Supplementary Information: An Algorithmic Framework for Full-order Physics-based Simulations of Electrochemical Impedance Spectroscopy

S1: Secondary Chemistry Validation

In the Section (6) of the main manuscript, the proposed ODE+Iterative solver is validated using an LCO/C6 cell chemistry. To ensure that solver is robust across multiple chemistries, in this section we repeat the solver validation results from Section (6.1)-(6.2) of the main manuscript, for a secondary chemistry. This secondary chemistry is for a Lithium-ion Battery (LiB) with a nickel-rich positive electrode (NMC532) and a graphite negative electrode (C6). The model parameters from this chemistry are primarily taken from the model proposed in Mohtat et al [1] and are listed in Table (1).

Parameter	Description	Units	Negative Electrode	Separator	Positive Electrode
Constant Parameters					
I_{1C}	Applied Current for a 1C charge/discharge rate	A/m^2	25.7	25.7	25.7
A	Cell Area	m^2	1.0	1.0	1.0
SoC_i^{100}	Initial State of Charge for SoC = 100%	-	0.9	-	0.27
SoC_i^{80}	Initial State of Charge for SoC = 80%	-	0.8268	-	0.331
SoC_i^{60}	Initial State of Charge for SoC = 60%	-	0.7198	-	0.4205
SoC_i^{40}	Initial State of Charge for SoC = 40%	-	0.5879	-	0.5307
$c_{s,i}^{max}$	Maximum Solid Phase Lithium Concentration	mol/m^3	28746	-	35380
c_e^{init}	Initial Electrolyte Lithium Ion Concentration	mol/m^3	1000	1000	1000
$D_{s,i}$	Solid Phase Lithium Diffusivity	m^2/s	$5\text{x}10^{-15}$	-	$1.2\text{x}10^{-14}$
$D_{e,i}$	Electrolyte Phase Lithium Ion Diffusivity	m^2/s	$5.35\text{x}10^{-10}$	$5.35\text{x}10^{-10}$	$5.35\text{x}10^{-10}$
$\sigma_{s,i}$	Effective Solid Phase Conductivity	S/m	100	-	100
L_i	Thickness of Domain	m	$120\text{x}10^{-6}$	$30\text{x}10^{-6}$	$160\text{x}10^{-6}$
R_i	Electrode Particle Radius	m	$2.5\text{x}10^{-6}$	-	$2\text{x}10^{-6}$
$\epsilon_{s,i}$	Solid Volume Fraction	-	0.61	-	0.445
$\epsilon_{e,i}$	Domain Porosity	-	0.3	0.4	0.3
$brugg_i$	Bruggeman Coefficient	-	1.5	1.5	1.5
$k_{int,i}$	Intercalation Reaction Rate Constant	$m^{2.5}/(mol^{0.5}s)$	$1.1\text{x}10^{-11}$	-	$5\text{x}10^{-11}$
α_{int}	Intercalation Transfer Coefficient	-	0.5	-	0.5
t_{li}	Lithium-ion Transference Number	-	0.363	0.38	0.38
$C_{dl,i}$	Double-Layer Capacitance	-	0.2	-	0.2
F	Faraday Constant	C/mol	96487	96487	96487
R	Ideal Gas Constant	$J/(mol \cdot K)$	8.314	8.314	8.314
T	Ambient Temperature	K	298.15	298.15	298.15
Parameter Corrections					
$D_{e,i}^{eff}$	Effective Electrolyte Phase Lithium Ion Diffusivity	m^2/s		$D_{e,i}^{eff} = \epsilon_{e,i}(x,t)^{brugg} D_{e,i}$	
$\sigma_{s,i}^{eff}$	Effective Solid Phase Conductivity	S/m		$\sigma_{s,i}^{eff} = \epsilon_{s,i}^{brugg} \sigma_{s,i}$	
$\kappa^{eff}(x,t)$	Effective Ionic Conductivity in the Electrolyte	S/m	1.3	1.3	1.3
Function Parameters					
$a_{s,i}$	Electrode Specific Surface Area	m^{-1}		$a_{s,i} = \frac{3\epsilon_{s,i}}{R_i}$	
$U_{OCP}(x, R_n, t)$	Negative Electrode Open Circuit Potential	V	$U_{OCP}(x,R_n,t) = 0.063 + 0.8\,e^{-75(x+0.001)} - 0.0120 \tanh\left(\frac{x-0.127}{0.016}\right) - 0.0118 \tanh\left(\frac{x-0.155}{0.016}\right)$ $-0.0035 \tanh\left(\frac{x-0.220}{0.020}\right) - 0.0095 \tanh\left(\frac{x-0.190}{0.013}\right) - 0.0145 \tanh\left(\frac{x-0.490}{0.020}\right) - 0.0800 \tanh\left(\frac{x-1.030}{0.055}\right)$		
$U_{OCP}(x, R_p, t)$	Positive Electrode Open Circuit Potential	V	$U_{OCP}(x,R_p,t) = 4.3452 - 1.6518\,y + 1.6225\,y^2 - 2.0843\,y^3 + 3.5146\,y^4 - 2.2166\,y^5 - 0.00005623\,e^{109.451\,y-100.006}$		

EAC model parameters are taken from the PyBaMM parameter set for an NMC532/C6 cell, as defined in the work by Mohtat et al [1]

Table 1: Electrochemical ageing capacitance model parameter values

Using these model parameters, this section follows a similar structure to that in Section (6) of the main manuscript.

Thus, this section is organised as follows: Section (S1.1) begins by validating the proposed hybrid ODE+iterative solver using a 1C rate CCCV cycle. Next, Sections (S1.2.1 - S1.2.2) compare the proposed solver with the PyBaMM solver for sEIS impedance simulations across various States of Charge (SoC). Section (S1.2.3) benchmarks the performance of the proposed solver against state-of-the-art solvers in MATLAB and PyBaMM. To conclude the sEIS benchmarking, Sections (S1.2.4)–(S1.2.5) demonstrate how Equivalent Circuit Model (ECM) fitting can significantly reduce the prediction error between the proposed solver and PyBaMM solvers.

S1.1: 1C CCCV Cycle Validation

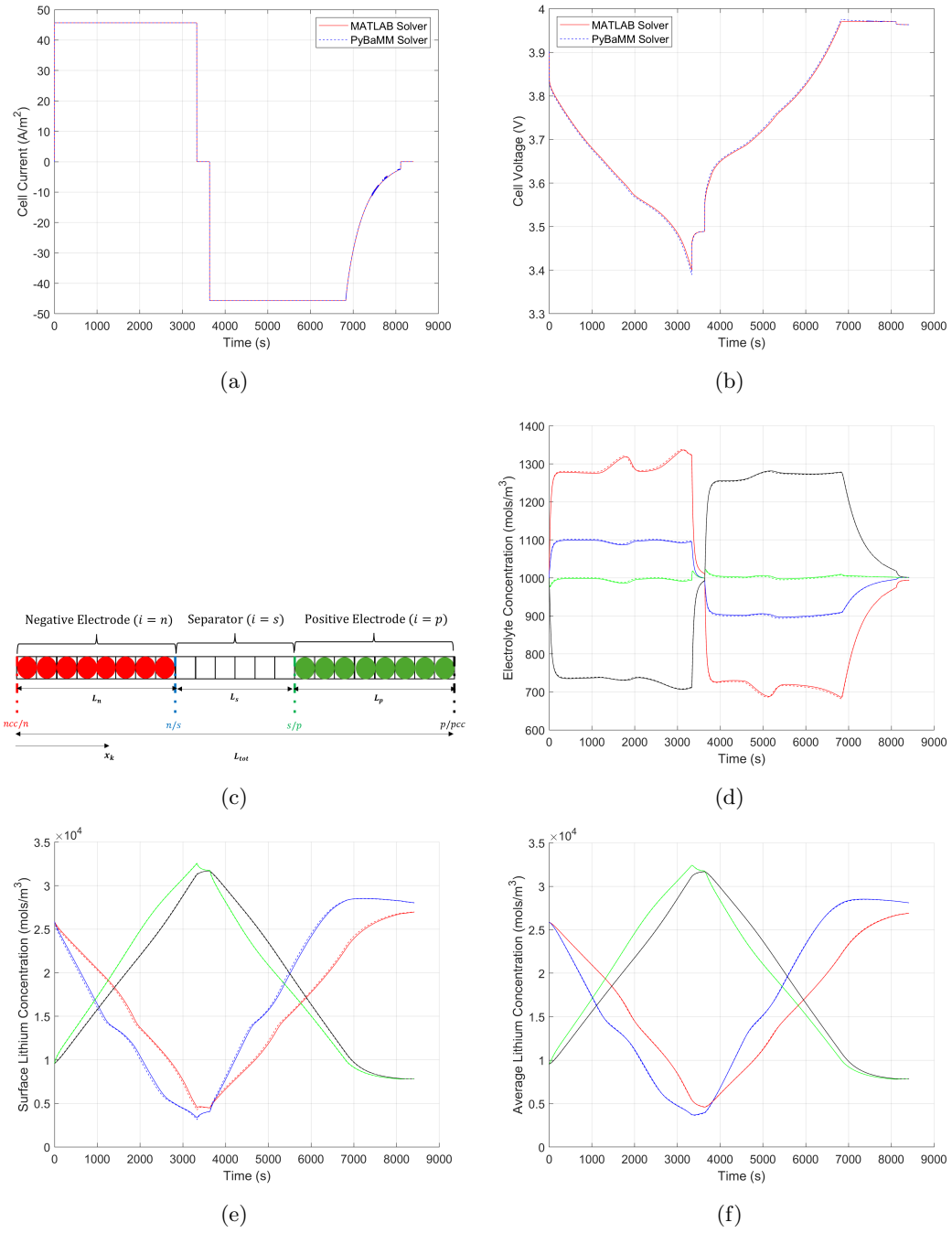


Figure 1: 1C CCCV Cycle validation comparison for NMC532/C6 chemistry: solid lines represent results from the proposed MATLAB solver, and dashed lines represent those from the PyBaMM solver; (a) Applied current; (b) Cell voltage; (c) Schematic showing line locations in the concentration plots — red: negative current collector/negative electrode boundary (ncc/n); blue: negative electrode/separator boundary (n/s); green: separator/positive electrode boundary (s/p); black: positive electrode/positive current collector boundary (p/pcc); (d) Lithium-ion electrolyte concentration; (e) Lithium surface concentration in electrodes; (f) Average lithium concentration in electrodes

For initial secondary chemistry validation of the proposed solver, the EAC model is simulated using both the PyBaMM and the MATLAB ODE+iterative solvers for a 1C CCCV cycle, where the final state of charge (SoC) at the end of simulation is $SoC = 100\%$. The results of this simulation comparison are shown in Figure (1).

Note, in Figure (1), five variables are analysed. These include the applied current $I_{app}(x,t)$, the cell voltage $V_{cell}(t)$, Lithium-ion concentration $c_e(x,t)$, the surface concentration of lithium in the electrodes $c_{s,e}(x,t)$ and the average lithium solid-phase concentration $c_{s,ave}(x,t)$.

For a more quantitative evaluation of solver accuracy, we use the Root Mean Square Error (RMSE) metric of Equation (36) in the main manuscript. This equation is repeated below for an example RMSE calculation as shown in Equation (1), for the $V_{cell}(x,t)$ variable.

$$V_{cell}(t)^{\%}_{error} = \frac{1}{N_{time}} \sum_{t=t_1}^{t_{stop}} \frac{\sqrt{\left(V_{cell}^{MATLAB}(t) - V_{cell}^{PyBaMM}(t)\right)^2}}{V_{cell}^{PyBaMM}(t)} \tag{1}$$

Using Equation (1), the errors for the variables in Figure (1) are calculated and presented in Table (2).

	$\boldsymbol{I_{app}(x,t)}$	$\boldsymbol{V_{cell}(t)}$	$\boldsymbol{c_e(x,t)}$	$\boldsymbol{c_{se}(x,R_i,t)}$	$\boldsymbol{c_{s,ave}(x,t)}$
Percentage Errors	<0.1%	0.1%	0.2%	0.9%	0.9%

Table 2: Error analysis for dependent variables in a single 1C discharge/charge cycle.

As shown in Figure (1) and in Table (2), it is evident that under the 1C CCCV operation condition, the proposed hybrid solver demonstrates high predictive accuracy when benchmarked against PyBaMM for a NMC532/C6 cell model.

S1.2: sEIS Validation

As in Section (6.2) of the main manuscript, in this section we validate the proposed solver against PyBaMM for multiple sEIS test cases for the NMC532/C6 cell chemistry.

S1.2.1: Input Current and Response Voltage Comparison

The first validation case compares the input current perturbation and the corresponding voltage response. In this test, the CCCV cycler is first used to charge the EAC model in MATLAB to 100% SoC. The SoC-related parameters for this case are listed in Table (1).

Subsequently, as outlined in Section (5) of the main manuscript, both the MATLAB based solver and PyBaMM solver are used to simulate sinusoidal currents at set frequencies of $f = [1.4mHz, 1.2Hz, 119Hz]$. The results of this comparison are shown in Figure (2).

The choice of frequencies in Figure (2) is intended to validate the accuracy of the MATLAB ODE+iterative solver across several orders of magnitude.

Using the RMSE definition in Equation (1), the corresponding error values are calculated and summarised in Table (3).

Frequency [Hz]	**Current Error [%]**	**Voltage Error [%]**
1.4x10^{-3}	< 1x10^{-7}	2.9x10^{-4}
1.2	< 1x10^{-7}	3.6x10^{-4}
119	< 1x10^{-7}	2.9x10^{-4}
Average Error	< 1x10^{-7}	3.13x10^{-4}

Table 3: Error analysis for current and voltage sEIS perturbation signals at 100% SoC

From Figure (2) and Table (3), it is evident that the proposed MATLAB solver shows excellent agreement with PyBaMM, demonstrating high accuracy in capturing both the input perturbation and the system's voltage response across a wide frequency range for the NMC532/C6 chemistry.

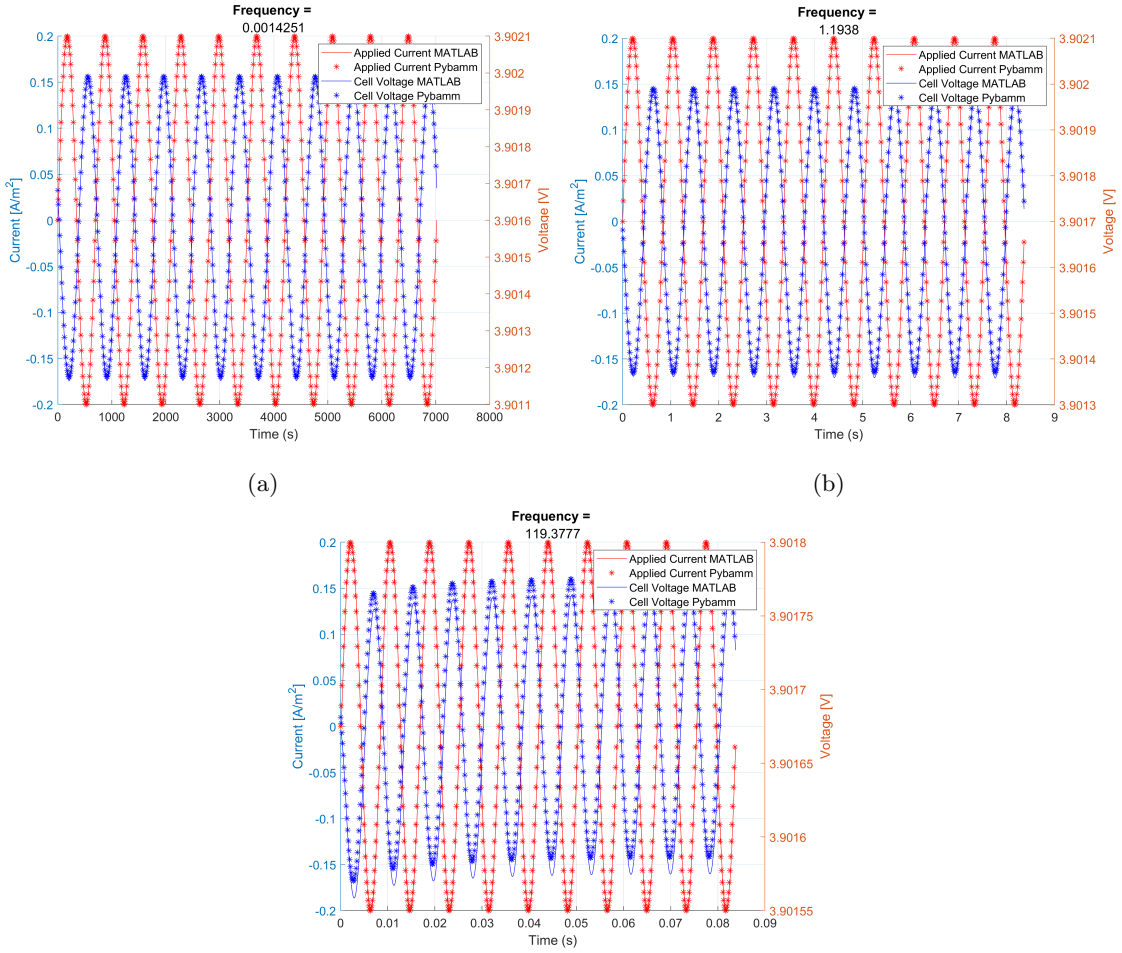


Figure 2: sEIS current & voltage validation for select frequencies; (a) f = 1.4mHz, (b) f = 1.2Hz, (c) f = 119 Hz

S1.2.2: Impedance Spectra Comparison

Given the accuracy of the proposed MATLAB ODE+iterative solver at selected frequencies, this section benchmarks the solver's performance across the full frequency range typically used in an sEIS test. This frequency range is the same as in Section (6.2.2) of the main manuscript, i.e. $1mHz < f < 1kHz$.

To comprehensively validate the solver under sEIS condition, the EAC model is cycled to the same four select SoC points as in Section (6.2.2) of the main manuscript, i.e. $SoC = [40\%, 60\%, 80\%, 100\%]$. Note, the SoC parameters for each of these cases is also shown in Table (1).

The Nyquist impedance spectra from both solvers, at each of the aforementioned SoC points, are shown in Figure (3).

As in section (6.2.2) of the main manuscript, the RMSE definition from Equation (1) is again used to compute the errors between the two solvers. Note however, for computing impedance estimation errors, Equation (1) is applied separately to the real and imaginary components of the impedance spectra in Figure (3). The resulting real and imaginary errors are then averaged to obtain a single estimation error value. The detailed results are shown in Table (4).

It is important to note that in Table (4), the average estimation error across all the SoC cases is 5.3%. It should be noted that this sEIS spectra error is larger than that seen for LCO/C6 cell in the main manuscript. However, as in the main manuscript, we will shown in Section (S1.2.5) that the error can be significantly reduced through ECM fitting.

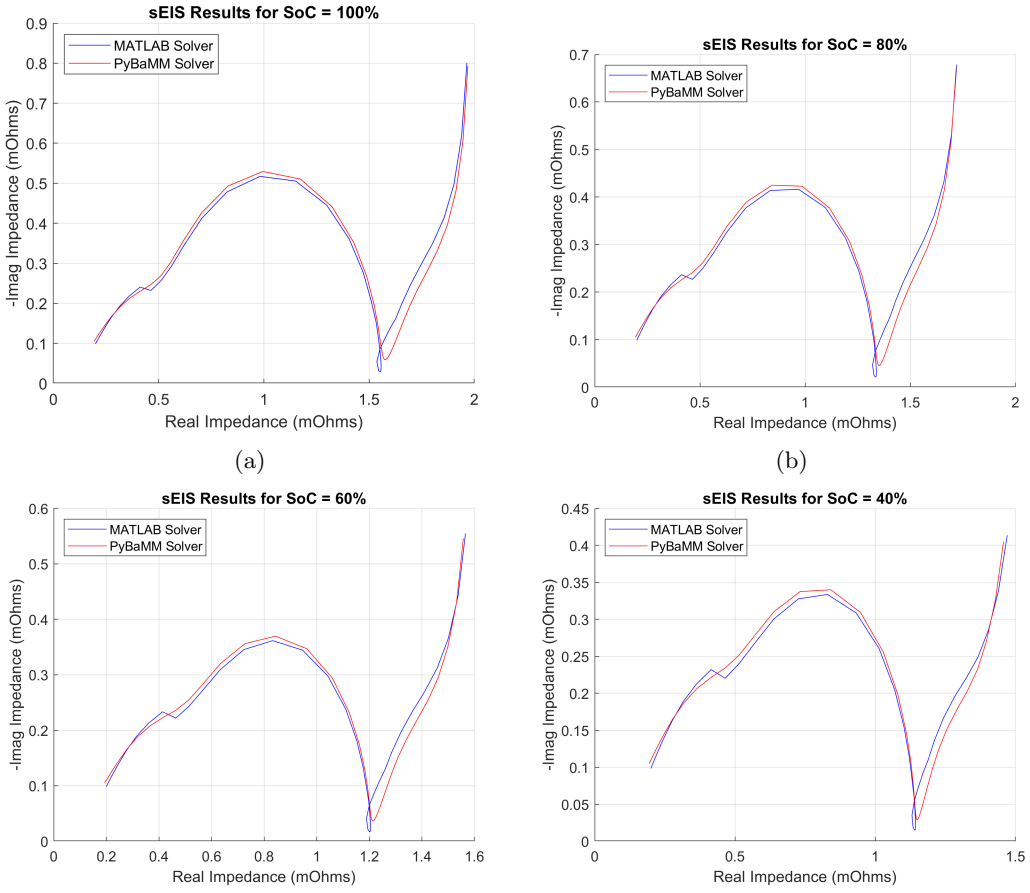


Figure 3: sEIS impedance spectra validation for selected states of charge; (a) SoC = 100%, (b) SoC = 80%, (c) SoC = 60%, (d) SoC = 40%

SoC [%]	**Estimation Error [%]**
100	4.9
80	5.4
60	5.6
40	5.38
Average Error [%]	5.3

Table 4: Error analysis for sEIS impedance spectra

S1.2.3: Solver Performance Comparison

As with Section (6.2.3), in this section we provide a brief discussion on the performance to the proposed ODE+iterative solver for a secondary NMC532/C6 chemistry.

To illustrate the performance benefits of this solving approach, the hybrid ODE+iterative solver is compared against two solvers in PyBaMM; the *SciPy* solver and the *Casadi* solver. In addition, it is also compared to the ODE-only solver implemented using the *ODE15s* integrator in MATLAB. This performance comparison is conducted using an sEIS simulation at a cell state of charge (SoC) of 100%. Note, as in the main manuscript, all performance tests were conducted on the same PC, which has the following specifications: Intel Core i7-9700@ 3.0 GHz processor and 8GB of RAM.

The results for this performance comparison are shown in Figure (4).

From Figure (4), it is visually apparent that the hybrid ODE+iterative solver provides a much more accurate prediction compared to the ODE-only MATLAB solver. The exact error values are presented

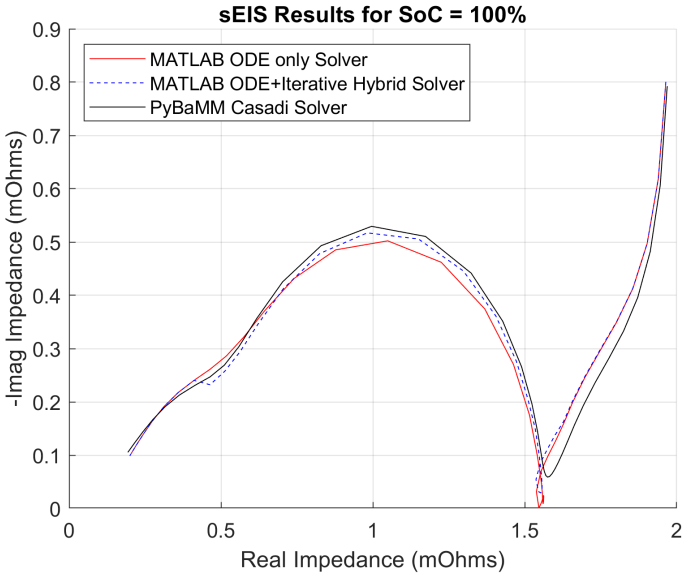


Figure 4: Comparison of sEIS impedance spectra for different solvers

in Table (5). It should be noted that, in Table (5), the difference between the two PyBaMM solvers was found to be negligible and is therefore omitted.

Platform	Frequency Range	EIS Execution Time	EIS Estimation Error
PyBaMM (SciPy Solver)	1mHz-1kHz	175s	-
PyBaMM (Casadi Solver)	1mHz-1kHz	10s	-
Matlab sEIS ODE Only Solver	1mHz-1kHz	90s	15.4%
Matlab sEIS ODE+Iterative Solver	1mHz-1kHz	25s	4.9%

Table 5: Performance analysis of different solvers for an SoC = 100% sEIS test

From the error analysis in Table (5), it is evident that the hybrid ODE+iterative solver achieves nearly a 3x reduction in sEIS estimation error compared to the ODE-only solver. This suggests that solving the DAEs using an iterative shooting method is more accurate than relying solely on MATLAB's *ODE15s* integrator.

To conclude this performance analysis, an sEIS spectra computation speed analysis is also presented in Table (5). Here, the hybrid ODE+iterative solver demonstrates approximately a 3.5x improvement in computational speed over the ODE-only solver. Moreover, the hybrid solver achieves a significantly faster execution time compared to the PyBaMM *SciPy* solver, and a competitive speed relative to the PyBaMM *Casadi* solver.

In summary, the proposed hybrid solver offers a strong balance of high impedance spectra accuracy and competitive computational performance when benchmarked against state-of-the-art electrochemical model solvers available in PyBaMM. The proposed hybrid solver also demonstrates approximately a 3x imporvement in solving accuracy and performance compared to MATLAB's built-in ODE15s solvers. As a result, it can be clearly concluded that the proposed solver structure yields accurate and efficient computations of sEIS impedance spectra.

S1.2.4: ECM Fitting

As in Section (6.2.4) of the main manuscript, in this section, we provide a brief discussion explaining how Equivalent Circuit Models (ECMs) are fitted to the sEIS impedance spectra discussed in Section (S1.2.2). Here, we again use the open-source Python library *impedance.py* to perform the ECM fitting.

In Section (6.2.4), we mentioned that the type of ECM used can vary based on the cell chemistry that is being analysed. For the LCO/C6 cell in the main manuscript, the chosen ECM type was a third-order

Randles model. However, for the secondary NMC532/C6 chemistry analysed in this section, the model is slightly adapted form of the model in the main manuscript and is shown in Figure (5) [2, 3].

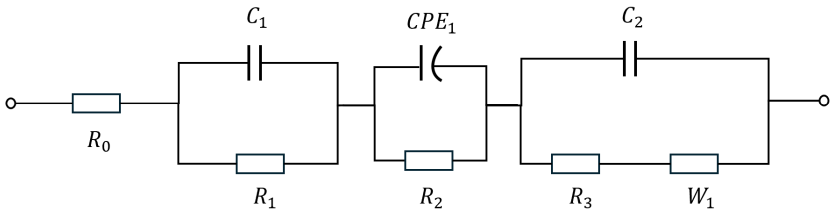
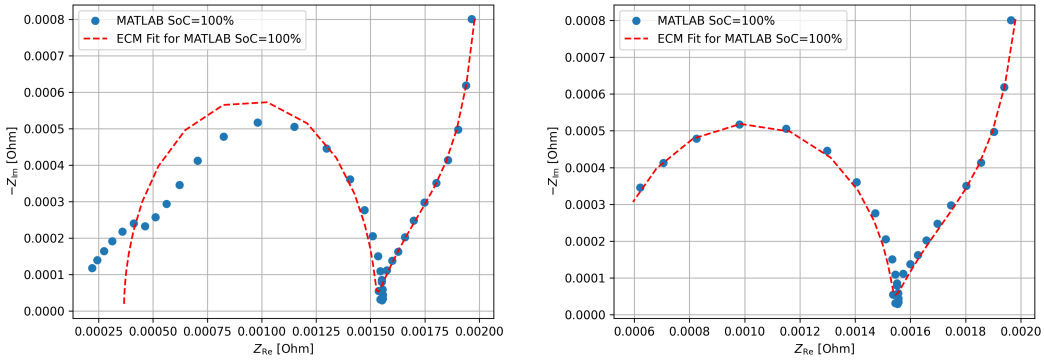


Figure 5: Schematic of an 'adapted' third-order 'Randles' equivalent circuit model

We term this model an 'adapted' third-order Randles model. The primary difference between this model and the one in Section (6.2.4) of the main manuscript is the capacitance C_2, which is swaped for a constant phase element CPE_1.

To analyse the effect of ECM fitting on sEIS data, the 100% SoC sEIS impedance spectrum is first considered. The results of the ECM fitting process are shown in Figure (6).



(a) (b)

Figure 6: Illustration of ECM fitting improvement when ignoring high frequency 'tail' effect; (a) ECM fit with high-frequency impedance points, (b) ECM fit excluding high-frequency points

In Figure (6a), the ECM is fitted across the full frequency range of the sEIS simulation, $1mHz < f < 1kHz$. In this case, the high-frequency sEIS points (i.e. those on the left-hand side of Figure (6a) exhibit increasing deviation from the ECM-predicted impedance. This is primarily due to the high-frequency 'tail' effect present in this region of the spectrum, where the impedance begins to deviate from the semi-circular behaviour induced by double-layer capacitance.

To obtain a more accurate ECM fit, as was done in Section (6.2.4) of the main manuscript, we simply exclude the high-frequency points by limiting the frequency range. For the NMC532/C6 chemistry, this frequency range is $1mHz < f < 50Hz$. The impact of this reduction in frequency range on the fitting error is summarised in Table (6).

Frequency Range [Hz]	**ECM Fit Error [%]**
1mHz-1kHz	7.92
1mHz-50Hz	0.9

Table 6: Analysis of error reduction for ECM fitting from removing high-frequency points

Thus, as was shown in Section (6.2.4) of the main manuscript, limiting the frequency range for ECM fitting can significantly improve the impedance prediction accuracy when applying ECMs to sEIS data.

S1.2.5: ECM Validation

Finally, to conclude this section on secondary chemistry validation, we demonstrate how ECM fitting can help reduce the error deviation between the impedance spectra computed by the proposed solver and PyBaMM. This is visually illustrated through the fitted ECM impedance spectra shown in Figure (7).

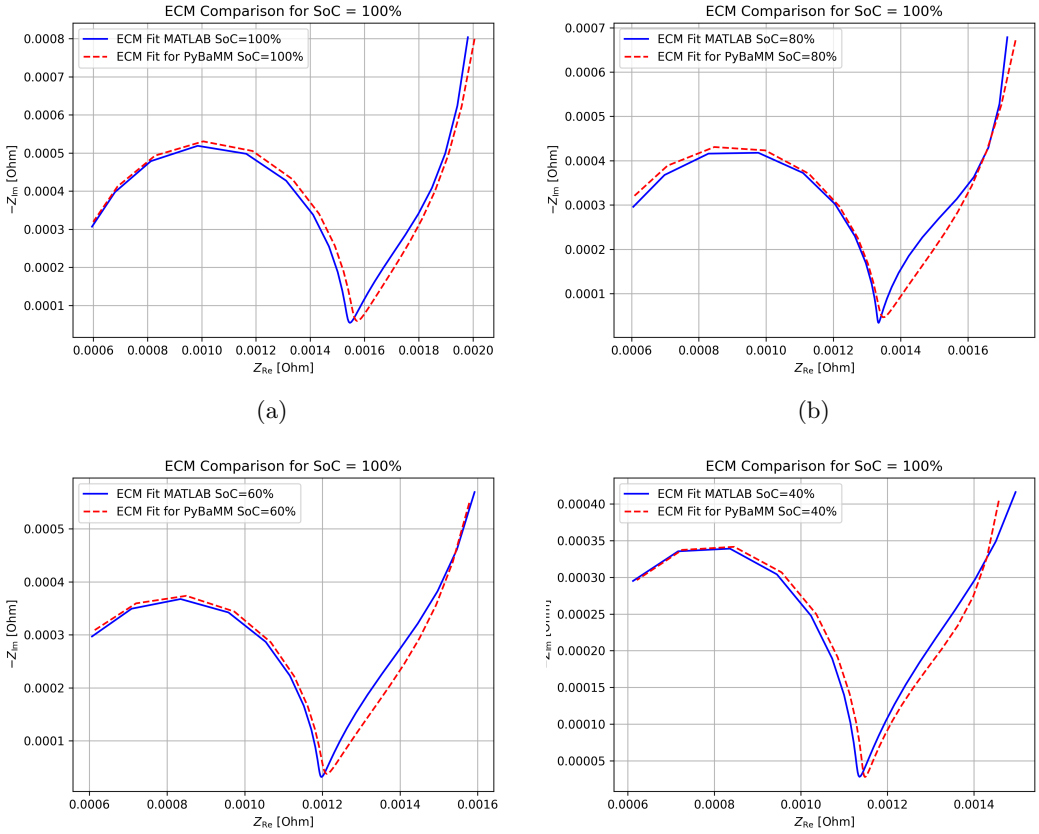


Figure 7: sEIS ECM impedance spectra validation for select states of charge; (a) SoC = 100%, (b) SoC = 80%, (c) SoC = 60%, (d) SoC = 40%

To quantify the impedance spectra shown in Figure (7), we again use Equation (1) to compute impedance error for the fitted ECM spectra. These results are shown in Table (7).

SoC [%]	**ECM Fit Error [%]**
100	0.96
80	1.04
60	0.79
40	0.73
Average Error	0.88

Table 7: Error analysis of fitted ECM impedance spectra

Thus, as was shown in Section (6.2.5) of the main manuscript, fitting ECMs to the sEIS spectra can lead to significant reductions in model impedance prediction error.

References

[1] P. Mohtat, S. Lee, V. Sulzer, J. B. Siegel, and A. G. Stefanopoulou, "Differential expansion and voltage model for li-ion batteries at practical charging rates," *Journal of The Electrochemical Society*, vol. 167, p. 110561, 7 2020.

[2] A. C. Lazanas and M. I. Prodromidis, "Electrochemical impedance spectroscopy-a tutorial," *ACS Measurement Science Au*, vol. 3, pp. 162–193, 6 2023.

[3] R. Li, H. Zhang, W. Li, X. Zhao, and Y. Zhou, "Toward group applications: A critical review of the classification strategies of lithium-ion batteries," *World Electric Vehicle Journal*, vol. 11, no. 3, 2020.