential in the solid phase is described by Poisson's equation with pure Neumann boundary conditions, indicating Eq. 39 has infinite number of solutions. Note that the potential in the solid phase is related to

the other variables via the Butler-Volmer kinetics, as shown in Eqs. 40 and 41, which helps to anchor the solution. The normalised equation is expressed in Eq. 42 with its matrix form given by Eq. 43.

$$j = kc_e^{\alpha_n} \left(c_{s,max} - c_{s,surf}\right)^{\alpha_n} c_{s,surf}^{\alpha_p} \left[exp\left(\frac{\alpha_n F}{RT}\eta\right) - exp\left(-\frac{\alpha_p F}{RT}\eta\right)\right] \tag{40}$$

$$\eta = \phi_s - \phi_e - U \tag{41}$$

$$\hat{j} = \frac{k}{\bar{j}} (\hat{c}_e c_{e0})^{\alpha_n} \left[c_{s,max}\left(1 - \hat{c}_{s,surf}\right)\right]^{\alpha_n} \left(\hat{c}_{s,surf} c_{s,max}\right)^{\alpha_p} \left[exp\left(\frac{\alpha_n F}{R\hat{T}T_{ref}}\eta\right) - exp\left(-\frac{\alpha_p F}{R\hat{T}T_{ref}}\eta\right)\right] \tag{42}$$

$$\mathbf{S}\left(\hat{\mathbf{j}}\right) = \mathbf{0} \tag{43}$$

2.5. Heat Conduction

As mentioned previously, the electrochemical model is temperature-dependent, and thus a thermal model is critical to accurately simulate the behaviours of batteries. The governing equation of the thermal model in 1D is shown in Eq. 44 with the source term q representing the volumetric heat generation from various sources. Note that the negative and positive current collectors are included in the thermal model, which can be considered as pure resistors with their ohmic heat generation expressed in Eq. 46. The reaction heat, the reversible entropic heat, and the ohmic heat appear in both electrodes, while the separator only has the ohmic heat. The expressions for computing these heats are shown in Eqs. 47, 48, and 49. The convective heat transfer is applied to the left of the negative current collector and the right of the positive current collector as the boundary condition, which is described in Eq. 50.

$$\rho c_p \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(\lambda \frac{\partial T}{\partial x}\right) + q \tag{44}$$

$$q = q_{ohm,cc} + q_{rxn} + q_{rev} + q_{ohm} \tag{45}$$

13

$$q_{ohm,cc} = \frac{I^2}{\sigma_{cc}} \tag{46}$$

$$q_{rxn} = a_s F j \eta \tag{47}$$

$$q_{rev} = a_s F j T \frac{\partial U}{\partial T} \tag{48}$$

$$q_{ohm} = \sigma^{eff} \left(\frac{\partial \phi_s}{\partial x} \right)^2 + \kappa^{eff} \left(\frac{\partial \phi_e}{\partial x} \right)^2 + \frac{2RT \left(1 - t_+ \right) \kappa^{eff}}{F} \frac{\partial ln c_e}{\partial x} \frac{\partial \phi_e}{\partial x} \tag{49}$$

$$-\lambda_{ncc} \left. \frac{\partial T}{\partial x} \right|_{x=0_{ncc}} = h \left(T_{amb} - T_{1_{ncc}} \right), \quad -\lambda_{pcc} \left. \frac{\partial T}{\partial x} \right|_{x=L_{pcc}} = h \left(T_{N_{pcc}} - T_{amb} \right) \tag{50}$$

The finite volume scheme is used to discretise the normalised governing equation and boundary conditions in Eqs. 51 and 52. The discretised governing equation for an internal control volume element of a random domain is given by Eq. 53. To handle the Robin boundary conditions in Eq. 52, Taylor expansion is applied to the two adjacent points of the boundary with respect to the boundary surface, and the second-order approximation for the boundary thermal flux can be obtained, as shown in Eqs. 54 and 55. With the addition of negative and positive current collectors, four internal interfaces are formed and thus eight control volume elements need to be discretised with interpolated thermal conductivities on these interfaces.

$$\rho c_p \frac{\partial \hat{T}}{\partial t} = \frac{\partial}{\partial \hat{x}} \left(\frac{\lambda}{L^2} \frac{\partial \hat{T}}{\partial \hat{x}} \right) + \frac{q}{T_{ref}} \tag{51}$$

$$-\lambda_{ncc} \left. \frac{\partial \hat{T}}{\partial \hat{x}} \right|_{\hat{x}_{ncc}=0} = \frac{hL \left(T_{amb} - T_{1_{ncc}} \right)}{T_{ref}}, \quad -\lambda_{pcc} \left. \frac{\partial \hat{T}}{\partial \hat{x}} \right|_{\hat{x}_{pcc}=1} = \frac{hL \left(T_{N_{pcc}} - T_{amb} \right)}{T_{ref}} \tag{52}$$

$$\rho c_p \frac{\partial \hat{T}_i}{\partial t} = \frac{\lambda}{L^2} \left(\frac{\hat{T}_{i+1} - \hat{T}_i}{\Delta \hat{x}} - \frac{\hat{T}_i - \hat{T}_{i-1}}{\Delta \hat{x}} \right) + \frac{q}{T_{ref}} \Delta \hat{x} \tag{53}$$

$$\rho_{ncc} c_{p,ncc} \frac{\partial \hat{T}_{1_{ncc}}}{\partial t} = \frac{\lambda_{ncc}}{L^2} \left(\frac{\hat{T}_{2_{ncc}} - \hat{T}_{1_{ncc}}}{\Delta \hat{x}_{ncc}} - \frac{9\hat{T}_{1_{ncc}} - \hat{T}_{2_{ncc}} - 8T_{amb}}{3\Delta \hat{x}_{ncc} + 8\lambda_{ncc}/h} \right) + \frac{q_{1_{ncc}}}{T_{ref}} \Delta \hat{x}_{ncc} \tag{54}$$

$$\rho_{pcc} c_{p,pcc} \frac{\partial \hat{T}_{N_{pcc}}}{\partial t} = \frac{\lambda_{pcc}}{L^2} \left(\frac{-9\hat{T}_{N_{pcc}} + \hat{T}_{N_{pcc}-1} + 8T_{amb}}{3\Delta \hat{x}_{pcc} + 8\lambda_{pcc}/h} - \frac{\hat{T}_{N_{pcc}} - \hat{T}_{N_{pcc}-1}}{\Delta \hat{x}_{pcc}} \right) + \frac{q_{N_{pcc}}}{T_{ref}} \Delta \hat{x}_{pcc} \tag{55}$$

The matrix form of the heat conduction model is given by Eq. 56 where $\mathbf{S_T^j}$ denotes the source term for the heat generation. Based on the Crank-Nicolson scheme, the fully discretised algebraic equation is expressed in Eq. 57.

$$\rho \mathbf{c_p} \frac{\partial \hat{\mathbf{T}}}{\partial t} = \mathbf{z}\left(\hat{\mathbf{T}}\right) + \mathbf{S_T^j}\left(\hat{\mathbf{j}}\right) \tag{56}$$

$$\rho \mathbf{c_p} \frac{\hat{\mathbf{T}}^{n+1} - \hat{\mathbf{T}}^n}{\Delta t} = \frac{1}{2}\left[\mathbf{z}\left(\hat{\mathbf{T}}^n\right) + \mathbf{z}\left(\hat{\mathbf{T}}^{n+1}\right)\right] + \mathbf{S_T^j}\left(\hat{\mathbf{j}}\right) \tag{57}$$

3. Methods for Non-linear Equations

The spatial discretisation of the governing equations produces a system of differential algebraic equations (DAEs) which is comprised of Eqs. 10, 22, 31, 39, 43, and 56 and can be solved by integrators, e.g., the IDA package from SUNDIALS [33] used by LIONSIMBA [15]. Note that MATLAB has its own DAE solver, ode15s, which was successfully applied by the authors in solving a combustion model coupled with chemical kinetics [34] and can certainly be implemented to solve the electrochemical-thermal model. However, a potential issue with these integrators is the relatively slow convergence rate with fast-changing inputs, which may limit their applications in the powertrain modelling.

In this work, the DAE system is further discretised by the Crank-Nicolson scheme, which formulates non-linear and coupled algebraic equations, including Eqs. 11, 23, 31, 39, 43, and 57. The numerical methods with and without Jacobian matrix are used to solve these algebraic equations, the general form of which is given by

$$\mathbf{F}(\mathbf{u}) = \mathbf{0} \tag{58}$$

where $\mathbf{F}$ is the collection of all algebraic equations, and $\mathbf{u}$ represents the state vector including c_s, c_e, ϕ_e, ϕ_s, T, j, as well as the current I which normally varies with time in electrified powertrain systems.

3.1. Jacobian-based Methods

Newton's method is probably the most commonly used Jacobian-based approach for solving Eq. 58. A multivariate Taylor expansion of $\mathbf{F}(\mathbf{u})$ with respect to the current solution $\mathbf{u^k}$ gives

$$\mathbf{F}\left(\mathbf{u^{k+1}}\right) = \mathbf{F}\left(\mathbf{u^k}\right) + \mathbf{F}'\left(\mathbf{u^k}\right)\left(\mathbf{u^{k+1}} - \mathbf{u^k}\right) + O\left[\left(\mathbf{u^{k+1}} - \mathbf{u^k}\right)^2\right] \tag{59}$$

Ignoring the high-order terms and setting $\mathbf{F}\left(\mathbf{u^{k+1}}\right) = \mathbf{0}$, Newton's method is derived by the following iteration starting from an initial estimation $\mathbf{u^0}$,

$$\mathbf{u^{k+1}} = \mathbf{u^k} + \delta\mathbf{u^k}, \quad \mathbf{J}\left(\mathbf{u^k}\right)\delta\mathbf{u^k} = -\mathbf{F}\left(\mathbf{u^k}\right) \tag{60}$$

where $\mathbf{J}\left(\mathbf{u^k}\right) = \mathbf{F}'\left(\mathbf{u^k}\right)$ is the Jacobian matrix. The iteration is terminated once $\left\|\mathbf{F}\left(\mathbf{u^k}\right)\right\|$ is small enough based on the user-defined convergence criterion.

The most challenging task in implementing Newton's method is the formulation of the Jacobian matrix, especially for the non-linear and coupled algebraic system in this work. With MATLAB's Symbolic Math Toolbox, the Jacobian matrix can be derived analytically. Since the diagonal blocks of the Jacobian matrix are computed explicitly when evaluating $\mathbf{F}\left(\mathbf{u^k}\right)$, they are directly used in formulating the Jacobian matrix, and those non-diagonal blocks rely on the analytic expressions. To further accelerate the formulation of the Jacobian matrix, the parallel computation is applied wherever possible, which helps to compensate for MATLAB's inefficient evaluation of large functions.

Alternatively, with the finite difference method, the Jacobian matrix can be approximated at a fixed point, and an example is given by Eq. 61

$$\mathbf{J^k} = \frac{\mathbf{F}\left(\mathbf{u^k} + \xi\right) - \mathbf{F}\left(\mathbf{u^k} - \xi\right)}{2\xi}, \quad \xi = eps^{1/3} \tag{61}$$

where eps is the floating-point relative accuracy, 2^{-52}. Since most numerical methods for solving non-linear equations only require evaluating the Jacobian matrix at the current solution, $\mathbf{u^k}$, the numerically approximated Jacobian matrix well serves for this purpose. For example, trust-region method [35] solves non-linear equations by approximating $\mathbf{F}$ with a quadratic function in a trusted region centred by the current solution point, $\mathbf{u^k}$. The quadratic function is formulated based on approximated Jacobian and Hessian matrices, which is applied to locate the next point, $\mathbf{u^{k+1}}$, leading to the regional minimum. The iteration goes on by updating approximated Jacobian and Hessian matrices at the new solution point until the convergence criterion is satisfied. The non-linear system solver provided by MATLAB, fsolve, is based on the trust-region algorithm and applies the finite difference method to approximate the Jacobian matrix if it is not provided. It should be noted that implementing the finite-difference approximation in each iteration is very computationally intensive, especially for large problems. Therefore, LionPower provides the numeric Jacobian matrix to fsolve, which significantly reduces the computational time.

3.2. Jacobian-free Method

Owing to the complexity involved in deriving the Jacobian matrix, the Jacobian-free methods are also widely used to solve non-linear equations. Among them, the Jacobian-Free Newton Krylov (JFNK) method [30, 36–39] is well known for solving large-scale problems, which replaces fsolve in LionPower when a fine mesh is used in the battery simulation or integrated into a hybrid electric powertrain model, as the derivation of the analytic Jacobian matrix becomes inefficient in MATLAB with an increased number of state variables.

The basic idea of JFNK is to minimise $\left\|\mathbf{J}\delta\mathbf{u^k} + \mathbf{F}\left(\mathbf{u^k}\right)\right\|$ by constructing $\delta\mathbf{u^k}$ from a subspace spanned by the Krylov vectors. Given an initial guess, $\delta\mathbf{u_0^k}$, as the Newton correction, the initial linear residual, $\mathbf{r_0^k}$, is computed by

$$\mathbf{r_0^k} = -\mathbf{F}\left(\mathbf{u^k}\right) - \mathbf{J}\delta\mathbf{u_0^k} \tag{62}$$

The Krylov subspace is formulated by the products of the Jacobian matrix and the initial residual, which can be written as

17

$$\mathbf{K_m} = \text{span}\left(\mathbf{r}_0^{\mathbf{k}}, \ \mathbf{Jr}_0^{\mathbf{k}}, \ \mathbf{J}^2\mathbf{r}_0^{\mathbf{k}}, \ ..., \ \mathbf{J}^{m-1}\mathbf{r}_0^{\mathbf{k}}\right) \tag{63}$$

where m is the dimension of the Krylov subspace and is smaller than the number of unknowns. Based on the Krylov subspace, Newton correction is constructed by

$$\delta\mathbf{u}_m^{\mathbf{k}} = \delta\mathbf{u}_0^{\mathbf{k}} + \sum_{i=0}^{m-1} \beta_i \mathbf{J}^i \mathbf{r}_0^{\mathbf{k}} \tag{64}$$

where β_i are coefficients that minimise the norm. The Generalized Minimal RESidual function (GMRES) is applied to compute β_i in MATLAB.

The construction of the Newton correction in Eq. 64 suggests that the Jacobian matrix is not required as long as the matrix-vector products can be approximated. The finite difference method is used in the approximation, which is given by

$$\mathbf{Jv} = \frac{\mathbf{F}\left(\mathbf{u}^{\mathbf{k}} + \zeta\mathbf{v}\right) - \mathbf{F}\left(\mathbf{u}^{\mathbf{k}}\right)}{\zeta} \tag{65}$$

where ζ is the perturbation parameter. The approach proposed by Brown and Saad [40] is applied in this work to estimate ζ, which is expressed in Eq. 66.

$$\zeta = \frac{b}{\|\mathbf{v}\|_2} \max\left[|\mathbf{u}^{\mathbf{kT}}\mathbf{v}|, \text{typ } \mathbf{u}^{\mathbf{k}}|\mathbf{v}|\right] \text{sign}\left(\mathbf{u}^{\mathbf{kT}}\mathbf{v}\right) \tag{66}$$

where b denotes the square root of relative error in computing $\mathbf{F}\left(\mathbf{u}^{\mathbf{k}}\right)$, and typ $\mathbf{u}^{\mathbf{k}}$ is the typical size of $\mathbf{u}^{\mathbf{k}}$.

To reduce the number of GMRES iterations, which is indicated by m in Eq. 64, the JFNK method is preconditioned by a frozen, representative Jacobian matrix, $\mathbf{J}^*$, as shown in Eq. 67.

$$\mathbf{J}^{*-1}\mathbf{J}\left(\mathbf{u}^{\mathbf{k}}\right)\delta\mathbf{u}^{\mathbf{k}} = -\mathbf{J}^{*-1}\mathbf{F}\left(\mathbf{u}^{\mathbf{k}}\right) \tag{67}$$

The term $\mathbf{J}^{*-1}\mathbf{J}\left(\mathbf{u}^{\mathbf{k}}\right)$ is close to an identity matrix, which substantially simplifies the GMRES iteration. In this work, the frozen Jacobian matrix is derived based on the finite difference method in Eq. 61, and its inverse is handled by the

18

incomplete LU factorization. Note that the frozen Jacobian matrix might be updated during the modelling when the simulation struggles to converge.

4. Model Performance

To highlight the features of LionPower, it is compared with two existing battery models: LIONSIMBA and GT-AutoLion. These three models mainly apply the finite volume method to discretise the governing equations. Special considerations are given to the spherical diffusion equation by LionPower and LIONSIMBA. The control volume method and a high-order finite difference method are used to discretise the spherical diffusion equation by LionPower and LIONSIMBA, respectively. In terms of the numerical solver, LIONSIMBA implemented the IDA package from SUNDIALS [33] to solve the spatially discretised equations, while LionPower applied both the Jacobian-based and the Jacobian-free methods to solve the temporally and spatially discretised non-linear equations. As a commercial model, details of GT-AutoLion are not available.

The performance of LionPower, LIONSIMBA, and GT-AutoLion is first compared by modelling the discharge processes of a 12Ah LiFePO$_4$/graphite battery with constant rates at an environmental temperature of 283.15 K. The electrochemical data are listed in Table 1 with expressions for the conductivity of the electrolyte, the open-circuit potentials and the entropy changes of both anode and cathode shown in Eqs. 68 to 72. In addition, both LionPower and LIONSIMBA are evaluated by a set of Urban Dynamometer Driving Schedule (UDDS) data which well emulates electrified powertrain systems. All the simulations in this work were carried out on a Windows laptop with Intel i7-7820HK processor and 16.0 GB memory.

$$
\begin{aligned}
\kappa = 1.2544 \times 10^{-3} c_e (-8.2488 + 0.053248T - 2.9871 \times 10^{-5}T^2 + 2.6235 \times 10^{-4}c_e - 9.3063 \times 10^{-6}c_e T + 8.069 \\
\times 10^{-9}c_e T^2 + 2.2002 \times 10^{-7}c_e^2 - 1.765 \times 10^{-10}c_e^2 T)^2
\end{aligned}
\tag{68}
$$

$$
\begin{aligned}
U_n = 0.6379 + 0.5416 exp\left(-305.5309\theta_n\right) + 0.044 tanh\left(\frac{0.1958 - \theta_n}{0.1088}\right) \\
- 0.1978 tanh\left(\frac{\theta_n - 1.0571}{0.0854}\right) - 0.6875 tanh\left(\frac{\theta_n - 0.0117}{0.0529}\right) - 0.0175 tanh\left(\frac{\theta_n - 0.5692}{0.0875}\right)
\end{aligned}
\tag{69}
$$

Table 1: Parameters used in the electrochemical-thermal modelling.

Parameter	Negative current collector	Negative electrode (anode)	Separator	Positive electrode (cathode)	Positive current collector	Ref.
L (m)	9×10^{-6}	5.9×10^{-5}	2×10^{-5}	9.2×10^{-5}	1.6×10^{-5}	[41]
R_s (m)	-	1.475×10^{-5}	-	1.15×10^{-6}	-	[41]
ϵ_s	-	0.56	-	0.435	-	[42]
ϵ_e	-	0.3	0.4	0.28	-	[41]
$c_{s,max}$ (mol/m^3)	-	31370	-	22806	-	[42]
c_{e0} (mol/m^3)	-	1000	1000	1000	-	[41]
D_s (m^2/s)	-	3.9×10^{-14}	-	1.25×10^{-15}	-	[41]
E_{aD} (J/mol)	-	3.5×10^4	-	3.5×10^4	-	[42, 43]
D_e (m^2/s)	-	\multicolumn{3}{c}{$10^{-8.43-54.0/(T-229-0.005c_e)-2.2\times10^{-4}c_e}$}	-	[44]
σ (S/m)	-	100	-	0.5	-	[45, 46]
κ (S/m)	-		Eq. 68		-	[47]
k (mol^a/m^b/s)	-	3×10^{-11}	-	1.4×10^{-12}	-	[48, 49]
E_{ak} (J/mol)	-	2×10^4	-	3×10^4	-	[42, 50]
α	-	0.5	-	0.5	-	assumed
γ	-	1.5	1.5	1.5	-	assumed
ρ (kg/m^3)	8900	2500	1200	1500	2700	[51]
c_p (J/mol/K)	385	800	800	800	897	[51]
λ (W/m/K)	401	1.04	1	1.48	237	[52]
U (V)	-	Eq. 69	-	Eq. 70	-	[42]
$\partial U/\partial T$ (V/K)	-	Eq. 71	-	Eq. 72	-	[46]
t_+			0.363			[45]
h (W/m^2/K)			20			assumed
T_{ref} (K)			298.15			-
T_{amb} (K)			283.15			assumed
R (J/mol/K)			8.314			-
F (C/mol)			96487			-

$$U_p = 3.4323 - 0.4828 exp\left[-80.2493\left(1 - \theta_p\right)^{1.3198}\right] - 3.2474 \times 10^{-6} exp\left[20.2645\left(1-\theta_p\right)^{3.8003}\right] + 3.2482 \times 10^{-6} exp\left[20.2646\left(1-\theta_p\right)^{3.7995}\right] \tag{70}$$

$$\frac{\partial U_n}{\partial T} = 344.1347148 \frac{exp\left(-32.9633287\theta_n + 8.316711484\right)}{1 + 749.0756003 exp\left(-34.79099646\theta_n + 8.887143624\right)} - 0.8520278805\theta_n + 0.362299229\theta_n^2 + 0.2698001697 \tag{71}$$

$$\frac{\partial U_p}{\partial T} = -0.35376\theta_p^8 + 1.3902\theta_p^7 - 2.2585\theta_p^6 + 1.9635\theta_p^5 - 0.98716\theta_p^4$$
$$+ 0.28857\theta_p^3 - 0.046272\theta_p^2 + 0.0032158\theta_p - 1.9186 \times 10^{-5} \tag{72}$$

4.1. Comparison among LionPower, LIONSIMBA and GT-AutoLion under constant rates

The discharge data of the 12Ah LiFePO$_4$/graphite battery with 1C and 2C rates at 283.15 K were modelled by LionPower and LIONSIMBA, as shown in Fig. 4(a). It is apparent that the overall trends of the discharge are well captured by both LionPower and LIONSIMBA, but noticeable discrepancies exist between the measurements and the simulations. The discrepancies mainly come from the following aspects. First, the chemical, physical, and electrochemical parameters used in the simulations are collected from the literature, which may not be fully representative of the battery in our experiments. A similar level of discrepancy can be found in [53] which used the same set of parameters to model LiFePO$_4$ batteries. In addition, the standard electrochemical-thermal battery model may not be sufficient to model modern batteries, even with carefully measured parameters [54]. Although Chen et al. [54] developed experimental techniques to parametrize a battery model, tuning is still required to match the measured voltage profile. Note that adjustments were only made to the stoichiometry ranges of anode and cathode, together with the cross-section area in this work, which are given by $\theta_n \in [0.012, 0.9656]$, $\theta_p \in [0.06, 0.92]$, and 0.5854 m^2. With the same set of the battery parameters, the modelling results from LionPower agree very well with those from LIONSIMBA, which validates our model. The JFNK solver was applied to model these measurements as well, which, not surprisingly, produces almost identical modelling results to the default Jacobian-based solver in LionPower, as shown in Fig. 4(a). To compare the thermal modelling of LionPower and LIONSIMBA, the initial temperature of the battery is set to 298.15 K and the thermal stabilisation to the environmental temperature 283.15 K is modelled. Fig. 4(b) compares the modelled average temperatures from LionPower and LIONSIMBA at 1C discharge rate and close agreement is observed between these two models.

Despite the similarity between the modelling results from LionPower and LIONSIMBA, the discrepancy is noticeable towards the end of the simulation in Fig. 4(a). To analyse the discrepancy at a representative time of 3500 s which is marked by a vertical dashed grey line in Fig. 4(a), the modelled mole concentrations of lithium-ion in the solid phase in both anode and cathode with 1C discharge rate from LionPower and LIONSIMBA are compared in Fig.

21
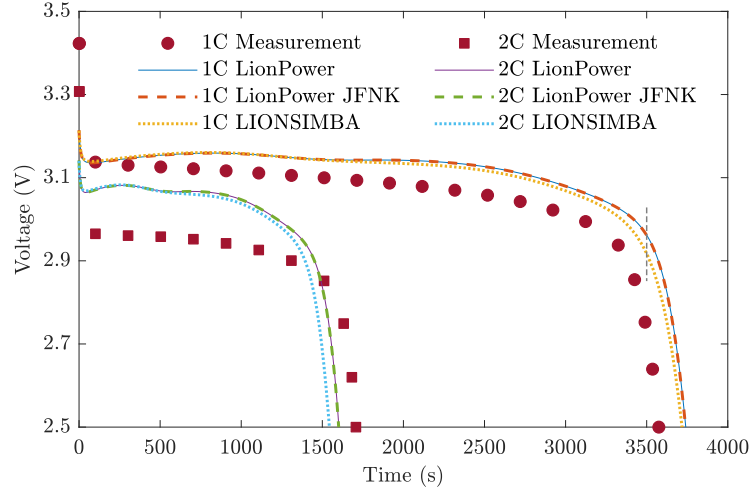
(a)
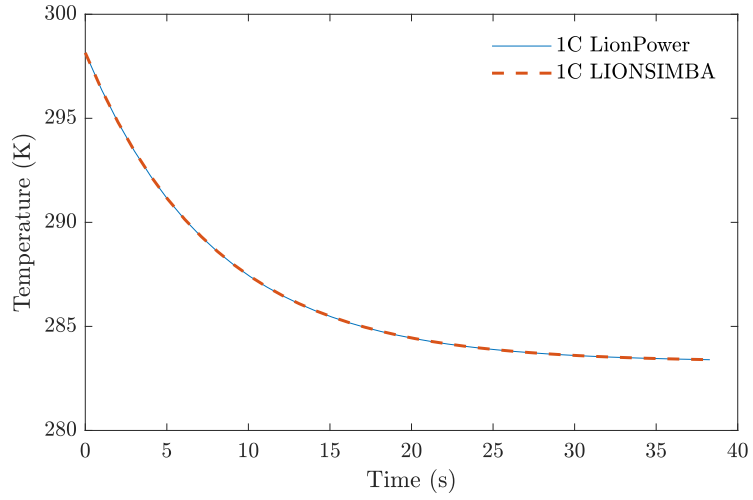
(b)

Figure 4: The measured voltages of the 12 Ah LiFePO$_4$ battery with 1C and 2C discharge rates at 283.15 K and the modelled (a) voltages and (b) temperatures by LionPower and LIONSIMBA.

5. It is clear that more lithium-ions migrate from anode to cathode in LionPower than in LIONSIMBA, which leads to the difference in Fig. 4(a) based on the open-circuit potentials expressed in Eqs. 69 and 70. The different distributions of lithium-ions in these two models result from the modelling of spherical diffusion. A further comparison is made by evaluating the property of mass conservation of the spherical diffusion in the solid phase in these two models. Mass is conserved when the change of number of moles inside a particle corresponds to the net molar flux via the boundaries of the particle in the absence of source terms, which is expressed as

$$\hat{n} = \frac{\sum V_i \frac{\partial c_i}{\partial t} + jR_s^2}{\|j\| R_s^2} \tag{73}$$

where V_i is the volume of shell i (divided by 4π) in Fig. 2, and $\hat{n} = 0$ when the mass is conserved. The normalisation by $\|j\| R_s^2$ helps to reveal the relative error. Eq. 73 is evaluated by the modelling results from LionPower and LIONSIMBA, and the computed values are compared in Fig. 6. It is apparent that the computed mass imbalances from LionPower are negligible compared with those from LIONSIMBA in all particles from both anode and cathode. Since LionPower applies the conservative scheme to discretise the spherical diffusion equation, its $\hat{n}$ on both anode and cathode are approximately five orders of magnitude smaller than those from LIONSIMBA which uses a high-order finite difference scheme. Therefore, it is preferable to use LionPower in fundamental studies where conserved spherical diffusion is critical.

The computational time for the 1C discharge modelling of the LiFePO$_4$/graphite battery by LionPower and LIONSIMBA is compared in Table 2. LionPower is slower than LIONSIMBA under constant rates, and the former with the Jacobian-based solver takes 138 s to complete the modelling of 1C discharge, while the latter only requires 17 s. With the implementation of the JFNK solver in LionPower, the simulation time then increases to 414 s. The difference in computational time is caused by different numerical approaches used in solving the discretised equations. LIONSIMBA applies the IDA solver from SUNDIALS, which solves the equations with adaptive time step under constant charge/discharge rates. In comparison, LionPower solves the equations at all sampling points from the measurements. With constant charge/discharge rates, integrators for DAEs are much more efficient than numerical solvers for discretised algebraic equations. However, charge/discharge processes with constant rates are not representative of
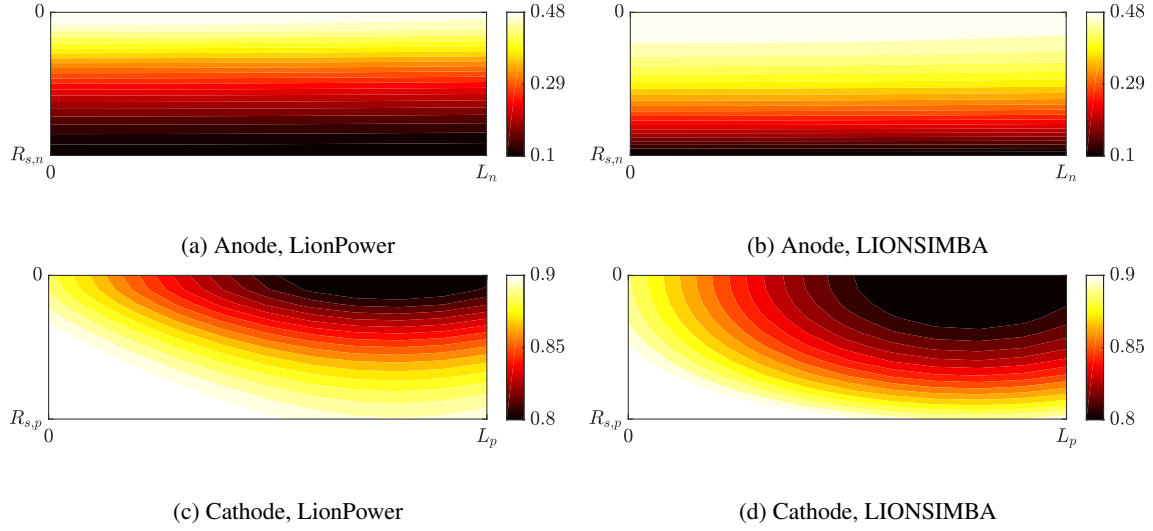
23

(a) Anode, LionPower

(b) Anode, LIONSIMBA

(c) Cathode, LionPower

(d) Cathode, LIONSIMBA

Figure 5: Contours of mole concentration of Li^+ in the solid phase in both anode and cathode of the 12 Ah $LiFePO_4$ battery with 1C discharge rate at 3500 s modelled by LionPower and LIONSIMBA.
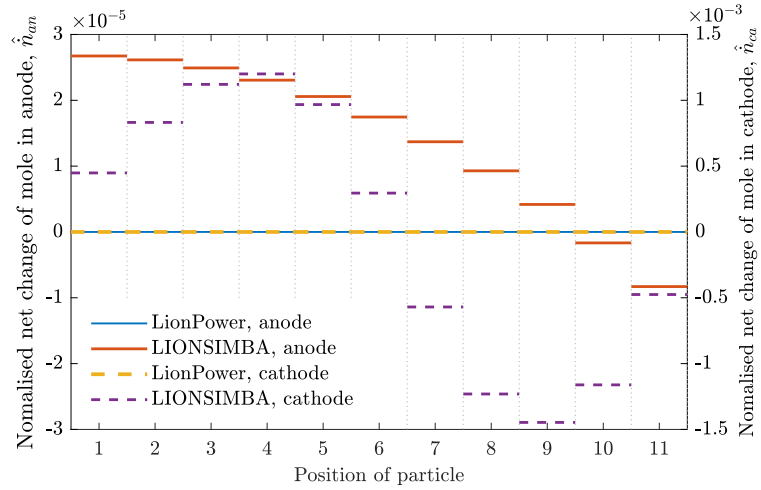

Figure 6: The normalised net change of mole in particles of anode and cathode in the 12 Ah $LiFePO_4$ battery with 1C discharge rate at 3500 s modelled by LionPower and LIONSIMBA. Note that the orders of magnitude of $\hat{n}_{an}$ and $\hat{n}_{ca}$ from LionPower are 10^{-10} and 10^{-8}, respectively, which cannot be viewed on the current scale.

24

electrified powertrain systems and LionPower is mainly developed for those applications, which will be discussed in a later section.

Table 2: The computational time required by LionPower and LIONSIMBA to model the 1C discharge of the 12Ah LiFePO$_4$/graphite battery.

Model	Time (s)
LionPower, fsolve	138
LionPower, JFNK	414
LIONSIMBA	17

As a recently released commercial model by Gamma Technologies, GT-AutoLion has a comprehensive electrochemical materials database and can seamlessly be integrated into the modelling of powertrain systems. Meanwhile, new features are constantly added to GT-AutoLion to improve its performance and usability. In the latest official release, the stoichiometry ranges of anode and cathode are automatically determined based on the detailed battery geometry and specifications, and it is currently not possible to directly change these ranges without carefully balancing the capacities of both anode and cathode. In this case, LionPower and LIONSIMBA were set to apply the auto-determined stoichiometry ranges from GT-AutoLion for the ease of comparison, which are given by $\theta_n \in [0.035, 0.844]$ and $\theta_p \in [0.059, 0.936]$. Furthermore, the lower limit of battery voltage is adjusted to 2.8 V, as defined by the stoichiometry ranges.

The modelled battery voltages from LionPower, LIONSIMBA, and GT-AutoLion at 1C discharge rate are compared in Fig. 7. Note that the relatively short discharge time is due to the increased lower cut-off voltage from 2.5 V to 2.8 V imposed by GT-AutoLion. It is clear that the one from GT-Autolion is higher than those from LionPower and LIONSIMBA at the early discharge stage. Since the detailed mole concentrations of lithium-ion in the solid phase are not directly available in the output of the current GT-AutoLion, it is difficult to carry out a comprehensive analysis of the modelling results. In this case, the surface SOC of cathode particles is compared in Fig. 8 to illustrate the difference among these three models. Initially, the evolution of surface SOC is similar among the three, but a larger spatial gradient is observed in GT-AutoLion at the later stage, while the results from LionPower and LIONSIMBA are similar. The large spatial gradient seems to suggest lithium-ions tend to aggregate in the particles adjacent to the separator and migrate slowly to those that are further away, which might be caused by different numerical schemes
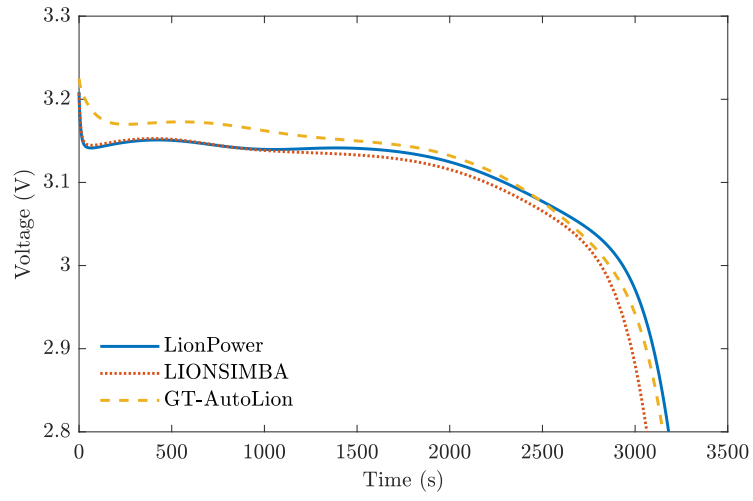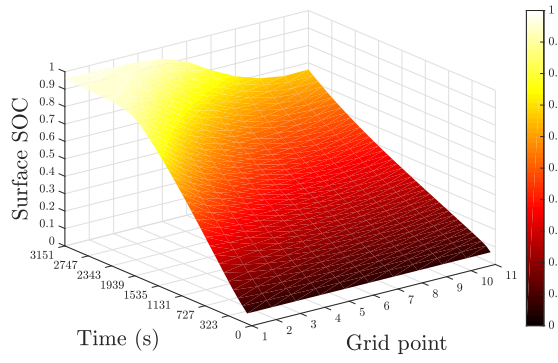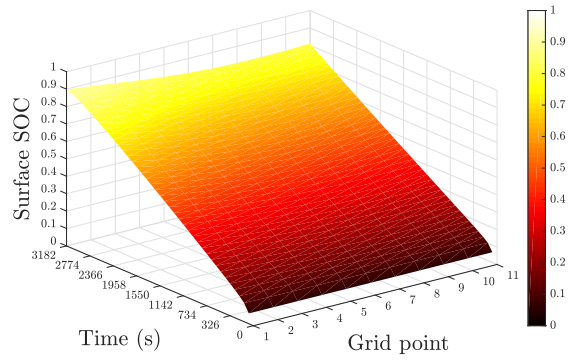
used in GT-AutoLion.

Figure 7: The comparison of modelling results of 1C discharge by LionPower, LIONSIMBA, and GT-AutoLion with the lower cut-off voltage of 2.8 V imposed by GT-AutoLion. The stoichiometry ranges of anode and cathode used in the modelling are automatically determined by GT-AutoLion.

(a) Surface stoichiometry, GT-AutoLion

(b) Surface stoichiometry, LionPower
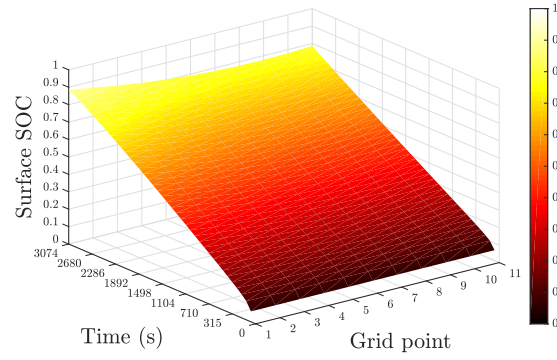
(c) Surface stoichiometry, LIONSIMBA

Figure 8: Spatial and temporal evolution of the surface SOC of the cathode modelled by GT-AutoLion, LionPower, and LIONSIMBA.

4.2. Comparison between LionPower and LIONSIMBA under UDDS

As mentioned previously, LionPower is mainly developed for the studies of electrified powertrain systems and it is of interest to evaluate its performance using the UDDS data as shown in Fig. 9(a). For comparison, LIONSIMBA was also applied to model the UDDS data based on its example script for the continuous custom current profile.
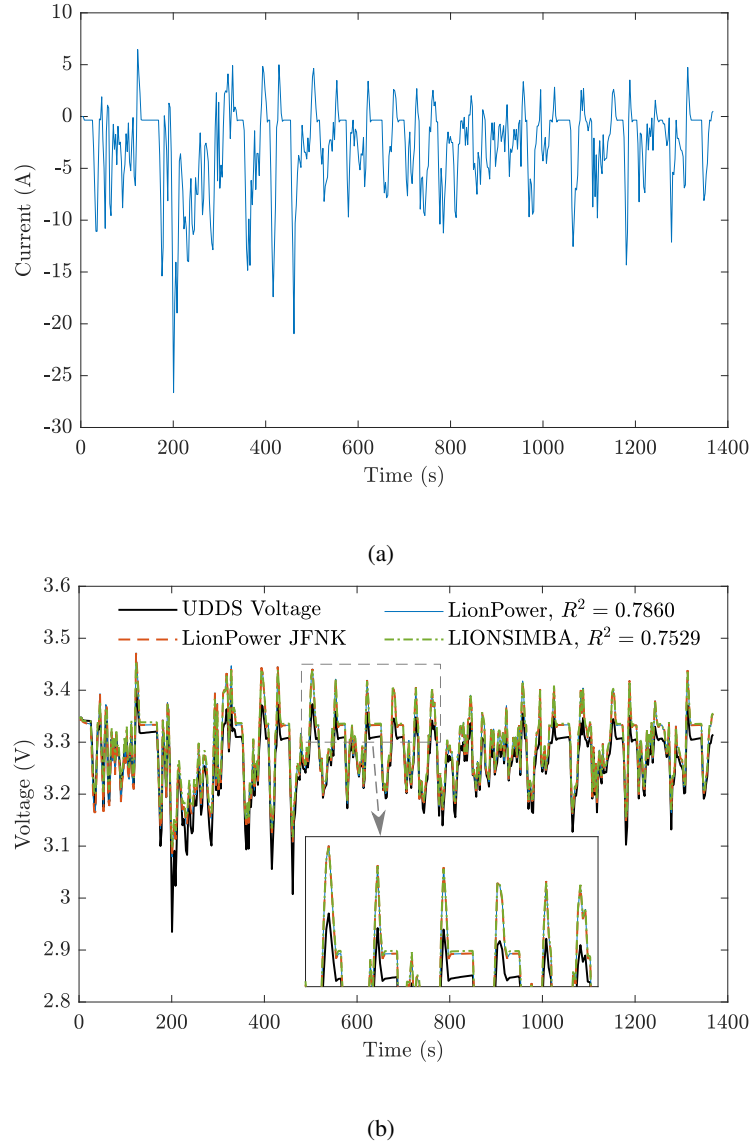

(a)

(b)

Figure 9: (a) The measured fast-changing time-dependent charge/discharge currents and (b) corresponding voltages from both measurements and simulations under UDDS.

The measured UDDS voltage, together with the modelling results from LionPower and LIONSIMBA are compared in Fig. 9(b). Both models well predict the oscillations of UDDS voltages but with consistent discrepancies

28

which can be viewed clearly from the zoomed-in plot. This discrepancies are in line with the comparisons shown in Fig. 4(a). LionPower performs slightly better than LIONSIMBA in modelling the UDDS data, as indicated by the R^2 values in Fig. 9(b). As expected, the modelled voltage by the JFNK method overlaps with the result from the Jacobian-based solver in LionPower.

Although the modelled voltages from both models are similar, LIONSIMBA is much slower than LionPower in modelling the UDDS data, as shown in Table 3. It takes LionPower 45 s to complete the simulation with the Jacobian-based solver and 84 s with the JFNK method, but the computational time increases by two orders of magnitude to 6218 s when using LIONSIMBA. Note that LIONSIMBA requires a Jacobian matrix with a representative current density as an input for the following simulation of the custom current profile, and the modelling may fail to converge if the selected representative current density is not desirable. By trial and error, LIONSIMBA converges with 8 A/m^2 as the representative current density. Although preconditioned Krylov iterative linear solvers are available in the IDA package from SUNDIALS, they are not currently implemented in LIONSIMBA to solve the problems with continuous custom current profiles. It is apparent that the present LIONSIMBA is not suitable for modelling fast-changing and time-dependent charge/discharge current profiles. In comparison, LionPower allows direct control of the time step and converges quickly at each time point, and is thus preferable for the modelling of powertrain systems.

Table 3: The computational time required by LionPower and LIONSIMBA to model the UDDS data.

Model	Time (s)
LionPower, fsolve	45
LionPower, JFNK	84
LIONSIMBA	6218

Based on the comparison of the computational time, the Jacobian-based solver in LionPower, which relies on the Symbolic Math Toolbox and fsolve in MATLAB, outperforms the JFNK solver. However, the Jacobian-based solver becomes less efficient with an increased number of state variables which could come from either a very fine mesh in the battery modelling or a hybrid electric powertrain model comprising a kinetic sub-model of engine combustion [34]. In these situations, it is desirable to apply the JFNK method which is designed to solve large-scale problems. Since LionPower is developed for the research of electrified powertrain systems, the inclusion of the JFNK solver is

29

critical for future modelling works at the system-level.

5. Conclusions

This work applied the mass-conserved control volume method to discretise the spherical diffusion equation and the finite volume method for the rest differential equations in the electrochemical-thermal battery model. Both Jacobian-based and Jacobian-free solvers were implemented in this work to solve the discretised, non-linear equations in an efficient manner, especially for the modelling of real-world driving cycles from electrified powertrains.

Comparisons were first made among LionPower, LIONSIMBA, and GT-AutoLion in modelling the discharge processes with constant rates. Both LionPower and LIONSIMBA reasonably capture the measured battery voltages. Furthermore, it was found that LionPower achieves mass conservation of the spherical diffusion with the control volume method, the mass imbalances of which are five orders of magnitude smaller than those of LIONSIMBA. Under constant rates, the computational time taken by LIONSIMBA is shorter than that with LionPower, as the former applies adaptive time step in the modelling, while the latter sticks to a fixed time step. Compared with LionPower and LIONSIMBA, the commercial model GT-AutoLion behaves slightly differently in the temporal and spatial evolution of surface SOC, which is probably caused by different numerical schemes used in GT-AutoLion. Further comparison between LionPower and LIONSIMBA was carried out based on the UDDS data representing real-world driving cycles. The overall trend of the UDDS data is well reproduced by both models but with consistent discrepancies which were also observed in the early simulations with constant discharge rates. The UDDS data challenge the performance of LIONSIMBA which takes two orders of magnitude longer computational time to complete the modelling compared with LionPower, indicating LionPower is more preferable to be applied in modelling real-world driving cycles of electrified powertrains.

As a battery model completely based on MATLAB, it is convenient to integrate LionPower as a sub-model in existing powertrain models which are commonly built in the MATLAB/Simulink modelling environment. The Jacobian-based method is mainly applied in the modelling of pure electric powertrains with a not too fine mesh, since MATLAB is not very efficient in formulating a large Jacobian matrix. In comparison, the Jacobian-free method can handle a fine mesh and is applicable in modelling hybrid electric powertrain systems comprising sub-models for both batteries

and engines. Future developments of LionPower may include applying MATLAB's integrator ode15s when modelling charge/discharge processes with constant rates, coupling a comprehensive, physics-based degradation model, and formulating efficient, function-based preconditioners to parallelise the JFNK method.

List of Symbols

a_s Specific interfacial surface area (1/m)

c_e Mole concentration of lithium-ion in the liquid phase (mol/m^3)

c_{e0} Initial mole concentration of lithium-ion in the liquid phase (mol/m^3)

c_p Specific heat capacity (J/mol/K)

c_s Mole concentration of lithium-ion in the solid phase (mol/m^3)

$c_{s,max}$ Maximum mole concentration of lithium-ion in the solid phase (mol/m^3)

$c_{s,surf}$ Mole concentration of lithium-ion on the surface of particle (mol/m^3)

D_e Diffusion coefficient of lithium-ion in the liquid phase (m^2/s)

D_s Diffusion coefficient of lithium-ion in the solid phase (m^2/s)

E_{aD} Activation energy for solid phase diffusion (J/mol)

E_{ak} Activation energy of electrochemical reaction (J/mol)

F Faraday constant (C/mol)

$f_\pm$ Average molar activity coefficient

h Heat transfer coefficient (W/m^2/K)

I Charge/discharge current flux (A/m^2)

j Molar flux (mol/m^2/s)

$\bar{j}$ Averaged molar flux (mol/m^2/s)

k Rate constant (mol^a/m^b/s) (a and b depends on the reaction)

L Length in the x direction (m)

q Volumetric heat generation rate (W/m^3)

R Gas constant (J/mol/K)

R_s Radius of solid particle (m)

r Radial distance of particle (m)

T Temperature (K)

T_{ref} Reference temperature (K)

t Time (s)

t_+ Transference number

U Open-circuit potential (V)

x 1D spatial distance (m)

Greek Symbols

α Transfer coefficient

γ Bruggeman exponent

ϵ_e Volume fraction in the liquid phase

ϵ_s Volume fraction in the solid phase

η Overpotential (V)

θ SOC of electrode

κ Conductivity of the electrolyte (S/m)

λ Thermal conductivity (W/m/K)

ρ Density (kg/m^3)

σ Conductivity of the electrode (S/m)

ϕ_e Potential in the liquid phase (V)

ϕ_s Potential in the solid phase (V)

Subscripts

n Negative electrode

ncc Negative current collector

p Positive electrode

pcc Positive current collector

sep Separator

Acknowledgements

References

[1] J. H. Williams, A. DeBenedictis, R. Ghanadan, A. Mahone, J. Moore, W. R. Morrow, S. Price, M. S. Torn, The Technology Path to Deep Greenhouse Gas Emissions Cuts by 2050: The Pivotal Role of Electricity, Science 335 (6064) (2012) 53–59. doi:10.1126/science.1208365.

[2] S. Chu, A. Majumdar, Opportunities and challenges for a sustainable energy future, Nature 488 (7411). doi:10.1038/nature11475.

[3] M. Tran, D. Banister, J. D. K. Bishop, M. D. McCulloch, Realizing the electric-vehicle revolution, Nature Climate Change 2 (5). doi:10.1038/nclimate1429.

[4] N. Melton, J. Axsen, D. Sperling, Moving beyond alternative fuel hype to decarbonize transportation, Nature Energy 1 (3). doi:10.1038/nenergy.2016.13.

[5] European Commission, Electrification of the Transport System, Tech. rep. (2017).

[6] M. M.Thackeray, C. Wolverton, E. D.Isaacs, Electrical energy storage for transportationapproaching the limits of, and going beyond, lithium-ion batteries, Energy & Environmental Science 5 (7) (2012) 7854–7863. doi:10.1039/C2EE21892E.

[7] Global EV Outlook 2020 Analysis.

[8] X. He, S. Zhang, Y. Wu, T. J. Wallington, X. Lu, M. A. Tamor, M. B. McElroy, K. M. Zhang, C. P. Nielsen, J. Hao, Economic and Climate Benefits of Electric Vehicles in China, the United States, and Germany, Environmental Science & Technology 53 (18) (2019) 11013–11022. doi:10.1021/acs.est.9b00531.

[9] N. A. Chaturvedi, R. Klein, J. Christensen, J. Ahmed, A. Kojic, Algorithms for Advanced Battery-Management Systems, IEEE Control Systems Magazine 30 (3) (2010) 49–68. doi:10.1109/MCS.2010.936293.

[10] M. Doyle, T. F. Fuller, J. Newman, Modeling of Galvanostatic Charge and Discharge of the Lithium/Polymer/Insertion Cell, Journal of The Electrochemical Society 140 (6) (1993) 1526–1533. doi:10.1149/1.2221597.

[11] T. F. Fuller, M. Doyle, J. Newman, Simulation and Optimization of the Dual Lithium Ion Insertion Cell, Journal of The Electrochemical Society 141 (1) (1994) 1. doi:10.1149/1.2054684.

[12] C. R. Pals, J. Newman, Thermal Modeling of the Lithium/Polymer Battery I . Discharge Behavior of a Single Cell, Journal of The Electrochemical Society 142 (10) (1995) 3274–3281. doi:10.1149/1.2049974.

[13] K. E. Thomas, J. Newman, R. M. Darling, Mathematical Modeling of Lithium Batteries, in: W. A. van Schalkwijk, B. Scrosati (Eds.), Advances in Lithium-Ion Batteries, Springer US, Boston, MA, 2002, pp. 345–392. doi:10.1007/0-306-47508-1˙13.

[14] A. N. Ford Versypt, R. D. Braatz, Analysis of finite difference discretization schemes for diffusion in spheres with variable diffusivity, Computers & Chemical Engineering 71 (2014) 241–252. doi:10.1016/j.compchemeng.2014.05.022.

[15] M. Torchio, L. Magni, R. B. Gopaluni, R. D. Braatz, D. M. Raimondo, LIONSIMBA: A Matlab Framework Based on a Finite Volume Model Suitable for Li-Ion Battery Design, Simulation, and Control, Journal of The Electrochemical Society 163 (7) (2016) A1192. doi:10.1149/2.0291607jes.

[16] A. Liu, P. Liu, S. Liu, Gas diffusion coefficient estimation of coal: A dimensionless numerical method and its experimental validation, International Journal of Heat and Mass Transfer 162 (2020) 120336. doi:10.1016/j.ijheatmasstransfer.2020.120336.

[17] Z. Chen, Effects of radiation and compression on propagating spherical flames of methane/air mixtures near the lean flammability limit, Combustion and Flame 157 (12) (2010) 2267–2276. doi:10.1016/j.combustflame.2010.07.010.

[18] W. Han, Z. Chen, Effects of Soret diffusion on spherical flame initiation and propagation, International Journal of Heat and Mass Transfer 82 (2015) 309–315. doi:10.1016/j.ijheatmasstransfer.2014.11.062.

[19] W. Han, Z. Chen, Effects of finite-rate droplet evaporation on the extinction of spherical burner-stabilized diffusion flames, International Journal of Heat and Mass Transfer 99 (2016) 691–701. doi:10.1016/j.ijheatmasstransfer.2016.04.003.

[20] G. G. Botte, J. A. Ritter, R. E. White, Comparison of finite difference and control volume methods for solving differential equations, Computers & Chemical Engineering 24 (12) (2000) 2633–2654. doi:10.1016/S0098-1354(00)00619-0.

[21] Y. Zeng, P. Albertus, R. Klein, N. Chaturvedi, A. Kojic, M. Z. Bazant, J. Christensen, Efficient Conservative Numerical Schemes for 1D Nonlinear Spherical Diffusion Equations with Applications in Battery Modeling, Journal of The Electrochemical Society 160 (9). doi:10.1149/2.102309jes.

[22] K. Chayambuka, G. Mulder, D. L. Danilov, P. H. L. Notten, A Hybrid Backward Euler Control Volume Method to Solve the Concentration-Dependent Solid-State Diffusion Problem in Battery Modeling, Journal of Applied Mathematics and Physics 8 (6). doi:10.4236/jamp.2020.86083.

[23] P. Peng, F. Jiang, Thermal safety of lithium-ion batteries with various cathode materials: A numerical study, International Journal of Heat and Mass Transfer 103 (2016) 1008–1016. doi:10.1016/j.ijheatmasstransfer.2016.07.088.

[24] T. Dong, P. Peng, F. Jiang, Numerical modeling and analysis of the thermal behavior of NCM lithium-ion batteries subjected to very high C-rate discharge/charge operations, International Journal of Heat and Mass Transfer 117 (2018) 261–272. doi:10.1016/j.ijheatmasstransfer.2017.10.024.

[25] C. Qi, Y. Zhu, F. Gao, K. Yang, Q. Jiao, Mathematical model for thermal behavior of lithium ion battery pack under overcharge, International Journal of Heat and Mass Transfer 124 (2018) 552–563. doi:10.1016/j.ijheatmasstransfer.2018.03.100.

[26] M. Xu, R. Wang, P. Zhao, X. Wang, Fast charging optimization for lithium-ion batteries based on dynamic programming algorithm and

electrochemical-thermal-capacity fade coupled model, Journal of Power Sources 438 (2019) 227015. doi:10.1016/j.jpowsour.2019.227015.

[27] W. Mei, H. Li, C. Zhao, J. Sun, Q. Wang, Numerical study on thermal characteristics comparison between charge and discharge process for lithium ion battery, International Journal of Heat and Mass Transfer 162 (2020) 120319. doi:10.1016/j.ijheatmasstransfer.2020.120319.

[28] W. Mei, Q. Duan, C. Zhao, W. Lu, J. Sun, Q. Wang, Three-dimensional layered electrochemical-thermal model for a lithium-ion pouch cell Part II. The effect of units number on the performance under adiabatic condition during the discharge, International Journal of Heat and Mass Transfer 148 (2020) 119082. doi:10.1016/j.ijheatmasstransfer.2019.119082.

[29] P. Nie, S.-W. Zhang, A. Ran, C. Yang, S. Chen, Z. Li, X. Zhang, W. Deng, T. Liu, F. Kang, G. Wei, Full-cycle electrochemical-thermal coupling analysis for commercial lithium-ion batteries, Applied Thermal Engineering 184 (2021) 116258. doi:10.1016/j.applthermaleng.2020.116258.

[30] D. A. Knoll, D. E. Keyes, Jacobian-free NewtonKrylov methods: a survey of approaches and applications, Journal of Computational Physics 193 (2) (2004) 357–397. doi:10.1016/j.jcp.2003.08.010.

[31] R. I. Hickson, S. I. Barry, G. N. Mercer, H. S. Sidhu, Finite difference schemes for multilayer diffusion, Mathematical and Computer Modelling 54 (1) (2011) 210–220. doi:10.1016/j.mcm.2011.02.003.

[32] N. G. March, E. J. Carr, Finite volume schemes for multilayer diffusion, Journal of Computational and Applied Mathematics 345 (2019) 206–223. doi:10.1016/j.cam.2018.06.041.

[33] A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, C. S. Woodward, SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers, ACM Transactions on Mathematical Software 31 (3) (2005) 363–396. doi:10.1145/1089014.1089020.

[34] H. Yuan, K. Giles, S. Zhu, S. Howson, A. Lewis, S. Akehurst, N. Turner, J. Harris, G. Fowler, J. Geddes, Kinetic modelling of combustion in a spark ignition engine with water injection, Fuel 283 (2021) 118814. doi:10.1016/j.fuel.2020.118814.

[35] Y.-x. Yuan, Recent advances in trust region algorithms, Mathematical Programming 151 (1) (2015) 249–281. doi:10.1007/s10107-015-0893-2.

[36] M. A. S. Bhuiyan, M. A. Hossain, J. M. Alam, A computational model of thermal monitoring at a leakage in pipelines, International Journal of Heat and Mass Transfer 92 (2016) 330–338. doi:10.1016/j.ijheatmasstransfer.2015.08.094.

[37] M. Kumar, G. Natarajan, On the role of discrete mass conservation for non-Boussinesq flow simulations in enclosures, International Journal of Heat and Mass Transfer 104 (2017) 1283–1299. doi:10.1016/j.ijheatmasstransfer.2016.09.073.

[38] S. Chawdhary, A. Khosronejad, G. Christodoulou, F. Sotiropoulos, Large eddy simulation of density current on sloping beds, International Journal of Heat and Mass Transfer 120 (2018) 1374–1385. doi:10.1016/j.ijheatmasstransfer.2017.12.063.

[39] W. Qu, C.-M. Fan, Y. Zhang, Analysis of three-dimensional heat conduction in functionally graded materials by using a hybrid numerical method, International Journal of Heat and Mass Transfer 145 (2019) 118771. doi:10.1016/j.ijheatmasstransfer.2019.118771.

[40] P. N. Brown, Y. Saad, Hybrid Krylov Methods for Nonlinear Systems of Equations, SIAM Journal on Scientific and Statistical Computing 11 (3) (1990) 450–481. doi:10.1137/0911026.

[41] F. Jiang, P. Peng, Y. Sun, Thermal analyses of LiFePO4/graphite battery discharge processes, Journal of Power Sources 243 (2013) 181–194. doi:10.1016/j.jpowsour.2013.05.089.

[42] M. Safari, C. Delacourt, Modeling of a Commercial Graphite/LiFePO4 Cell, Journal of The Electrochemical Society 158 (5). doi:10.1149/1.3567007.

[43] O. Y. Egorkina, A. M. Skundin, The effect of temperature on lithium intercalation into carbon materials, Journal of Solid State Electrochemistry 2 (4) (1998) 216–220. doi:10.1007/s100080050091.

[44] L. O. Valen, J. N. Reimers, Transport Properties of LiPF6-Based Li-Ion Battery Electrolytes, Journal of The Electrochemical Society 152 (5). doi:10.1149/1.1872737.

[45] S. Santhanagopalan, Q. Guo, P. Ramadass, R. E. White, Review of models for predicting the cycling performance of lithium ion batteries, Journal of Power Sources 156 (2) (2006) 620–628. doi:10.1016/j.jpowsour.2005.05.070.

[46] R. E. Gerver, J. P. Meyers, Three-Dimensional Modeling of Electrochemical Performance and Heat Generation of Lithium-Ion Batteries in Tabbed Planar Configurations, Journal of The Electrochemical Society 158 (7). doi:10.1149/1.3591799.

[47] W. Wu, X. Xiao, X. Huang, The effect of battery design parameters on heat generation and utilization in a Li-ion cell, Electrochimica Acta 83 (2012) 227–240. doi:10.1016/j.electacta.2012.07.081.

[48] K. Kumaresan, G. Sikha, R. E. White, Thermal Model for a Li-Ion Cell, Journal of the Electrochemical Society 155 (2) (2007) A164. doi:10.1149/1.2817888.

[49] Y. Ye, Y. Shi, A. A. O. Tay, Electro-thermal cycle life model for lithium iron phosphate battery, Journal of Power Sources 217 (2012) 509–518. doi:10.1016/j.jpowsour.2012.06.055.

[50] T. L. Kulova, A. M. Skundin, E. A. Nizhnikovskii, A. V. Fesenko, Temperature effect on the lithium diffusion rate in graphite, Russian Journal of Electrochemistry 42 (3) (2006) 259–262. doi:10.1134/S1023193506030086.

[51] V. Srinivasan, C. Y. Wang, Analysis of Electrochemical and Thermal Behavior of Li-Ion Cells, Journal of The Electrochemical Society 150 (1). doi:10.1149/1.1526512.

[52] C. Lin, K. Chen, F. Sun, P. Tang, H. Zhao, Research on thermo-physical properties identification and thermal analysis of EV Li-ion battery, in: 2009 IEEE Vehicle Power and Propulsion Conference, 2009, pp. 1643–1648. doi:10.1109/VPPC.2009.5289653.

[53] Z. An, L. Jia, L. Wei, C. Dang, Q. Peng, Investigation on lithium-ion battery electrochemical and thermal characteristic based on electrochemical-thermal coupled model, Applied Thermal Engineering 137 (2018) 792–807. doi:10.1016/j.applthermaleng.2018.04.014.

[54] C.-H. Chen, F. B. Planella, K. O'Regan, D. Gastol, W. D. Widanage, E. Kendrick, Development of Experimental Techniques for Parameterization of Multi-scale Lithium-ion Battery Models, Journal of The Electrochemical Society 167 (8) (2020) 080534. doi:10.1149/1945-7111/ab9050.