

Research Papers

A computationally informed realisation algorithm for lithium-ion batteries implemented with LiiBRA.jl

Brady Planden *, Katie Lukow, Paul Henshall, Gordana Collier, Denise Morrey

School of Engineering, Computing, and Mathematics, Oxford Brookes University, Wheatley Campus, Wheatley, Oxford, OX33 1HX, United Kingdom



ARTICLE INFO

Keywords:

Lithium-ion battery
Battery management system
Reduced-electrochemical model
Battery modelling
Julia

ABSTRACT

Real-time battery modelling advancements have quickly become required as the adoption of battery electric vehicles (BEVs) has rapidly increased. In this paper an open-source, improved discrete realisation algorithm, implemented in Julia for the creation and simulation of reduced-order, real-time capable physics-based models is presented. This work reduces the Doyle–Fuller–Newman electrochemical model into continuous-form transfer functions and introduces a computationally informed discrete realisation algorithm (CI-DRA) to generate the reduced-order representation. Further improvements in conventional offline model creation are obtained as well as achieving in-vehicle capable model creation for ARM-based computing architectures. Furthermore, a parametric sensitivity analysis of the presented architecture is completed as well as experimental validation of a worldwide harmonised light vehicle test procedure (WLTP) for an LG Chem. M50 21700 parameterisation. A performance comparison to a MATLAB implementation is completed showcasing a mean computational time improvement of 3.51 times for LiiBRA.jl on x86 hardware. Finally, an ARM-based implementation showcases full system model generation within three minutes for potential in-vehicle updates.

1. Introduction

With the rapidly increasing adoption of battery electric vehicles (BEVs), improvements with in-vehicle battery modelling and control are required to improve safety and driving performance, while ensuring the vehicle battery pack reaches the desired lifetime across a multitude of performance ranges. Providing a viable method for capturing real-time degradation mechanisms coupled with physics-based electrochemical models is a key achievement required for future electric vehicle advancements [1]. In order to achieve this, battery models capable of being deployed onto battery control systems with electrochemical information are required. These in-vehicle battery control systems, better known as battery management systems (BMS), ensure hardware limits are maintained while providing the requested interaction from the operator. These systems accomplish this by ensuring the pack is in a safe state for operation, protecting the individual cells from abuse and reducing the battery pack degradation over the lifetime of operation achieved most frequently through online predictive models that are utilised for hardware limit forecasting, plant-based control structures, and state estimation.

Data-driven models such as an equivalent circuit model (ECM) [2–4] is commonly utilised for this prediction as they provide reasonable performance and have a well-established path for model creation.

These models are numerically deployed onto in-vehicle embedded systems and provide information to the BMS that typically would not be attainable via direct sensing methods. This information is provided at designated non-flexible time intervals to the onboard control strategy with key performance indicators such as state-of-power (SOP), state-of-charge (SOC), and state-of-health (SOH) calculated through onboard measurements. Each of these state variables provides insight into the vehicle's capabilities for future operation. These models can provide a fast, reliable solution; however, the creation requires existing data that encompasses the entire operating range of the cell to ensure a stable response to the predicted operating conditions. Obtaining this data is not only time-consuming, on the order of multiple months to years of test channel time, but also requires expensive test equipment. These models also tend to lack electrochemical generality due to their nature and the model data requirements needed to achieve acceptable performance. As an example, ECMs utilise idealised, theoretical electrical components to represent cell behaviour, whose properties are numerically calibrated such that the model output is consistent with only a few basic measured cell characteristics, such as terminal voltage. As such, generality is not achievable for cell characteristics across varying chemistries, geometries, and operating conditions. Additionally, without observing internal electrochemical states during data

* Corresponding author.

E-mail address: brady.planden@brookes.ac.uk (B. Planden).

<https://doi.org/10.1016/j.est.2022.105637>

Received 14 May 2022; Received in revised form 13 August 2022; Accepted 4 September 2022

Available online 18 September 2022

2352-152X/© 2022 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Nomenclature

n	Negative domain
s	Separator
p	Positive domain
L_k	Domain length, $k \in \{n, s, p\}$
*	Dimensionless operator
$\phi_{s,k}$	Solid potential, $k \in \{n, p\}$
$\phi_{e,k}$	Electrolyte potential, $k \in \{n, s, p\}$
$c_{s,k}$	Solid lithium concentration, $k \in \{n, p\}$
$c_{e,k}$	Electrolyte salt concentration, $k \in \{n, s, p\}$
$i_{e,k}$	Ionic current density, $k \in \{n, s, p\}$
$N_{e,k}$	Electrolyte molar flux, $k \in \{n, s, p\}$
σ_k	Solid conductivity, $k \in \{n, p\}$
κ_e	Electrolyte conductivity
I_{app}	Applied current density
R_s	Particle radius
D_s	Solid diffusivity
D_e	Electrolyte diffusivity
ϵ_k	Electrolyte volume fraction, $k \in \{n, s, p\}$
t^+	Transference number
a_k	Solid surface area density, $k \in \{n, s, p\}$
m_k	Reaction rate, $k \in \{n, p\}$
$U_{k,ref}$	Reference open circuit potential, $k \in \{n, p\}$
R_{ct}	Charge Transfer Resistance, $k \in \{n, p\}$
R_{film}	Film Resistance, $k \in \{n, p\}$
x	Cell coordinate location across cell
z	Unitless dimension across electrode
β	Jacobsen-West transfer function parameter
$S_{e,m}$	Number of spacial electrolyte locations
\mathcal{H}_m	Number of Hankel columns
\mathcal{H}_n	Number of Hankel rows
T_{len}	Transfer function sampling length
F_s	Transfer function sampling frequency
$S_{s,m}$	Number of spacial electrode locations
M	System order
T_s	Final system sampling time
F	Faraday's constant
T	Cell Temperature
R	Universal gas constant

acquisition, insight into these properties of the cell are not available, making predictions of long-term battery pack degradation difficult and inaccurate [1,5–8]. For many OEMs that are looking to have a vast array of performance from commuter vehicles to high-performance sports cars, this lengthy process is needed for each separate cell and pack configuration they wish to utilise. An alternative to data-driven models is physics-based models, such as the Doyle–Fuller–Newman Pseudo-2D (DFN) [9,10] or the Single Particle Model (SPM) [11]. These models provide internal electrochemical insight and can offer a viable solution for degradation-sensitive next-generation cells such as anode-free liquid electrolyte lithium-metal [12–14]. Furthermore, accurate long-term predictions are within these models' capabilities, providing coupling for cell degradation mechanisms such as intercalation electrode lithium plating, loss of active material (LAM) and loss of lithium inventory (LLI), pore-clogging, and dendrite growth [1,14]. This coupling is mathematically complex and requires knowledge of multiple physical parameters which can be difficult and/or expensive to obtain. The information provided from these models is beneficial for theoretical development, cell design iteration and development, as well as pack

design and validation. This work enables advancements at a lab-based level where computational load and time are available; however, it is not currently a reasonable solution for on-board deployment. Due to the beneficial information provided by physics-based models, work has been completed to reduce the numerical complexity and computational performance requirements. Simplification of the partial differential equations governing the system is one such method and has resulted in the SPM and its electrolyte capable (SPMe) form [11]. Additional methods include, Padé approximations [15], residue grouping [16,17], and parabolic solid-phase diffusion approximations [18]. Further reduction is required to achieve deployment on battery management systems. One such reduction method reduces the partial differential equations to continuous-form transfer functions combined with eigen-system realisation algorithms [4,19,20]. Likewise, Jin et al. developed a reduced-order capacity-loss model for graphite anodes, that focused on only the most significant degradation mechanisms to improve computational efficiency [21]. Similarly, Han et al. developed a reduced order lumped electrochemical-thermal cell model by applying a state space approach to transform partial differential equations into ordinary differential equations [22]. These reductions aim to deploy capable predictive models to battery management systems and are heavily numerically reduced. The reduction of these models provides a computationally reasonable model that provides electrochemical information that is then capable to provide this additional insight beyond what a conventional ECM is capable of.

As discussed above, there are numerous benefits to deploying these reduced-order electrochemical models onto a BMS, including improved accuracy for predictions in SOC, SOP, and SOH; [23,24] however, to ensure robust, stable operation of the BMS, the deployed model needs to be real-time capable for the given hardware. This requirement is fulfilled if the online model can be solved before the BMS is required to communicate the solution or provide a control interaction. Depending on the application, this solution rate can have requirements as low as 1 Hz to upwards of 10 Hz in fast dynamic systems. Therefore, the final reduced model must be capable within these ranges to be seen as a viable solution. In this paper, a novel software package developed in Julia [25] is presented for generating numerically reduced real-time capable physics-based models. This work presents an improved realisation algorithm for fast solution generation as well as investigates in-vehicle model creation as a viable method for degradation-informed models. Finally, a sensitivity analysis is performed alongside numerical verification and experimental validation. By combining the capabilities of reduced-order models with the high performance of Julia, LiIBRA.jl enables adaptive physics-informed models to be implemented into embedded systems, opening up the capabilities of battery management and control systems within energy storage.

2. Methodology

This section discusses the methodology for the model development starting with the initial full-order model. A derivation of the continuous-form transfer functions is presented, followed by an introduction to the improved reduction method. Finally, the novel open-source LiIBRA.jl package is introduced.

2.1. Continuum order model

This work starts with the Doyle–Fuller–Newman (DFN) continuum model as the high-order model for reduction. This model, first presented in two main publications [9,10], is a popular choice for continuum-level electrochemical battery modelling due to its ability to capture multi-scale electrochemical processes within a lithium-ion cell. The DFN describes electrochemical electrodes of scale $\sim 100 \mu\text{m}$ and active material particle size of scale $\sim 1 \mu\text{m}$. These length scales are modelled one-dimensionally and coupled to produce a pseudo-two-dimensional model space, often alternatively known as the “P2D”

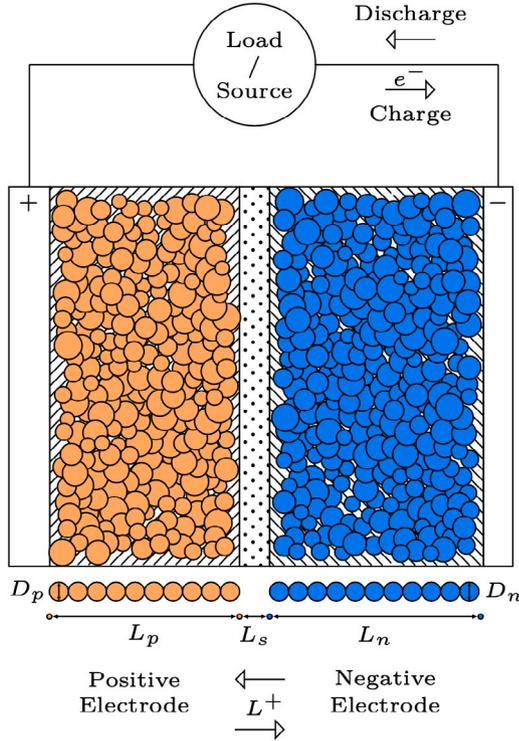


Fig. 1. Doyle-Fuller-Newman model diagram with scale accurate sizing of electrodes, separator, and current collectors for an LG M50 cell.

model. The geometry captured includes three domains: the positive electrode, the negative electrode, and the separator. A diagram describing this structure is presented in Fig. 1 below.

Charge transfer reactions are distributed through electrode thickness $x \in [0, L_{tot}^*]$ with intercalated lithium diffused through the spherical domain $r \in [0, R_k^*]$ where $k \in \{n, p\}$ and “*” denotes variables of dimensionless domain. Porous electrode theory is utilised to capture mass and charge balance in the electrolyte and electrode domains as well as charge transfer kinetics. Through this theory, the porous electrodes are defined as a superposition of three states: electrolyte, electrochemically active material, and non-active materials [26]. Concentration solution theory describes the ionic species transport in the electrolyte domain through relation of the electrochemical potential gradient to the mass flux [27]. The above, combined with the differentiated applied current density source term gives an electrolyte mass balance capturing the concentration evolution as shown in Eq. (3). Charge balance describes the current transferred from the porous electrode to the electrolyte with Ohm’s law governing the charge conservation in the electrodes via Eq. (6). In the electrolyte, the transport equation, as shown in Eq. (4) accounts for ionic diffusion and migration. Coupled charge transfer kinetics in the electrode and electrolyte are captured via the presented Butler-Volmer relation in Eq. (7). This provides a relation between exchange current density and the potential difference across domains. Lastly, solid state (de)intercalation is represented through Fickian diffusion in Eq. (3) in Table 1.

To simplify the spatial domains for the DFN, the respective dimensionless species lengths are transformed to,

$$\gamma_n^* = [0, L_n^*], \quad (10)$$

$$\gamma_s^* = [L_n^*, L_n^* + L_s^*], \quad (11)$$

$$\gamma_p^* = [L_n^* + L_s^*, L_n^* + L_s^* + L_p^*] \quad (12)$$

The spacial variables are then defined within the above domains and shown below. These variables are then numerically solved in the governing equations for the DFN.

$$\begin{aligned} c_{e,n}^*, \phi_{s,n}^*, \phi_{e,n}^*, i_{e,n}^*, N_{s,n}^* & x \in [0, \gamma_n^*] \\ c_{e,s}^*, \phi_{e,s}^*, i_{e,s}^*, N_{e,s}^* & x \in [\gamma_n^*, \gamma_s^*] \\ c_{e,p}^*, \phi_{s,p}^*, \phi_{e,p}^*, i_{e,p}^*, N_{s,p}^* & x \in [\gamma_s^*, \gamma_p^*] \\ c_{s,n}^* & r^* \in [0, R_n^*], x \in [0, \gamma_n^*] \\ c_{s,p}^* & r^* \in [0, R_p^*], x \in [\gamma_s^*, \gamma_p^*] \end{aligned}$$

The coupled system above has been numerically solved through methods such as finite element [28], finite difference [29], orthogonal collocation [30–32], and Chebyshev polynomials; [33] however, due to the model complexity these methods are not feasible for implementation in current generation embedded automotive hardware. To achieve a real-time capable representation of the above model, the work present in this article forgoes these methods and investigates eigensystem realisation algorithms, which provide a data-driven approach to capturing system dynamics in state-space form. This method requires observable state data to achieve stable, robust realisation; however, to achieve the targeted in-vehicle model creation capability, the above coupled partial derivative system must be reduced. To accomplish this, a reduction methodology is presented in the section below.

2.2. Derived transfer functions

To achieve the required computational performance for in-vehicle model generation, the nonlinear governing equations shown above are reduced to low-order transfer functions. The derivation of these transfer functions is presented in this section; however, the reader is pointed to Jacobsen and West [34], Smith et al. [35], and Lee et al. [36] for the original derivations. The starting point for this derivation is linearising the Butler-Volmer equation shown in Eq. (7) above. This is completed by defining a system point for linearisation, in this work, this is completed around the equilibrium point, i.e. no system dynamics occurring as per Eq. (13) below.

$$\rho = [\phi_{s-e} = U_{ocp}(c_{s,0}), c_{s,e} = c_{s,0}, c_e = c_{e,0}, j = 0] \quad (13)$$

where ϕ_{s-e} is defined as $\phi_s - \phi_e$ and ρ is introduced as the equilibrium linearisation point. Combining Eqs. (7) and (8) and linearising about ρ produces,

$$\frac{j_k^*}{m_k^*(c_{s,k}^*)^{1/2}(c_{s,k,max}^* - c_{s,k}^*)^{1/2}(c_{e,k}^*)^{1/2}} \Big|_{\rho} = \sinh\left(\frac{F^* \eta_k^*}{2R^* T^*}\right) \Big|_{\rho} \quad (14)$$

The above can then be represented via a two-term Taylor expansion as shown as,

$$\frac{j_k^*}{j_0^*} = \frac{F}{RT} \tilde{\phi}_{s,e} - \frac{F}{RT} \left[\frac{\partial U_{ocp}}{\partial c_{s,e}} \right]_{c_{s,0}} \tilde{c}_{s,e} - \frac{F^2 R_{film}}{RT} j_k^* \quad (15)$$

Finally, solving for $\tilde{\phi}_{s,e}$ and simplifying gives the following linearised Butler-Volmer representation,

$$\tilde{\phi}_{s,e}(z, t) = FR_{tot} j(z, t) \left[\frac{\partial U_{ocp}}{\partial c_{s,e}} \Big|_{c_{s,0}} \right] \tilde{C}_{s,e}(z, s) \quad (16)$$

where R_{tot} is defined as $R_{ct} + R_{film}$, and \sim introduces the debiased parameter definition such that $\tilde{c}_e = c_e - c_{e,0}$. Next, a previously derived transfer function for electrode surface concentration of type shown in 1 has been presented by Jacobsen and West [34] and is presented in Eq. (17) below.

$$\frac{\tilde{C}_{s,e}(z, s)}{J(z, s)} = \frac{R_s}{D_s} \left(\frac{\tanh(\beta)}{\tanh(\beta) - \beta} \right) \quad (17)$$

where, $\tilde{C}_{s,e}$ is the debiased surface concentration defined as $\tilde{C}_{s,e} = C_{s,e} - C_{s,0}$, and $\beta = R_s \sqrt{s/D_s}$. In this context, z is defined as the spatial electrode location, and s is the complex frequency domain. By applying Laplace transformations to both the linearised Butler-Volmer

Table 1
Doyle–Fuller–Newman Governing Equations [9–11].

Governing Equation	Boundary conditions	Equation number
Electrolyte Mass Conservation:		
$\frac{\partial(c_{e,k}c_{e,k}^*)}{\partial t^*} = \frac{\partial N_{e,k}^*}{\partial x^*} + \frac{1}{F^*} \frac{\partial i_{e,k}^*}{\partial x^*},$	$c_{s,n}^* _{x^*=L_n^*} = c_{e,s}^* _{x^*=L_n^*}$	(1)
$N_{e,k}^* = \epsilon_k^b D_e^*(c_{e,k}^*) \frac{\partial c_{e,k}^*}{\partial x^*} + \frac{I^*}{F^*} i_{e,k}^*$	$c_{s,p}^* _{x^*=L^*-L_p^*} = c_{e,s}^* _{x^*=L^*-L_p^*}$ $N_{e,n}^* _{x^*=0} = 0, N_{e,p}^* _{x^*=L^*} = 0,$ $N_{s,n}^* _{x^*=L_n^*} = N_{e,s}^* _{x^*=L_n^*},$ $N_{s,p}^* _{x^*=L^*-L_p^*} = N_{e,s}^* _{x^*=L^*-L_p^*},$	(2)
Electrode Mass Conservation		
$\frac{\partial c_{s,k}^*}{\partial t^*} = \frac{1}{(r^*)^2} \frac{\partial}{\partial r^*} \left(D_{s,k}^* (r^*)^2 \frac{\partial c_{s,k}^*}{\partial r^*} \right)$	$\frac{\partial c_{s,k}^*}{\partial r^*} _{r^*=0} = 0,$ $-D_{s,k}^* \frac{\partial c_{s,k}^*}{\partial r^*} = \frac{-j_k^*}{F^*}$	(3)
Charge Conservation		
$i_{e,k}^* = \epsilon_k^b \kappa_e^*(c_{e,k}^*) \left(-\frac{\partial \phi_{e,k}^*}{\partial x^*} + 2(1 - t^+) \frac{R^* T^*}{F^*} \frac{\partial}{\partial x^*} (\log(c_{e,k}^*)) \right),$	$i_{e,n}^* _{x^*=0} = i_{e,p}^* _{x^*=L^*} = 0,$ $\partial \phi_{e,n}^* _{x^*=L_n^*} = \partial \phi_{e,s}^* _{x^*=L_n^*}$ $\partial i_{e,n}^* _{x^*=L_n^*} = \partial i_{e,s}^* _{x^*=L_n^*} = I_{app}^*$ $\partial \phi_{e,s}^* _{x^*=L^*-L_p^*} = \partial \phi_{e,p}^* _{x^*=L^*-L_p^*}$ $\partial i_{e,p}^* _{x^*=L^*-L_p^*} = \partial i_{e,n}^* _{x^*=L^*-L_p^*} = I_{app}^*$	(4)
$\frac{\partial i_{e,k}^*}{\partial x^*} = \begin{cases} a_k^* j_k^*, & k=n,p, \\ 0, & n=s, \end{cases}$		(5)
$I_{app}^* - i_{e,k}^* = \sigma_k^* \frac{\partial \phi_{s,k}^*}{\partial x^*}$		(6)
Charge Transfer Kinetics		
$j_k^* = j_{0,k}^* \sinh\left(\frac{F^* \eta_k^*}{2R^* T^*}\right),$		(7)
$j_{0,k}^* = m_k^*(c_{s,k}^*)^{1/2} (c_{s,k}^{*,max}) - (c_{s,k}^*)^{1/2} (c_{e,k}^*)^{1/2},$		(8)
$\eta_k^* = \phi_{s,k}^* - \phi_{e,k}^* - U_k^*(c_{s,k}^* _{r^*=R_k^*})$		(9)

equation and the solid phase charge conservation 1 while utilising the transfer function solution, the electrode surface potential is presented in Eq. (18).

$$\frac{\tilde{\Phi}_{s,e}(z, s)}{I_{app}(s)} = \frac{L_n}{A v(s) \sinh(v(s))} \left(\frac{1}{\kappa^{eff}} \cosh(v(s)z) + \frac{1}{\sigma^{eff} \cosh(v(s)(z-1))} \right) \quad (18)$$

where $v(s)$ is introduced as a dimensionless variable defined as,

$$v(s) = L_n \sqrt{\frac{a_s \left(\frac{1}{\sigma^{eff}} + \frac{1}{\kappa^{eff}} \right)}{R_{tot} + \left[\frac{\partial U_{k,ref}}{\partial C_{s,e}} \right] \frac{R_s}{F D_s} \left(\frac{\tanh(\beta)}{\tanh(\beta) - \beta} \right)}} \quad (19)$$

Next, the reaction flux transfer function can then be derived by combining the results obtained through the linearised Butler–Volmer (16) and the surface potential in (18). The final form is shown in Eq. (20) below.

$$\frac{J(z, s)}{I_{app}(s)} = \left(\frac{v(s)}{a_s F L_n A (\kappa^{eff} + \sigma^{eff})} \right) + \left(\frac{\sigma^{eff} \cosh(v(s)z) + \kappa^{eff} \cosh(v(s)(z-1))}{\sinh(v(s))} \right) \quad (20)$$

Using the results from the previous steps, the electrode surface concentration transfer function can be derived as,

$$\frac{\tilde{C}_{s,e}(z, s)}{I_{app}(s)} = \left(\frac{v(s) R_s (z-1)}{a_s F L_n A (\kappa^{eff} + \sigma^{eff})} \right) \times \frac{\sigma^{eff} \cosh(v(s) \cdot z) + \kappa^{eff} \cosh(v(s))}{\sinh(v(s))} \times \left(\frac{\tanh(\beta)}{\tanh(\beta) - \beta} \right) \quad (21)$$

Starting with the solid phase charge conservation and integrating, the corresponding solid potential ϕ_s transfer function is,

$$\frac{\phi_s(z, s)}{I_{app}(s)} = -\frac{L_n \kappa^{eff} (\cosh((z-1)v(s)))}{A \sigma^{eff} (\kappa^{eff} + \sigma^{eff}) v(s) \sinh(v(s))} - \frac{L_n \sigma^{eff} (1 - \cosh(zv(s)) + zv(s) \sinh(v(s)))}{A \sigma^{eff} (\kappa^{eff} + \sigma^{eff}) v(s) \sinh(v(s))} \quad (22)$$

The electrolyte potential can be likewise found through integration of electrolyte charge conservation Eq. 1 and combining it with the previous reaction flux transfer function to obtain an ionic current representation. This process produces a two-term representation that is shown and further expanded on below.

$$\frac{\tilde{\Phi}_e(z, s)}{I_{app}(s)} = [\tilde{\phi}_e(z, s)]_1 + [\tilde{\phi}_e(z, s)]_2 \quad (23)$$

where the first term is subdomain dependent with the positive electrode domain obtained through the previously defined transfer functions as,

$$[\tilde{\phi}_e(z, s)]_1 = -\frac{L_s}{A \kappa_s^{eff}} + \frac{L_n \left(\left(1 - \frac{\sigma_n^{eff}}{\kappa_n^{eff}} \right) \tanh\left(\frac{v_n(s)}{2}\right) - v_n(s) \right)}{A (\kappa_n^{eff} + \sigma_n^{eff}) v_n(s)}$$

$$- \frac{L_p \left(1 + \frac{\sigma_p^{eff}}{\kappa_p^{eff}} \cosh(v_p(s)) \right)}{A (\kappa^{eff} + \sigma^{eff}) \sinh(v_p(s)) v_p(s)}$$

$$+ \frac{L_p \cosh\left(\frac{(L_n + L_s - x) v_p(s)}{L_p}\right)}{A (\kappa^{eff} + \sigma^{eff}) \sinh(v_p(s)) v_p(s)}$$

$$\begin{aligned}
& + \frac{L_p \left(\frac{\sigma_p^{\text{eff}}}{\kappa_p^{\text{eff}}} \cosh\left(\frac{(L_t-x)v_p(s)}{L_p}\right) \right)}{A(\kappa^{\text{eff}} + \sigma^{\text{eff}}) \sinh(v_p(s)) v_p(s)} \\
& + \frac{(L_n + L_s) - x}{A(\sigma_p^{\text{eff}} + \kappa_p^{\text{eff}})} \quad (24)
\end{aligned}$$

and the second term is determined by the value of $c_e(x, t)$ and is shown as,

$$[\tilde{\phi}_e(x, t)]_2 = \frac{2RT(1 - t_+^0)}{F} \log\left(\frac{c_e(x, t)}{c_e(0, t)}\right) \quad (25)$$

Finally, to acquire the electrolyte concentration transfer function the problem is split into homogeneous and non-homogeneous components. This allows for the homogeneous component to obtain an orthonormal eigenfunction representation of the ϵ_e weighting function through the separation of variables. The non-homogeneous component performs a projection of the concentration function into ϵ_e to solve for Fourier coefficients. These are then used to derive the electrolyte concentration transfer function. This derivation is complex and the detail is not presented in this paper, but the final steps in the process give the negative electrode reaction flux as,

$$\begin{aligned}
\frac{j^{\text{neg}}(s)}{I_{\text{app}}(s)} &= \frac{k_1(1 - t_+^0) \hat{L}_n \sin(\hat{L}_n) (\kappa_n^{\text{eff}} + \sigma_n^{\text{eff}} \cosh(v_n(s))) v_n(s)}{AF(\kappa_n^{\text{eff}} + \sigma_n^{\text{eff}}) (\hat{L}_n^2 + v_n^2(s)) \sinh(v_n(s))} \\
&+ \frac{k_1(1 - t_+^0) \hat{L}_n \sin(\hat{L}_n) (\kappa_n^{\text{eff}} + \sigma_n^{\text{eff}}) v_n^2(s)}{AF(\kappa_n^{\text{eff}} + \sigma_n^{\text{eff}}) (\hat{L}_n^2 + v_n^2(s))} \quad (26)
\end{aligned}$$

where $\hat{L}_n = L_n \sqrt{\epsilon_n \lambda_k / D_n}$, and λ_k are the eigenvalues obtained from the homogeneous problem, which is numerically obtained through root finding. The corresponding positive electrode definition is then,

$$\begin{aligned}
\frac{j^{\text{pos}}(s)}{I_{\text{app}}(s)} &= \frac{k_6(1 - t_+^0) \hat{L}_p \cos(\hat{L}_p) (\kappa_p^{\text{eff}} + \sigma_p^{\text{eff}} \cosh(v_p(s))) v_p(s)}{AF(\kappa_p^{\text{eff}} + \sigma_p^{\text{eff}}) (\hat{L}_p^2 + v_p^2(s)) \sinh(v_p(s))} \\
&- \frac{k_5(1 - t_+^0) \hat{L}_p \sin(\hat{L}_p) (\kappa_p^{\text{eff}} + \sigma_p^{\text{eff}} \cosh(v_p(s))) v_p(s)}{AF(\kappa_p^{\text{eff}} + \sigma_p^{\text{eff}}) (\hat{L}_p^2 + v_p^2(s)) \sinh(v_p(s))} \\
&+ \frac{k_6(1 - t_+^0) \hat{L}_p \cos(\hat{L}_{ns}) (\kappa_p^{\text{eff}} + \sigma_p^{\text{eff}} \cosh(v_p(s))) v_p(s)}{AF(\kappa_p^{\text{eff}} + \sigma_p^{\text{eff}}) (\hat{L}_p^2 + v_p^2(s)) \sinh(v_p(s))} \\
&- \frac{k_5(1 - t_+^0) \hat{L}_p \sin(\hat{L}_t) (\kappa_p^{\text{eff}} + \sigma_p^{\text{eff}} \cosh(v_p(s))) v_p(s)}{AF(\kappa_p^{\text{eff}} + \sigma_p^{\text{eff}}) (\hat{L}_p^2 + v_p^2(s)) \sinh(v_p(s))} \\
&- \frac{k_5(1 - t_+^0) \sigma_p^{\text{eff}} (\cos(\hat{L}_{ns}) \kappa_p^{\text{eff}} + \cos(\hat{L}_s) \sigma_p^{\text{eff}}) v_p^2(s)}{AF(\kappa_p^{\text{eff}} + \sigma_p^{\text{eff}}) (\hat{L}_p^2 + v_p^2(s))} \\
&- \frac{k_6(1 - t_+^0) \sigma_p^{\text{eff}} (\sin(\hat{L}_{ns}) \kappa_p^{\text{eff}} + \sin(\hat{L}_s) \sigma_p^{\text{eff}}) v_p^2(s)}{AF(\kappa_p^{\text{eff}} + \sigma_p^{\text{eff}}) (\hat{L}_p^2 + v_p^2(s))} \quad (27)
\end{aligned}$$

with, $\hat{L}_p = L_p \sqrt{\epsilon_p \lambda_k / D_p}$, $\hat{L}_{ns} = L_n s \sqrt{\epsilon_p \lambda_k / D_p}$, and $\hat{L}_t = L_t \sqrt{\epsilon_p \lambda_k / D_p}$. As well, k_1 , k_3 , k_4 , k_5 , and k_6 are obtained by solving the system of equations below.

$$\Psi_n(x; \lambda) = k_1 \cos\left(\sqrt{\frac{\lambda \epsilon_n}{D_n}} x\right) \quad (28)$$

$$\Psi_m(x; \lambda) = k_3 \cos\left(\sqrt{\frac{\lambda \epsilon_m}{D_m}} x\right) + k_4 \cos\left(\sqrt{\frac{\lambda \epsilon_m}{D_m}} x\right) \quad (29)$$

$$\Psi_p(x; \lambda) = k_5 \cos\left(\sqrt{\frac{\lambda \epsilon_p}{D_p}} x\right) + k_6 \cos\left(\sqrt{\frac{\lambda \epsilon_p}{D_p}} x\right) \quad (30)$$

where Ψ_k is the corresponding eigenfunction for the corresponding electrodes and separator. Finally, the electrolyte concentration is presented as,

$$\frac{C_{e,k}(x, s)}{I_{\text{app}}(s)} = \frac{1}{s + \lambda_k} \left[\frac{J_k^{\text{neg}}(s)}{\text{app}(s)} + \frac{J_k^{\text{pos}}(s)}{\text{app}(s)} \right] \quad (31)$$

Combining the above individual transfer functions into a single input multiple output (SIMO) response array provides a single mathematical structure comprising the continuous-time cell impulse response shown

as $G(s)$ below. Efficiently translating this formation into a state-space representation is the basis for the computationally informed discrete realisation algorithm defined in Section 2.3 below.

$$G(s) = \begin{Bmatrix} \frac{C_e(x, s)}{I_{\text{app}}(s)} \\ \frac{\phi_e(x, s)}{I_{\text{app}}(s)} \\ \frac{\hat{C}_{s,e}(z, s)}{I_{\text{app}}(s)} \\ \frac{\phi_s(z, s)}{I_{\text{app}}(s)} \\ \frac{J(z, s)}{I_{\text{app}}(s)} \end{Bmatrix} \quad (32)$$

2.3. Computationally informed discrete realisation algorithm

To utilise the transfer functions derived in the previous section, and create a state-space representation of the system dynamics, a computationally informed discrete realisation algorithm (CI-DRA) is presented in this section. This method utilises the sampled impulse response from the continuous transfer functions derived in Section 2.2 above. The CI-DRA provides a mathematical pathway to achieve the linear state-space realisation of the form shown in (33) below.

$$\begin{aligned}
x[k+1] &= \mathbf{A}x[k] + \mathbf{B}u[k] \\
y[k] &= \mathbf{C}x[k] + \mathbf{D}u[k] \quad (33)
\end{aligned}$$

The CI-DRA incorporates the zero-order hold methodology first presented in the conventional DRA; [19] however, the final system sampling frequency (F_s) and transfer function system sampling period (T_s) are aligned to remove additional mathematical operations and enabling fast solution times. First, the approximate discrete system response is introduced as [37],

$$G(z) \approx G(s) \Big|_{s=\frac{2(z-1)}{T_s(z+1)}} \quad (34)$$

where T_s is the transfer function sampling period. Next, by relating the discrete Fourier transformation of a sequence to its z-transform [38], the following equation is formed,

$$G_d[f] = G\left(\frac{2 \exp(j2\pi f/N) - 1}{T_s \exp(j2\pi f/N) + 1}\right) \quad (35)$$

where N is defined as the number of points captured in the transfer function response and is dependent on the transfer function response length and sampling frequency. For this work, N is sized to be memory efficient instead of computationally efficient in the conventional DRA. Through this sizing, the CI-DRA provides additional computational performance as the size of N impacts the realisation speed. Finally, due to creating a relation between F_s and T_s , the CI-DRA can forgo the interpolation steps required in the conventional DRA. This removal provides a substantial performance improvement for realisation with large values of N . Additionally, this allows for direct manipulation of the discretely sampled transfer function responses, without additional cumulation steps that are required in the conventional DRA. Continuing the realisation process, the Ho-Kálmán [39] algorithm is utilised to form the state-space representation. This is completed via exploitation of the Markov parameters that comprise the resultant discrete response, $G_{(i)}$. This response can be shown in the following form,

$$G_{(i)} = \begin{cases} \mathbf{D} & i = 0 \\ \mathbf{C} \mathbf{A}^{i-1} \mathbf{B} & i = 1, 2, 3, \dots \end{cases} \quad (36)$$

where \mathbf{D} is gathered from the system response at time-step zero. The resultant transfer function response can be formulated into a block Hankel matrix, (37), of Markov parameters. This block Hankel ($H_{k,m}$) has indices corresponding to a subset domain of the discrete-time

```

1 pkg> add LiibRA
2
3 using LiibRA
4
5 Cell = Construct("LG M50") # Construct LG M50 Parameterisation
6 ψ = collect(0.8:-0.05:0.7) # SOC Range
7 T = collect(273.15:5.0:318.15) # Temperature Range
8 SOC = 0.75 # Initial SOC
9
10 A,B,C,D = Realise(Cell,SList,T) # Create Models
11
12 HPPC(Cell,ψ,SOC,4.0,-3.0,A,B,C,D) # HPPC at 75% SOC with 4A, and -3A pulses

```

Fig. 2. Example usage of LiibRA.jl, providing a simple package for creating and simulating reduced-order models.

impulse response.

$$\mathbf{H}_{k,m} = \begin{pmatrix} G_1 & G_2 & G_3 & \dots & G_m \\ G_2 & G_3 & G_4 & \dots & G_{m+1} \\ G_3 & G_4 & G_5 & \dots & G_{m+1} \\ \vdots & \vdots & \vdots & \ddots & G_{m+l} \\ G_k & G_{k+1} & G_{k+2} & \dots & G_{m+k-1} \end{pmatrix} \quad (37)$$

The block Hankel has an additional feature that allows for relation to the controllability and observability matrices,

$$\mathbf{H}_{k,m} = \mathcal{O}_k \mathcal{E}_m \quad (38)$$

This relation provides a mechanism to form the A, B, and C matrices from the block Hankel matrix and is shown in the observability (\mathcal{O}) and the controllability (\mathcal{E}) definitions below. Initially, by exploiting this relation and factoring $\mathcal{H}_{k,m}$ into the two matrices the first step to obtaining these matrices can be completed.

$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{k-1} \end{bmatrix} \quad (39)$$

$$\mathcal{E} = [B \quad AB \quad A^2B \quad \dots \quad A^{m-1}B] \quad (40)$$

To accomplish this factoring, singular value decomposition (SVD) provides the mechanism to reduce the block Hankel through truncation of the system order. The truncated SVD is shown in Eq. (41) below, where Σ_s captures the highest order singular values of the block Hankel in descending order while Σ_n captures the remaining values and is approximately zero. Selection of the size of Σ_s is a compromise between numerical performance and final system fidelity. Further discussion on this compromise is presented in Section 3.2 below.

$$\mathbf{H}_{k,m} = [U_s \quad U_n] \begin{bmatrix} \Sigma_s & 0 \\ 0 & \Sigma_n \end{bmatrix} \begin{bmatrix} V_s^T \\ V_n^T \end{bmatrix} = U_s \Sigma_s V_s^T \quad (41)$$

Next, combining the SVD and the observability and controllability definitions, Eqs. (39) and (40) become,

$$\mathcal{O}_k = U_s \Sigma_s^{1/2} \quad (42)$$

$$C_l = \Sigma_l^{1/2} V_l^T \quad (43)$$

It is then possible to exploit the original structure of the matrices and utilise the intrinsic Markov parameters and obtain the resulting state-space representation as:

$$A = \mathcal{O}_k^\dagger \mathcal{H}_{k,m+1} \mathcal{E}_m^\dagger \quad (44)$$

$$B = \mathcal{E}_l[1:n, 1:m] \quad (45)$$

$$C = \mathcal{O}_k[1:p, 1:n] \quad (46)$$

where n and p are the state-space output size and input size respectively, $\mathcal{H}_{k,m+1}$ denotes a single index forward shifted of the block Hankel matrix, and \dagger is the matrix transpose. With the above equations, the realisation process is completed and a linear system model of state-space form as shown previously in Eq. (33) is acquired. A summary of the CI-DRA method is shown in the list below.

Summary of the CI-DRA:

1. Alignment of final system sampling period and transfer function sampling period, ensuring that T_s is a subset of the transfer function sampling period, F_s enabling the removal of response interpolation.
2. Compute the continuous form frequency response $G(s)$ from derived transfer functions and convert to discrete-time impulse response via inverse fast Fourier transformation, $G_d[f]$.
3. Form the block Hankel matrix from the discrete-time impulse response and compute the in-place singular value decomposition.
4. Perform an in-place operation on the block Hankel matrix to obtain the time-shifted block Hankel matrix.
5. Form a linear state-space system from the Ho–Kálmán algorithm with unstable poles replaced by their magnitudes, and oscillating poles replace by reciprocals.

2.4. Computational implementation

LiibRA.jl, a Julia [25] based package, has been created by the authors for fast computational implementation of the above computationally informed discrete realisation algorithm. This package improves on previously presented implementations of the eigensystem realisation algorithm by providing fast computational solutions while maintaining fidelity. Key improvements include performant truncated SVD support, large array memory optimisation, and performance benefits from the Julia language largely due to the bottleneck from block Hankel matrix formation. Julia provides a high-performance dynamic type-set, with just-in-time compilation and multiple dispatch capabilities. These features provide an effective computational language for scientific computing while providing modern syntax. The open-source code repository for LiibRA.jl can be found at: <https://github.com/BradyPlanden/LiibRA.jl>. Example usage of LiibRA.jl for creation and simulation of the reduced-order models is presented in the code shown in Fig. 2 below.

The high-level structure of LiibRA.jl is shown in Fig. 3 below. The package dependencies are shown and offer improved code reusability while minimising the size of LiibRA.jl. Through distributing the code base and utilising Julia's open-source packages, LiibRA.jl can be modular and flexible while providing improved algorithm selection for compatibility and performance. These dependencies include TSVD.jl [40] for the truncated SVD, Interpolations.jl [41] for spline fitting of the resultant impulse response, and FFTW.jl [42], providing interface for inverse fast Fourier transforms.

To provide stable operation, LiibRA.jl implements the conventional DRA as a fallback method when the conditions required for the CI-DRA are not met. This provides an easy interface for model creation, with feedback provided to end-users on the computational method being utilised. An additional achievement for LiibRA.jl is its capability to be numerically solved on ARM hardware. This provides a mechanism for one of the aims of this article, enabling in-vehicle model creation for parameterisation variability. Further investigations into LiibRA.jl's capabilities are presented in the next section.

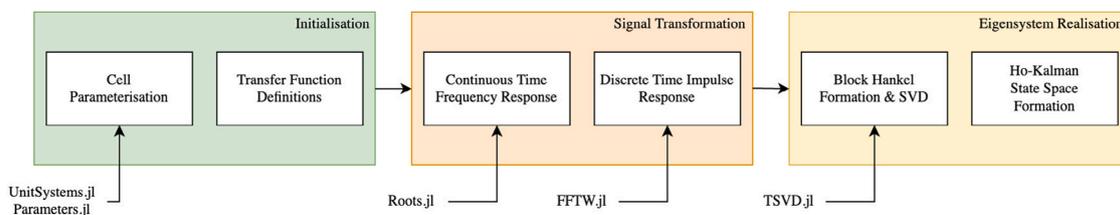


Fig. 3. High-level flowchart of LiiBRA.jl's implementation of CI-DRA with package dependencies listed.

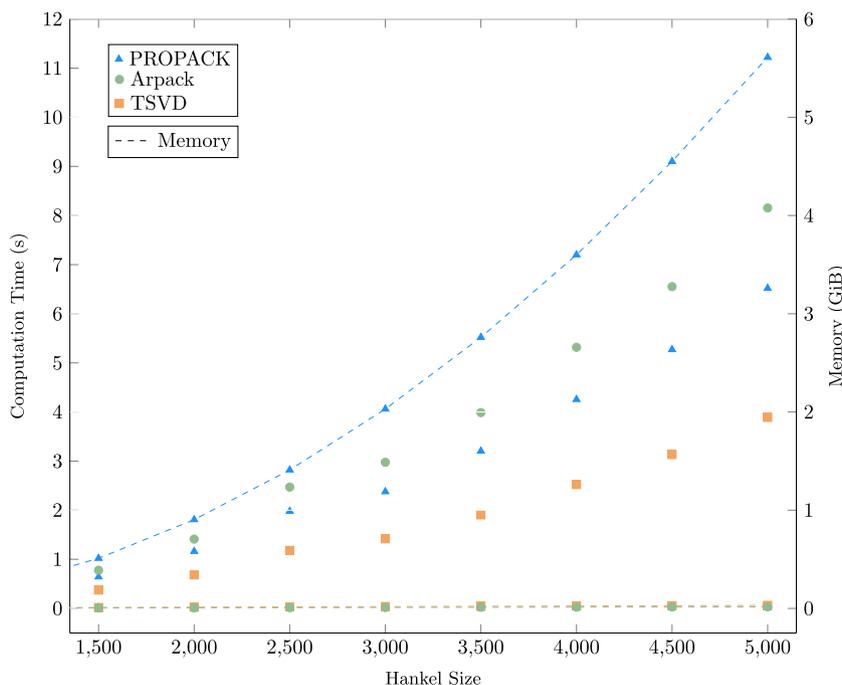


Fig. 4. Computation results of PROPACK.jl, TSVD.jl, and Arpack.jl completing SVD of varying block Hankel sizes.

3. Results

In this section, numerical optimisation of the SVD method used in LiiBRA.jl is presented, followed by parametric sensitivities investigated and verification of the reduced-order models to the full-order system. Finally, experimental validation of the reduced-order model is presented through an automotive drive cycle. x86 computational results for this work were gathered on a 2019 Macbook Pro 13" Intel i5 with ARM results obtain on a Qualcomm Snapdragon 845 with a Ubuntu 18.04.5 LTS operating system, both utilising Julia version 1.7.2.

3.1. Singular value decomposition methods

The singular value decomposition was found to have a large impact on the total numerical solution time for the CI-DRA algorithm. Optimising this truncated singular value decomposition was required to achieve the performance aims in this work. Three algorithms were investigated due to their high computational efficiency: Arpack.jl [43], PROPACK.jl [44], and TSVD.jl [40]. Arpack.jl is a Fortran wrapper of the implicitly restarted Arnoldi method [45], reducing to the implicitly restarted Lanczos method for symmetric matrices. PROPACK.jl is likewise a Fortran wrapper of the Fortran PROPACK software, initially developed by R.M. Larsen [40]. This package implements the Lanczos bidiagonalization method with partial reorthogonalisation and implicit restart in which it acts directly on the system matrix without forming the equivalent system in memory. Similarly, TSVD.jl implements the Lanczos bidiagonalization method with partial reorthogonalisation; however, it is implemented directly with Julia.

To capture each methodology's performance, the block Hankel size, as defined in Eq. (37), is varied. As the block Hankel determines the amount of system response that is captured in the model formation, it is utilised as a variable in this comparison. Analysing the results presented in Fig. 4 below, it is clear that TSVD.jl provides the lowest solution time across the presented range of block Hankel sizes, with PROPACK.jl and Arpack.jl following respectively. PROPACK.jl is shown to use significantly more memory with the difference between TSVD.jl and Arpack.jl being negligible. For SVD computations in LiiBRA.jl, the TSVD.jl package was then chosen as it enables a large range of solutions regarding block Hankel size without having to compromise on performance.

To investigate the CI-DRA's capabilities over the conventional DRA, a comparison between the two is completed across differing transfer function system response lengths as presented in Fig. 5 below. As the CI-DRA provides a mechanism to simplify the model generation through interpolation removal and sampling response point optimisation, the improvements are dependent on the total system response length. Fig. 5 showcase the CI-DRA's ability to capture approximately twice the length of system response over the conventional DRA method for an equivalent computational time from 8 h onwards. This performance improvement enables longer system dynamics to be captured in the reduced-order model generation for improved fidelity in long-term electrochemical predictions. For a 12 h system response, a 21.7% improvement in model generation time is available through the CI-DRA over the conventional DRA.

Table 2
Default values and corresponding ranges for LiIBRA.jl sensitivity analysis of initialisation variables.

Variable	Definition	Default	Range
\mathcal{H}	Size of square Hankel matrix	2500	1500–3500
$S_{e,m}$	Number of spacial particles in electrolyte	6	4–8
M	System order	8	4–12
$S_{s,m}$	Number of spacial particles in electrodes	4	2–6
T_{len}	Length of transfer function sampling time [hr]	4.5	1.0–8.0
F_s/T_s	System sampling frequencies [Hz]	4	2–6

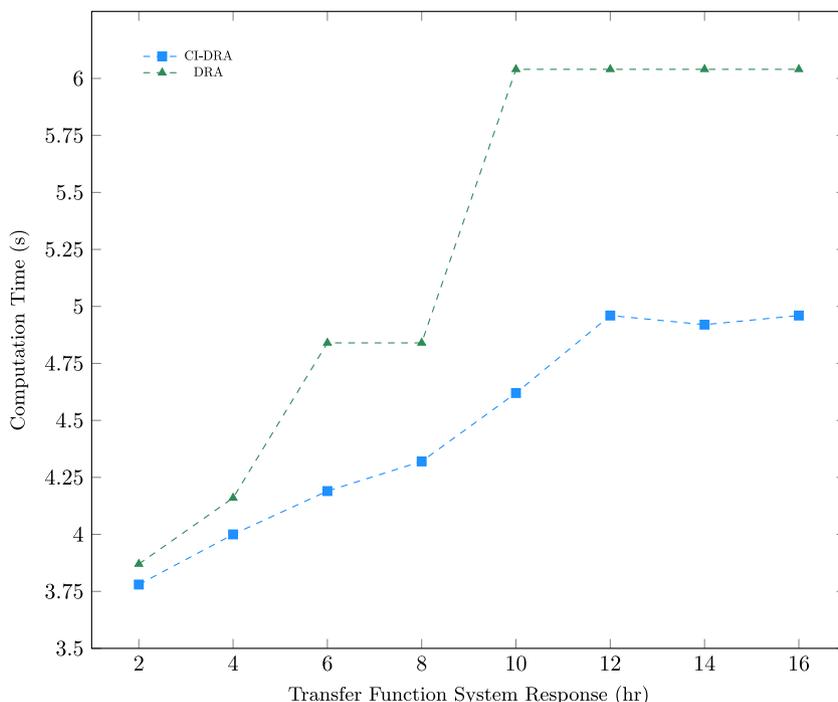


Fig. 5. Computation results comparing the CI-DRA to the conventional DRA for varying transfer function system response length utilising LiIBRA.jl.

3.2. Computational sensitivity

A numerical sensitivity analysis was also completed for the CI-DRA initialisation variables. The variables and the tested ranges are listed in Table 2. An initial investigation was completed to determine stable confines for the model, which were then selected as the default values to minimise numerical instabilities. The minimum and maximum of each range were tested to determine each variable's sensitivity on the resultant computational time. The benchmarking package BenchmarkingTools.jl [46] was utilised to obtain the relevant statistical results. For this work, the minimum number of simulations for each variable set was selected at six to constrain the total number of simulations while reducing the effect of numerical jitter on the analysis. The median computation time for each variable is shown in Fig. 6.

This analysis provides insight towards a minimal configuration of the package for fast model generation. The number of particles in the electrode ($S_{s,m}$), and transfer function sampling length (T_{len}) have the lowest sensitivities, and thus should be selected based on the required model fidelity. The block Hankel size (\mathcal{H}), number of particles in the electrolyte ($S_{e,m}$), and model order (M), have large impacts on the total computational time and these variables should be selected based on a compromise between model fidelity and generation time requirements. For this analysis, the transfer function sampling frequency (F_s) and final system sampling time (T_s) are coupled together, as this provides a stable solution for investigating the CI-DRA.

3.3. Numerical verification

A worldwide harmonised light vehicle test procedure (WLTP) [47] 3B has been implemented for experimental validation as well as nu-

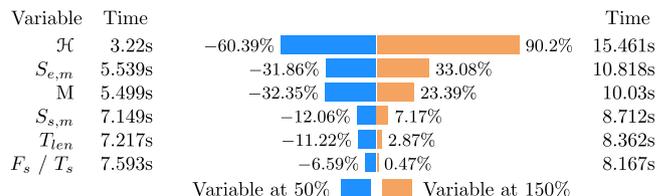


Fig. 6. CI-DRA numerical sensitivity for reduced-order model creation at each variable's lower bound (50% default) and higher bound (150% default). Block Hankel size is shown to have the highest sensitivity, with F_s and T_s having the lowest.

merical verification. This drive-cycle was created from specifications provided for a Tesla Model 3, which are provided in Table A.1 in Appendix below. The resultant specifications were utilised to generate the single-cell scaled power cycle, for a pack designed with LG Chem. M50 cells. LiIBRA.jl was then parameterised with the electrochemical characterisation presented by Chen et al. [48] and was utilised for both model generation and simulation of the WLTP 3B drive-cycle. Additionally, the open-source python battery mathematical modelling package (PyBaMM) [49] was utilised to solve the full order model with the identical parameterisation. Fig. 7 below outlines the predicted terminal voltage and negative electrode concentration for both the reduced-order and full-order models when initialised at the experimentally aligned 75% state of charge value.

For this verification, the block Hankel was sized at 2500 by 2500 elements, with the transfer function and final system sampling time set to 4 Hz, a reduced system order of 6 was utilised, and finally, the transfer function sampling length was set to 4.5 h. The root-mean-square

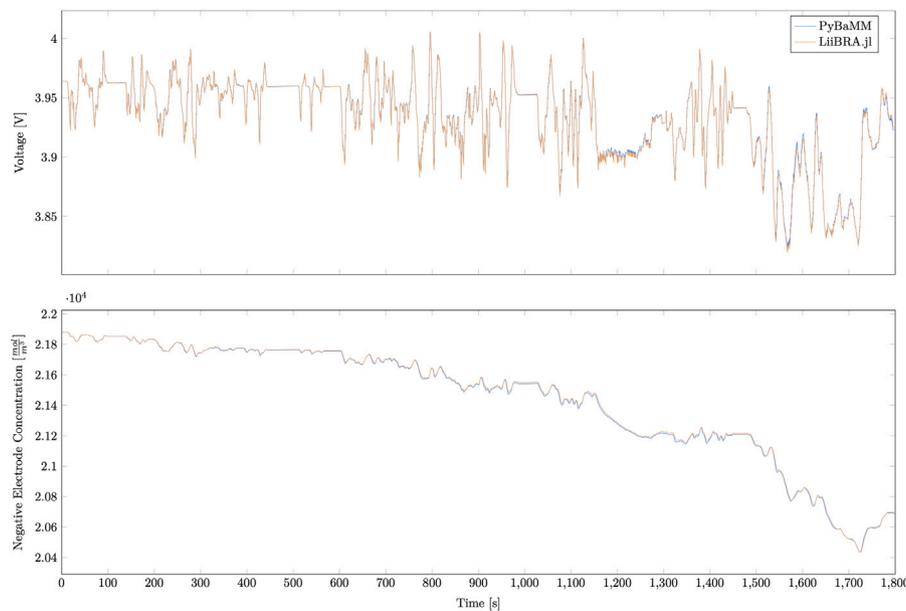


Fig. 7. Comparison of LiiBRA.jl to the full order implementation in PyBaMM, WLTP 3B at 75% SOC and 25 °C for terminal voltage and negative electrode concentration.

deviation between the full-order and reduced-order was 3.64 mV and 5.59 mol/m³ with absolute maximum deviation of 46.68 mV and 14.24 mol/m³ observed. LiiBRA.jl had a total model creation time of 20.2 s with a mean reduced-order simulation time of 135.5 ms for a total combined time of 20.34 s. In comparison, PyBaMM had a mean runtime of 46.28 s. It can be seen from these results that LiiBRA.jl has a similar fidelity of the full-order DFN in PyBaMM; however, runs significantly faster. It should be noted that the 20.34 s model generation time is only required when first generating the reduced-order models, any additional simulations would only require the 135.5 millisecond simulation time.

An additional performance investigation was performed to compare LiiBRA.jl to a previously reported MATLAB implementation [3,4] of the DRA. This implementation was modified to reproduce the CI-DRA methodology, allowing for a direct investigation of LiiBRA.jl's performance. Both models were initialised with identical variable and parameterisation sets. The block Hankel size was varied to represent different fidelity and computational timing compromises. Computational timings are displayed in Fig. 8 below. An additional ARM implementation of LiiBRA.jl is also presented, to confirm capabilities for in-vehicle model generation. The x86 and ARM implementations of LiiBRA.jl perform very similarly with a mean computational time of 4.04 s and 6.06 s respectively. The MATLAB implementation performs significantly worse with a mean computational time of 13.27 s across the simulated block Hankel sizes. A mean computational improvement of 3.51 times is presented across the varied block Hankel sizes for the x86 results. These results enable a three-minute total model generation time for ARM hardware, thus showcasing the viability of LiiBRA.jl for in-vehicle model creation, and enabling physics-based model modifications over the lifetime of the battery pack.

3.4. Experimental validation

An experimental validation of LiiBRA.jl's capabilities is presented below through the parameterisation of an LG Chem. M50 cylindrical 21700 cell [48]. This cell was selected as it is widely available cell that provides a strong reference for the current state of a high-energy intercalation cell with an NCM 811 positive electrode and bi-component Graphite-SiO_x negative electrode. To the author's knowledge, lithium-ion discrete realisation algorithms been verified from full-order and linearised partial differential implementations; [20,36,50,51] however,

experimental validation has not been presented in the literature. In this section, an experimental voltage validation of the CI-DRA utilising LiiBRA.jl and the parameterised LG Chem. M50 dataset is presented.

For this validation, three cells were experimentally tested to reduce cell-to-cell variance. This is seen as the minimum requirement and future investigations are recommended to verify the minimum number of experimentally tested cells required to capture adequate statical variations [52] with respect to LiiBRA.jl. Each cell is initially conditioned at 25 °C for five cycles at a 1C discharge rate and a C/2 charge rate utilising an Arbin LBT21084 cycler and a Binder KB115 incubator. This is followed by a constant current discharge to 75% SOC with a WLTP 3B drive-cycle performed based on the specifications shown in Table A.1. A T-type thermocouple is surface mounted with thermal paste at the body centre of the cell to ensure data consistency.

Fig. 9 below showcases the predicted voltages for the CI-DRA and the experimentally measured cell for the WLTP 3B drive-cycle. These results verify the capabilities of the CI-DRA method and LiiBRA.jl for physics-based predictions with the prediction values producing a root mean square deviation of 7.54 mV to the experimental cell. It should be noted that LiiBRA.jl experiences an increased cell voltage error throughout the length of the drive cycle. This is believed to result from the variation between the experimental and modelled applied current, as well as assumptions made during the experimental cell parameterisation by Chen et al. [48] Online SOC estimation for the conventional DRA has been previously presented [53], which provides a viable correction for this longer-term deviation. Finally, an updated parameterisation of the LG M50 cell is available [54] and includes temperature effects during characterisation, requiring fewer calibration parameters. This parameterisation presents the potential to improve the predicted fidelity without modifying the model architecture.

4. Conclusion

This paper presents an open-source modelling package, LiiBRA.jl, developed in Julia for the creation and simulation of real-time capable electrochemical models. An improved realisation algorithm (CI-DRA) is presented with computational implementation discussed and resulting showing improvements over the conventional method. This work presents capabilities in both offline model creation as well as expansion into in-vehicle creation via ARM compilation. This advancement was feasible due to the high-performance capabilities of LiiBRA.jl, the ease

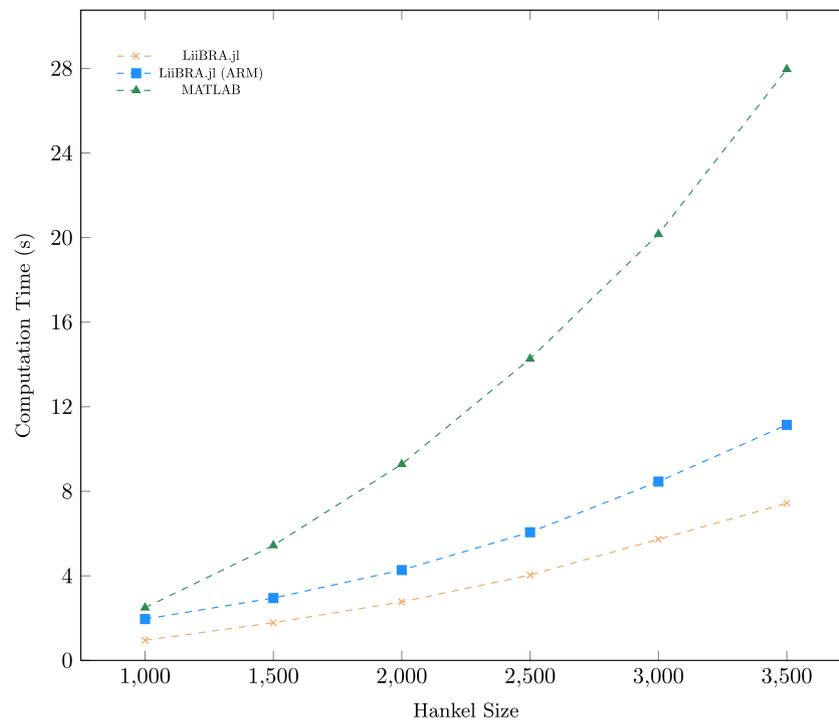


Fig. 8. Computation results for CI-DRA using LiiBRA.jl (x86, ARM), and Matlab (x86) for varying block Hankel sizes with identical system parameterisation.

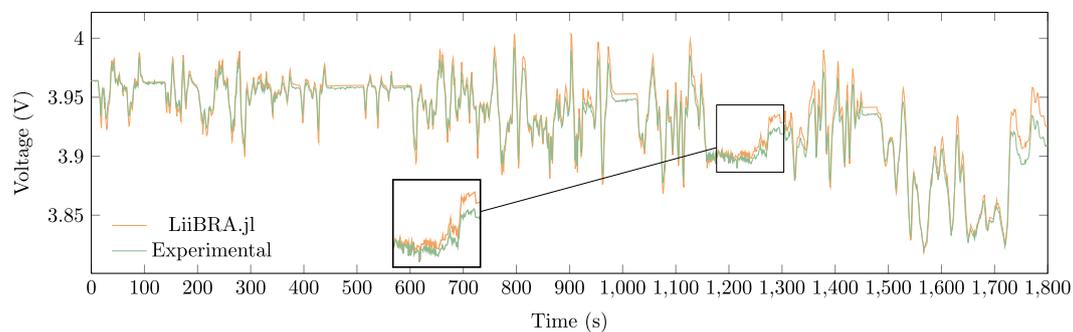


Fig. 9. WLTP 3B voltage validation of CI-DRA implemented with LiiBRA.jl to the experimentally tested cell. This drive-cycle is started at 75% SOC and 25deg C chamber temperature.

of ARM-based compilation, and the improvements enabled by the CI-DRA. This improvement opens future parameterisation of in-vehicle online models, providing a vital mechanism for individualised pack degradation predictions over its lifetime. This package provides a mean value improvement over the presented MATLAB CI-DRA implementation of 3.51 times. For ARM deployment, this package provides a modest 1.53 times decrease in performance when compared to an equivalent x86 characterisation. Investigations showed a computational solution time of 6.06 s per model for ARM-based generation providing a total model creation time of three minutes.

An investigation into the CI-DRA's capabilities over the conventional DRA was presented, showcasing a performance improvement of 21.7% for 12 h of transfer function system response sampling. This was continued into an initialisation variable sensitivity analysis, which presented the CI-DRA's dependencies on block Hankel size, number of spatial particles in the electrolyte, and reduced system order. Experimental validation of the CI-DRA was completed, with voltage prediction of a WLTP 3B drive-cycle resulting in an RMSE value of 3.67 mV. Introducing LiiBRA.jl for this work has provided a performant solution to the two language problem in real-time embedded computing and further reduces resources in software creation and maintenance.

Further work includes the implementation of a memory-efficient singular value decomposition algorithm similar to the method presented in Gopalakrishnan et al. [51] As well, degradation coupling for in-vehicle degradation informed predictions are currently being implemented.

CRediT authorship contribution statement

Brady Planden: Conceptualization, Methodology, Software, Validation, Visualisation, Writing – original draft, Formal analysis, Investigation. **Katie Lukow:** Investigation, Software, Validation, Visualisation, Writing – original draft. **Paul Henshall:** Writing – review & editing. **Gordana Collier:** Supervision, Funding acquisition. **Denise Morrey:** Supervision, Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Table A.1

Tesla Model 3 Long Range specifications used for the creation of the WLTP 3B.

Variable	Definition	Value	Unit
M	Total Vehicle Mass	1931	kg
E	Onboard Useable Energy	82	kWh
F_D	Vehicle Drivetrain Losses	[55]	N
C_{Nom}	Rated Single Cell Capacity	5	Ahr
V_{Lim}	Operational Voltage Limits	2.5/4.2	V
N_s/N_p	Electric System Orientation	96s47p*	–
η_M	Motor Efficiency	0.827*	–

Data availability

Data will be made available on request.

Appendix

To create the worldwide harmonised light vehicle test procedure (WLTP) 3B for validation of the CI-DRA, a model capturing the longititude vehicle dynamics was created. This model utilised parameter specifications from a 2022 T Model 3 long-range vehicle. Vehicle mass and useable energy were taken from production vehicle specifications. Vehicle drivetrain losses were taken from [55]. From these values, a theoretical pack was fitted using an LG M50 to achieve the reported WLTP 3B range and stored pack energy. The fitted parameters are denoted by *. These parameters are shown in Table A.1 below. It should be noted that this parameterisation is intended as a reference, and not to exactly match the presented vehicle.

References

- [1] J.S. Edge, S. O’Kane, R. Prosser, N.D. Kirkaldy, A.N. Patel, A. Hales, A. Ghosh, W. Ai, J. Chen, J. Yang, S. Li, M.-C. Pang, L. Bravo Diaz, A. Tomaszewska, M.W. Marzook, K.N. Radhakrishnan, H. Wang, Y. Patel, B. Wu, G.J. Offer, Lithium ion battery degradation: What you need to know, *Phys. Chem. Chem. Phys.* 23 (14) (2021) 8200–8221, <http://dx.doi.org/10.1039/D1CP00359C>, URL <http://xlink.rsc.org/?DOI=D1CP00359C>.
- [2] S. Zhao, S.R. Duncan, D.A. Howey, Observability analysis and state estimation of lithium-ion batteries in the presence of sensor biases, *IEEE Trans. Control Syst. Technol.* 25 (1) (2017) 326–333, <http://dx.doi.org/10.1109/TCST.2016.2542115>, URL <http://ieeexplore.ieee.org/document/7464285/>.
- [3] X. Hu, S. Li, H. Peng, A comparative study of equivalent circuit models for Li-ion batteries, *J. Power Sources* 198 (2012) 359–367, <http://dx.doi.org/10.1016/j.jpowsour.2011.10.013>, URL <https://linkinghub.elsevier.com/retrieve/pii/S0378775311019628>.
- [4] G.L. Plett, *Battery Management Systems: Battery Modeling. Vol. 1*, Artech House, Boston, London, 2015, OCLC: ocn909081842.
- [5] M. Petit, E. Prada, V. Sauvant-Moynot, Development of an empirical aging model for Li-ion batteries and application to assess the impact of vehicle-to-grid strategies on battery lifetime, *Appl. Energy* 172 (2016) 398–407, <http://dx.doi.org/10.1016/j.apenergy.2016.03.119>, URL <https://linkinghub.elsevier.com/retrieve/pii/S0306261916304500>.
- [6] J. de Hoog, J.-M. Timmermans, D. Ioan-Stroe, M. Swierczynski, J. Jaguemont, S. Goutam, N. Omar, J. Van Mierlo, P. Van Den Bossche, Combined cycling and calendar capacity fade modeling of a nickel-manganese-cobalt oxide cell with real-life profile validation, *Appl. Energy* 200 (2017) 47–61, <http://dx.doi.org/10.1016/j.apenergy.2017.05.018>, URL <https://linkinghub.elsevier.com/retrieve/pii/S0306261917305251>.
- [7] J.M. Reniers, G. Mulder, S. Ober-Blöbaum, D.A. Howey, Improving optimal control of grid-connected lithium-ion batteries through more accurate battery and degradation modelling, *J. Power Sources* 379 (2018) 91–102, <http://dx.doi.org/10.1016/j.jpowsour.2018.01.004>, URL <https://linkinghub.elsevier.com/retrieve/pii/S0378775318300041>.
- [8] M. Schimpe, M.E. von Kuepach, M. Naumann, H.C. Hesse, K. Smith, A. Jossen, Comprehensive modeling of temperature-dependent degradation mechanisms in lithium iron phosphate batteries, *J. Electrochem. Soc.* 165 (2) (2018) A181–A193, <http://dx.doi.org/10.1149/2.1181714jes>, URL <https://iopscience.iop.org/article/10.1149/2.1181714jes>.
- [9] M. Doyle, Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell, *J. Electrochem. Soc.* 140 (6) (1993) 1526, <http://dx.doi.org/10.1149/1.2221597>, URL <https://iopscience.iop.org/article/10.1149/1.2221597>.
- [10] T.F. Fuller, M. Doyle, J. Newman, Simulation and optimization of the dual lithium ion insertion cell, *J. Electrochem. Soc.* 141 (1) (1994) 1–10, <http://dx.doi.org/10.1149/1.2054684>, URL <https://iopscience.iop.org/article/10.1149/1.2054684>.
- [11] S.G. Marquis, V. Sulzer, R. Timms, C.P. Please, S.J. Chapman, An asymptotic derivation of a single particle model with electrolyte, 2019, [arXiv:1905.12553](https://arxiv.org/abs/1905.12553) [Physics], [arXiv:1905.12553](https://arxiv.org/abs/1905.12553), URL <http://arxiv.org/abs/1905.12553>.
- [12] T. Jang, L. Mishra, K. Shah, A. Subramaniam, M. Uppaluri, S.A. Roberts, V.R. Subramanian, Towards real-time simulation of two-dimensional models for electrodeposition/stripping in lithium-metal batteries, *ECS Trans.* 104 (1) (2021) 131–152, <http://dx.doi.org/10.1149/10401.0131ecst>, URL <https://iopscience.iop.org/article/10.1149/10401.0131ecst>.
- [13] L. Mishra, A. Subramaniam, T. Jang, K. Shah, M. Uppaluri, S.A. Roberts, V.R. Subramanian, Perspective—Mass conservation in models for electrodeposition/stripping in lithium metal batteries, *J. Electrochem. Soc.* 168 (9) (2021) 092502, <http://dx.doi.org/10.1149/1945-7111/ac2091>, URL <https://iopscience.iop.org/article/10.1149/1945-7111/ac2091>.
- [14] J.M. Reniers, G. Mulder, D.A. Howey, Review and performance comparison of mechanical-chemical degradation models for lithium-ion batteries, *J. Electrochem. Soc.* 166 (14) (2019) A3189–A3200, <http://dx.doi.org/10.1149/2.0281914jes>, URL <https://iopscience.iop.org/article/10.1149/2.0281914jes>, Number: 14.
- [15] J.C. Forman, S. Bashash, J.L. Stein, H.K. Fathy, Reduction of an electrochemistry-based Li-ion battery model via quasi-linearization and Padé approximation, *J. Electrochem. Soc.* 158 (2) (2011) A93, <http://dx.doi.org/10.1149/1.3519059>.
- [16] K.A. Smith, C.D. Rahn, C.-Y. Wang, Model order reduction of 1D diffusion systems via residue grouping, *J. Dyn. Syst. Meas. Control* 130 (1) (2008) <http://dx.doi.org/10.1115/1.2807068>.
- [17] V. Ramadesigan, V. Boovaragavan, J.C. Pirkle, V.R. Subramanian, Efficient reformulation of solid-phase diffusion in physics-based lithium-ion battery models, *J. Electrochem. Soc.* 157 (7) (2010) A854, <http://dx.doi.org/10.1149/1.3425622>.
- [18] V.R. Subramanian, V.D. Diwakar, D. Tapriyal, Efficient macro-micro scale coupled modeling of batteries, *J. Electrochem. Soc.* 152 (10) (2005) A2002, <http://dx.doi.org/10.1149/1.2032427>.
- [19] J.L. Lee, A. Chemistruck, G.L. Plett, Discrete-time realization of transcendental impedance models, with application to modeling spherical solid diffusion, *J. Power Sources* 206 (2012) 367–377, <http://dx.doi.org/10.1016/j.jpowsour.2012.01.134>, URL <https://linkinghub.elsevier.com/retrieve/pii/S0378775312002789>.
- [20] A. Rodríguez, G.L. Plett, M.S. Trimboli, Comparing four model-order reduction techniques, applied to lithium-ion battery-cell internal electrochemical transfer functions, *ETransportation* 1 (2019) 100009, <http://dx.doi.org/10.1016/j.etrans.2019.100009>, URL <https://linkinghub.elsevier.com/retrieve/pii/S2590116819300098>.
- [21] X. Jin, A. Vora, V. Hoshing, T. Saha, G. Shaver, R.E. García, O. Wasynczuk, S. Varigonda, Physically-based reduced-order capacity loss model for graphite anodes in Li-ion battery cells, *J. Power Sources* 342 (2017) 750–761, <http://dx.doi.org/10.1016/j.jpowsour.2016.12.099>, URL <https://linkinghub.elsevier.com/retrieve/pii/S037877531631802X>.
- [22] S. Han, Y. Tang, S. Khaleghi Rahimian, A numerically efficient method of solving the full-order pseudo-2-dimensional (P2D) Li-ion cell model, *J. Power Sources* 490 (2021) 229571, <http://dx.doi.org/10.1016/j.jpowsour.2021.229571>.
- [23] K. Smith, C.-Y. Wang, Power and thermal characterization of a lithium-ion battery pack for hybrid-electric vehicles, *J. Power Sources* 160 (1) (2006) 662–673, <http://dx.doi.org/10.1016/j.jpowsour.2006.01.038>, URL <https://www.sciencedirect.com/science/article/pii/S0378775306001017>.
- [24] M.A. Xavier, A.K. de Souza, K. Karami, G.L. Plett, M.S. Trimboli, A computational framework for lithium ion cell-level model predictive control using a physics-based reduced-order model, *IEEE Control Syst. Lett.* 5 (4) (2021) 1387–1392, <http://dx.doi.org/10.1109/LCSYS.2020.3038131>.
- [25] J. Bezanson, A. Edelman, S. Karpinski, V.B. Shah, Julia: A fresh approach to numerical computing, *SIAM Rev.* 59 (1) (2017) 65–98, <http://dx.doi.org/10.1137/141000671>, URL <https://epubs.siam.org/doi/10.1137/141000671>.
- [26] J. Newman, W. Tiedemann, Porous-electrode theory with battery applications, *AIChE J.* 21 (1) (1975) 25–41, <http://dx.doi.org/10.1002/aic.690210103>, URL <https://onlinelibrary.wiley.com/doi/10.1002/aic.690210103>.
- [27] J. Newman, K.E. Thomas-Alyea, *Electrochemical Systems*, John Wiley & Sons, 2012.
- [28] K. Smith, C.-Y. Wang, Solid-state diffusion limitations on pulse operation of a lithium ion cell for hybrid electric vehicles, *J. Power Sources* 161 (1) (2006) 628–639, <http://dx.doi.org/10.1016/j.jpowsour.2006.03.050>, URL <https://linkinghub.elsevier.com/retrieve/pii/S0378775306006161>.
- [29] V.R. Subramanian, V. Boovaragavan, V. Ramadesigan, M. Arabandi, Mathematical model reformulation for lithium-ion battery simulations: Galvanostatic boundary conditions, *J. Electrochem. Soc.* 156 (4) (2009) A260, <http://dx.doi.org/10.1149/1.3065083>, URL <https://iopscience.iop.org/article/10.1149/1.3065083>.
- [30] A.M. Bizeray, S. Zhao, S.R. Duncan, D.A. Howey, Lithium-ion battery thermal-electrochemical model-based state estimation using orthogonal collocation and a modified extended Kalman filter, *J. Power Sources* 296 (2015) 400–412, <http://dx.doi.org/10.1016/j.jpowsour.2015.07.019>, URL <http://arxiv.org/abs/1506.08689>, arXiv:1506.08689.

- [31] L. Cai, R.E. White, Lithium ion cell modeling using orthogonal collocation on finite elements, *J. Power Sources* 217 (2012) 248–255, <http://dx.doi.org/10.1016/j.jpowsour.2012.06.043>, URL <https://linkinghub.elsevier.com/retrieve/pii/S0378775312010439>.
- [32] P.W.C. Northrop, V. Ramadesigan, S. De, V.R. Subramanian, Coordinate transformation, orthogonal collocation, model reformulation and simulation of electrochemical-thermal behavior of lithium-ion battery stacks, *J. Electrochem. Soc.* 158 (12) (2011) A1461, <http://dx.doi.org/10.1149/2.058112jes>, URL <https://iopscience.iop.org/article/10.1149/2.058112jes>.
- [33] R. Drummond, A.M. Bizeray, D.A. Howey, S.R. Duncan, A feedback interpretation of the Doyle–Fuller–Newman lithium-ion battery model, *IEEE Trans. Control Syst. Technol.* 28 (4) (2020) 1284–1295, <http://dx.doi.org/10.1109/TCST.2019.2909722>, URL <https://ieeexplore.ieee.org/document/8694827/>.
- [34] T. Jacobsen, K. West, Diffusion impedance in planar, cylindrical and spherical symmetry, *Electrochim. Acta* 40 (2) (1995) 255–262, [http://dx.doi.org/10.1016/0013-4686\(94\)E0192-3](http://dx.doi.org/10.1016/0013-4686(94)E0192-3), URL <https://linkinghub.elsevier.com/retrieve/pii/S0013468694E01923>.
- [35] K.A. Smith, C.D. Rahn, C.-Y. Wang, Control oriented 1D electrochemical model of lithium ion battery, *Energy Convers. Manage.* 48 (9) (2007) 2565–2578, <http://dx.doi.org/10.1016/j.enconman.2007.03.015>, URL <https://linkinghub.elsevier.com/retrieve/pii/S0196890407000908>.
- [36] J.L. Lee, A. Chemistruck, G.L. Plett, One-dimensional physics-based reduced-order model of lithium-ion dynamics, *J. Power Sources* 220 (2012) 430–448, <http://dx.doi.org/10.1016/j.jpowsour.2012.07.075>, URL <https://linkinghub.elsevier.com/retrieve/pii/S0378775312012104>.
- [37] G.F. Franklin, J.D. Powell, M.L. Workman, et al., *Digital Control of Dynamic Systems*, Vol. 3, Addison-wesley Reading, MA, 1998, pp. 187–210.
- [38] A.V. Oppenheim, J.R. Buck, R.W. Schaffer, *Discrete-Time Signal Processing*. Vol. 2, Prentice Hall, Upper Saddle River, NJ, 2001.
- [39] B. HO, R.E. Kálmán, Effective construction of linear state-variable models from input/output functions, *At-Automatisierungstechnik* 14 (1–12) (1966) 545–548.
- [40] R.M. Larsen, Lanczos bidiagonalization with partial reorthogonalization, *DAIMI Rep. Ser.* 27 (537) (1998) <http://dx.doi.org/10.7146/dpb.v27i537.7070>, URL <https://tidsskrift.dk/daimipb/article/view/7070>.
- [41] P. Dierckx, *Curve and Surface Fitting with Splines*, Oxford University Press, 1995, URL <https://github.com/kbarbary/Dierckx.jl>.
- [42] M. Frigo, S.G. Johnson, The design and implementation of FFTW3, *Proc. IEEE* 93 (2) (2005) 216–231, <http://dx.doi.org/10.1109/JPROC.2004.840301>, Special issue on “Program Generation, Optimization, and Platform Adaptation”.
- [43] R.B. Lehoucq, D.C. Sorensen, C. Yang, *ARPACK users guide: Solution of large scale eigenvalue problems by implicitly restarted arnoldi methods*, 1997.
- [44] Dominique, Alexis, A. Noack, JSOBot, J. Chen, MonssafToukal, A. Siqueira, J. TagBot, *Juliasmoothoptimizers/PROPACK.jl: v0.4.0*, 2021, <http://dx.doi.org/10.5281/zenodo.5090075>.
- [45] W.E. Arnoldi, The principle of minimized iterations in the solution of the matrix eigenvalue problem, *Quart. Appl. Math.* 9 (1) (1951) 17–29.
- [46] J. Chen, J. Revels, Robust benchmarking in noisy environments, 2016, arXiv e-prints, [arXiv:1608.04295](https://arxiv.org/abs/1608.04295).
- [47] United Nations Economic Commission for Europe, Global technical regulation - ECE/TRANS/180/add.15/amend.6, 2021, <https://unece.org/transport/standards/transport/vehicle-regulations-wp29/global-technical-regulations-gtrs>.
- [48] C.-H. Chen, F. Brosa Planella, K. O'Regan, D. Gastol, W.D. Widanage, E. Kendrick, Development of experimental techniques for parameterization of multi-scale lithium-ion battery models, *J. Electrochem. Soc.* 167 (8) (2020) 080534, <http://dx.doi.org/10.1149/1945-7111/ab9050>, URL <https://iopscience.iop.org/article/10.1149/1945-7111/ab9050>.
- [49] V. Sulzer, S.G. Marquis, R. Timms, M. Robinson, S.J. Chapman, Python battery mathematical modelling (PyBaMM), 2021, <http://dx.doi.org/10.5334/jors.309>, URL <https://github.com/pybamm-team/PyBaMM>.
- [50] A. Rodríguez, G.L. Plett, M.S. Trimboli, Improved transfer functions modeling linearized lithium-ion battery-cell internal electrochemical variables, *J. Energy Storage* 20 (2018) 560–575, <http://dx.doi.org/10.1016/j.est.2018.06.015>, URL <https://linkinghub.elsevier.com/retrieve/pii/S2352152X18302032>.
- [51] K. Gopalakrishnan, T. Zhang, G.J. Offer, A fast, memory-efficient discrete-time realization algorithm for reduced-order Li-ion battery models, *J. Electrochem. Energy Convers. Storage* 14 (1) (2017) 011001, <http://dx.doi.org/10.1115/1.4035526>, URL <https://asmedigitalcollection.asme.org/electrochemical/article/doi/10.1115/1.4035526/380537/A-Fast-MemoryEfficient-DiscreteTime-Realization>.
- [52] P. Dechent, S. Greenbank, F. Hildenbrand, S. Jbabdi, D.U. Sauer, D.A. Howey, Estimation of Li-ion degradation test sample sizes required to understand cell-to-cell variability**, *Batter. Supercaps* 4 (12) (2021) 1821–1829, <http://dx.doi.org/10.1002/batt.202100148>, URL <https://onlinelibrary.wiley.com/doi/10.1002/batt.202100148>.
- [53] K.D. Stetzel, L.L. Aldrich, M.S. Trimboli, G.L. Plett, Electrochemical state and internal variables estimation using a reduced-order physics-based model of a lithium-ion cell and an extended Kalman filter, *J. Power Sources* 278 (2015) 490–505, <http://dx.doi.org/10.1016/j.jpowsour.2014.11.135>, URL <https://linkinghub.elsevier.com/retrieve/pii/S0378775314020023>.
- [54] K. O'Regan, F. Brosa Planella, W.D. Widanage, E. Kendrick, Thermal-electrochemical parameters of a high energy lithium-ion cylindrical battery, 2022, <http://dx.doi.org/10.26434/chemrxiv-2022-d2q4n>, preprint, Chemistry, URL <https://chemrxiv.org/engage/chemrxiv/article-details/61e2fd304a603d7d99314784>.
- [55] U.S. Environmental Protection Agency, 2022 Tesla model 3 long range AWD - certification summary information report, 2022.