Imperial College London

Introducing the smart toolset for battery parameterisation – BatPar

Tao Zhu

09 September 2022

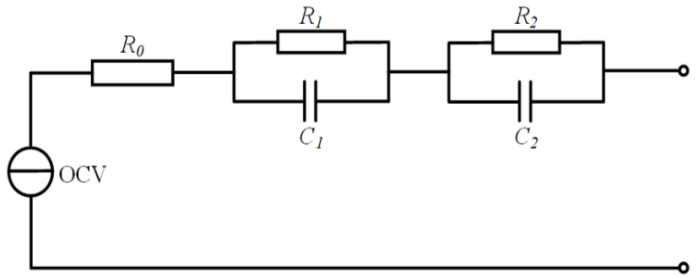
Electrochemical Science and Engineering Group, Imperial College London

Mission of battery parameterisation

Battery parameterisation is to extract equivalent circuit parameters (OCV, Rs and Cs) of the cell from experimental datasheets

	A	B	C	D	E	F	G	H	I	J
1	Today"s D	########								
2	Date of Te	########								
3	Rec#	Cyc#	Step	TestTime	StepTime	Amp-hr	Watt-hr	Amps	Volts	Temp 1
4	1	0	1	0	0	0	0	0	3.7827	27.47
5	2	0	1	3600	3600	0	0	0	3.7826	27.46
6	3	0	2	3600.05	0.05	0	0	-1.6004	3.7789	27.46
7	4	0	2	3634.44	34.44	0.0148	0.0559	-1.5454	3.7689	27.44
8	5	0	2	3705.67	105.67	0.0455	0.1714	-1.5454	3.7589	27.42
9	6	0	2	3796.75	196.75	0.0847	0.3186	-1.5528	3.7489	27.41
10	7	0	2	3899.55	299.55	0.129	0.4843	-1.5454	3.7388	27.37
11	8	0	2	4010.79	410.79	0.1769	0.6632	-1.5454	3.7287	27.32
12	9	0	2	4125.99	525.99	0.2265	0.8479	-1.5491	3.7187	27.32
13	10	0	2	4245.1	645.1	0.2778	1.0384	-1.5509	3.7087	27.36
14	11	0	2	4367.91	767.91	0.3307	1.2343	-1.5546	3.6986	27.36
15	12	0	2	4494.66	894.66	0.3852	1.4359	-1.5509	3.6886	27.32
16	13	0	2	4627.89	1027.89	0.4426	1.6472	-1.5509	3.6786	27.32
17	14	0	2	4761.89	1161.89	0.5003	1.8592	-1.5509	3.6685	27.31

- The experimental data, saved in the format of .csv or .xlsx, should contain time, current and voltage at least, while temperature is optional.
- Users need to know cell capacity and voltage range as inputs. Besides, users also need to decide on the number of RC pairs and the SOC window size as inputs.

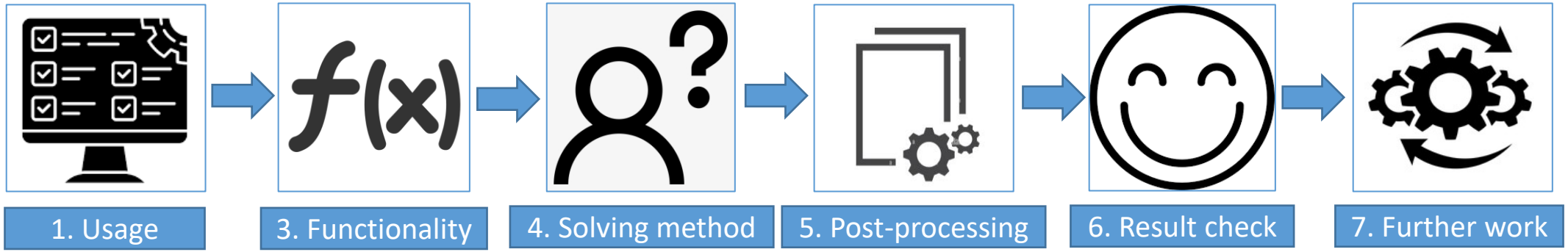


Inputs → BAT PAR → Outputs

	A	B	C	D	E	F	G
1	SOC	OCV	R0	R1	R2	C1	C2
2	0.013787	2.860692796	0.003934135	0.161784309	0.161416353	148.2749022	8720.250554
3	0.05	3.220176513	0.003934135	0.148656373	0.150835751	161.3691504	9331.945697
4	0.1	3.396689543	0.003473767	0.190723964	0.156181796	125.7762901	9012.516687
5	0.15	3.438578653	0.003473767	0.088136379	0.209447006	272.1753817	6720.511635
6	0.2	3.49891243	0.00302554	0.108295281	0.229995166	221.5105998	6120.089672
7	0.25	3.552129566	0.002804208	0.163034183	0.16614686	147.1381781	8471.968939
8	0.3	3.588069143	0.002570741	0.134774002	0.152890283	177.9909496	9206.543521
9	0.35	3.610625759	0.002266572	0.073770173	0.146933999	325.179562	9579.750433
10	0.4	3.652219529	0.002253149	0.100961122	0.1791013	237.6018831	7859.189428
11	0.45	3.688088997	0.002205914	0.148452947	0.165624631	161.5902758	8498.681826
12	0.5	3.712644198	0.002190235	0.12155729	0.151298328	197.3435942	9303.414382
13	0.55	3.75323033	0.002141747	0.079914397	0.207802356	300.1781076	6773.701064
14	0.6	3.802008064	0.002136793	0.155412133	0.164108609	154.3544388	8577.191948
15	0.65	3.83682805	0.002172069	0.097248385	0.161853841	246.6730188	8696.679892
16	0.7	3.855158806	0.002190222	0.019081755	0.086890711	1257.146035	16199.55725
17	0.75	3.9295541	0.002193411	0.101016409	0.142720971	237.4718416	9862.538288
18	0.8	3.995159919	0.002275834	0.158287674	0.158340965	151.5503509	8889.620184
19	0.85	4.013641155	0.002383425	0.011382786	0.094546368	2107.441164	14887.83839
20	0.9	4.10814595	0.002582714	0.14343172	0.145029891	167.2471937	9705.523681
21	0.95	4.148569874	0.002848653	0.110425718	0.106767556	217.2370074	13183.6964
22	1	4.192	0.002858747	0.150345253	0.165755128	159.5564349	8491.99089

- The parameters are SOC dependent and can also be temperature and/or current (rate) dependent - as long as the experimental data is collected at multiples temperatures and/or currents.

Story line



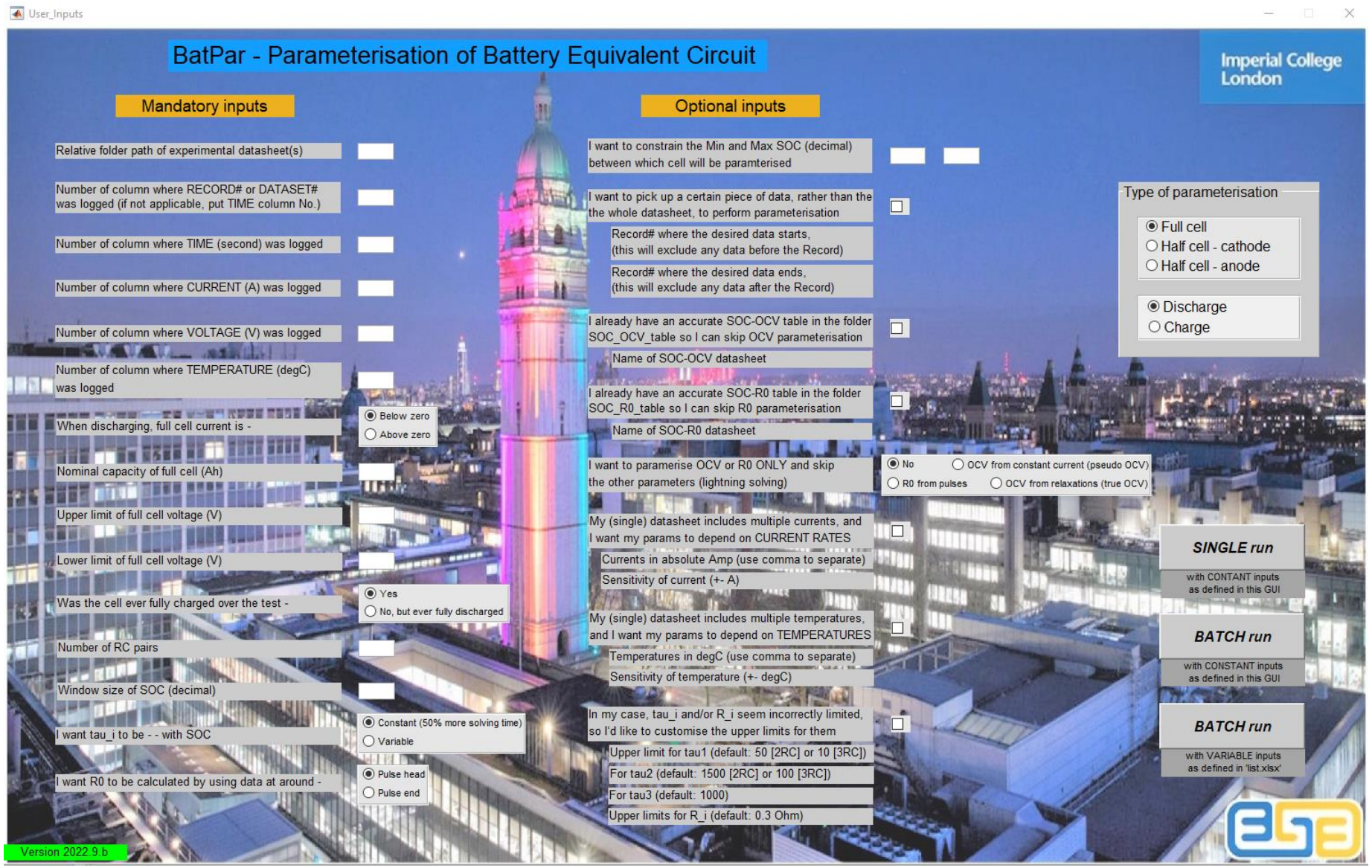
Following the story line, the remaining presentation aims to answer the questions below:

1. Is BatPar easy to use?
2. What functionalities do BatPar have?
3. What is the method / algorithm used to solve OCV, Rs and Cs?
4. How to collect the parameters (results), and is there any visualisation of the parameters?
5. How to check the quality of the parameters, or shall we be confident with the parameters?
6. Any places of improvement in the future?

Usage

Is BatPar easy to use?

- All user operations can be done with the GUI on the right.
- User inputs are divided into mandatory and optional ones. Optional ones can be left blank.
- Fill in the edit boxes, select the radio buttons, tick the check boxes → click one of the three run buttons
- A pop-up window will appear, showing the real-time status of parameterisation.



Functionalities

What functionalities do BatPar have?

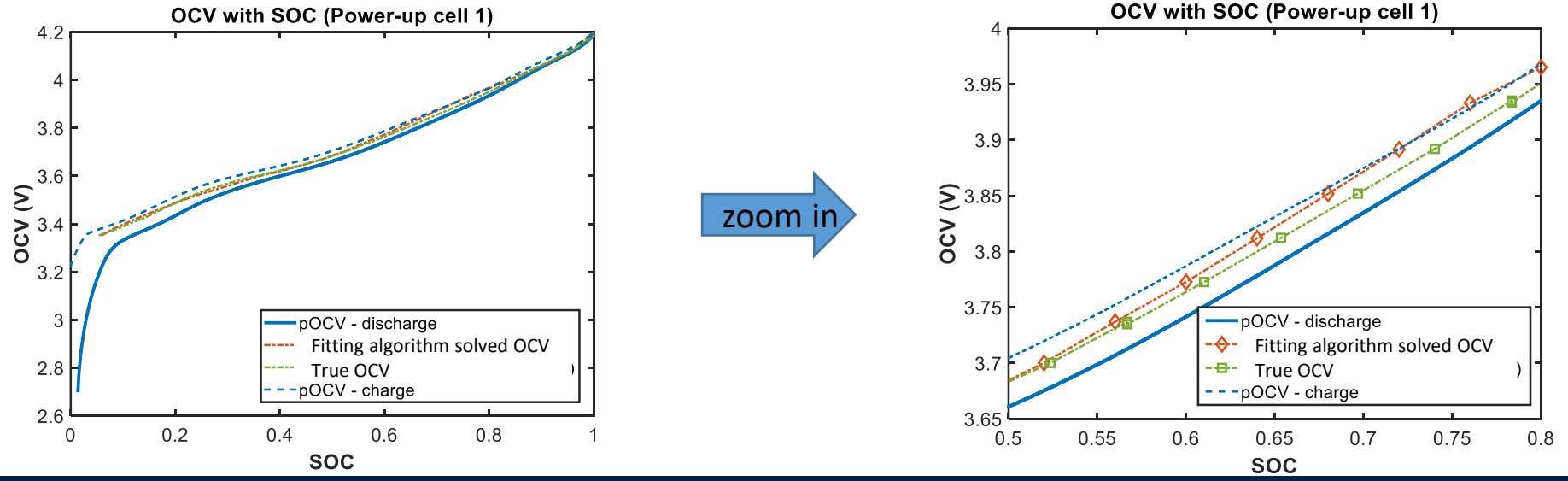
$$\begin{bmatrix}\text{full cell}\\ \text{anode}\\ \text{cathode}\end{bmatrix} \times \begin{bmatrix}\text{discharge}\\ \text{charge}\end{bmatrix} \times \begin{bmatrix}\text{1 RC}\\ \text{2 RC}\\ \text{3 RC}\end{bmatrix} \times \begin{bmatrix}\text{OCV, Rs, Cs}\\ \text{Rs, Cs}\\ \text{OCV only}\\ \text{R0 only}\end{bmatrix} \times \begin{bmatrix}\text{variable tau}\\ \text{constant tau}\end{bmatrix} \times \begin{bmatrix}\text{SOC dependence}\\ \text{SOC + I dependences}\\ \text{SOC + T dependences}\\ \text{SOC + I \& T dependences}\end{bmatrix} \times \begin{bmatrix}\text{Single processing}\\ \text{Batch processing}\end{bmatrix}$$

- Users can customise their demands of parameterisation by selecting one option in each matrix above. No incompatibility witnessed (just for now).
- BatPar can accept data from any standard test procedure like GITT, AMPP, HPPC, or arbitrary tests without pulses (prescribed R0 needed as inputs).
- The functionality combo of “full cell – discharge – SOC dependence only – GITT” was very well verified by parameterising data from many different cells. The other combos were tested with few sets of data.
- I and/or T dependences. The user need to specify: at what currents/temperatures the dependences exist; and the sensitivity of current (±A) / temperature (±degC)
- Batch processing. User puts all datasheets in one folder, and the codes will read and parameterise each datasheet successively. This functionality was used to process 208 datasheets in a row – no crash happened.

Solving method

What is the method / algorithm used to solve OCV, Rs and Cs?

- In BatPar, there are three methods of extracting OCV, 2 methods of extracting R0, and only one method for the other Rs and Cs. User can customise the preferred method of extracting OCV and R0, which is also subject to the type of data user has and data quality.
- OCV extraction: (1) true OCV from relaxations; (2) pseudo OCV from very low current test; (3) Treat OCV as one unknown to be solved together with Rs (excluding R0) and Cs by the fitting algorithm.
- Ture OCV is the best, followed by fitting algorithm solved OCV, pseudo OCV is the worst choice. The following plots give justification.



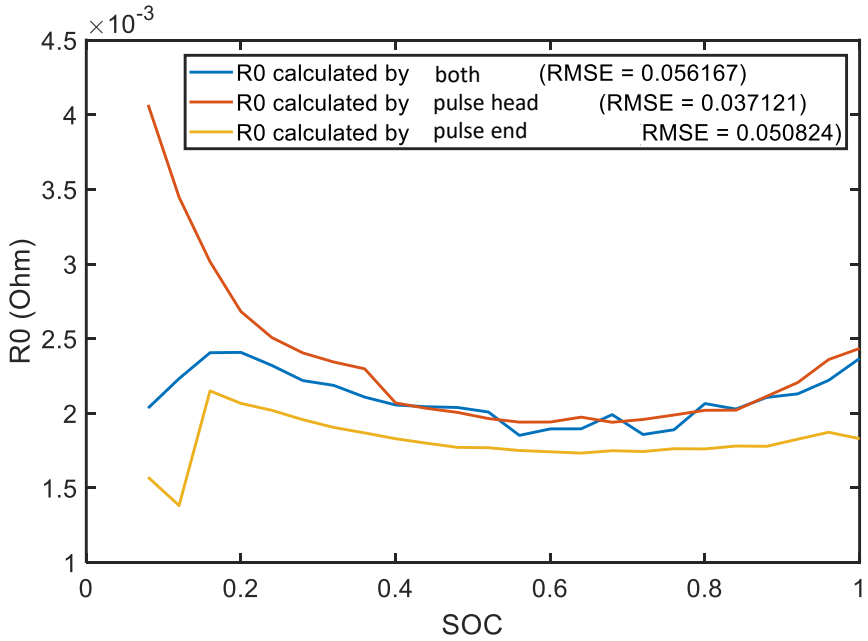
Solving method

What is the method / algorithm used to solve OCV, Rs and Cs?

- R0 extraction: (1) dV/dI at pulse head; (2) dV/dI at pulse end; ~~(3) dV/dI at both head and end of the pulse.~~ Below the plot demonstrates that R0 extracted at pulse head may be the best choice.
- As a rule of thumb, you should always use pulse head rather than pulse end to calculate R0, unless data quality at pulse head is poor, like the screenshot on bottom left.
- Sometimes the data quality at pulse end is also poor (screenshot on bottom middle) → accept the truth that extracted R0 is not accurate

F	G	H	I	J	
ommand	U[V]	I[A]	P[W]	Ah[Ah]	Ah-C
ause	3.790216	0	0	0	
ause	3.789257	0	0	0	
ause	3.788299	0	0	0	
ause	3.78734	0	0	0	
ause	3.786381	0	0	0	
ause	3.785422	0	0	0	
ause	3.785422	0	0	0	
ischarge	3.785614	0.011024	0.041733	7.22E-07	7.2
ischarge	3.75167	-7.49184	-28.1069	-7.65E-06	7.2
ischarge	3.747451	-7.50222	-28.1142	-2.14E-05	7.2
ischarge	3.745917	-7.49963	-28.093	-3.54E-05	7.2
ischarge	3.744958	-7.49379	-28.0639	-4.95E-05	7.2
ischarge	3.743807	-7.49184	-28.048	-0.00011	7.2
ischarge	3.742848	-7.49638	-28.0578	-0.0003	7.2
ischarge	3.741889	-7.49833	-28.0579	-0.0008	7.2
ischarge	3.74093	-7.49638	-28.0435	-0.00142	7.2
ischarge	3.739972	-7.49833	-28.0435	-0.0022	7.2
ischarge	3.739013	-7.50028	-28.0436	-0.00316	7.2

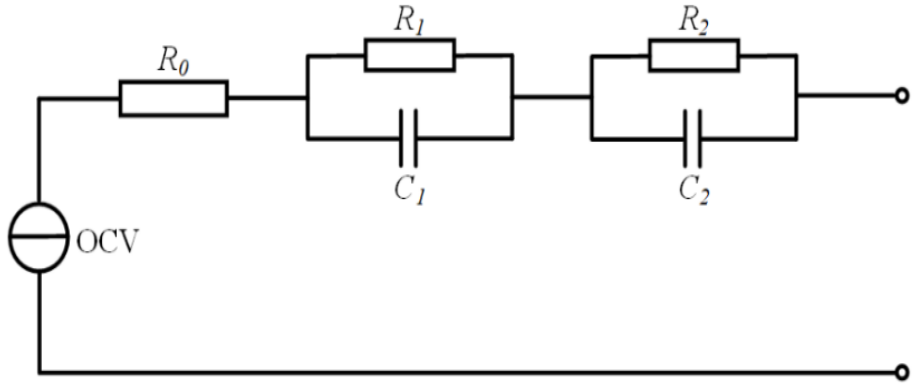
	G	H	I	J
and	U[V]	I[A]	P[W]	Ah[Ah]
rge	2.70727	-7.50028	-20.3053	-2.9812
rge	2.706311	-7.50028	-20.2981	-2.9814
rge	2.705352	-7.49963	-20.2891	-2.9816
rge	2.704393	-7.50028	-20.2837	-2.9818
rge	2.703435	-7.50028	-20.2765	-2.9820
rge	2.702476	-7.50028	-20.2693	-2.982
rge	2.701517	-7.50028	-20.2621	-2.9825
rge	2.700558	-7.50028	-20.2549	-2.9827
rge	2.699599	-7.49444	-20.232	-2.9829
rge	2.700174	-0.06809	-0.18385	-3.0501
rge	2.700174	-0.06874	-0.18561	-3.0501
	2.701133	0	0	-3.0501
	2.701517	0	0	-3.0501
	2.702476	0	0	-3.0501
	2.703435	0	0	-3.0501



Solving method

What is the method / algorithm used to solve OCV, Rs and Cs?

- Rs (excluding R0) and Cs extraction: all Rs and Cs are treated as unknowns to be solved by a fitting algorithm. The fitting algorithm, combined with the governing equations below, aims to find Rs and Cs that can best match the fitting model with experimental data.



$$V = OCV - IR_0 - V_{r1} - V_{r2}$$

$$\dot{V}_{r1} = -\frac{V_{r1}}{R_1 C_1} + \frac{I}{C_1} \xrightarrow{discretisation} V_{r1}(t+1) - V_{r1}(t) = -\frac{V_{r1}(t)}{R_1 C_1} + \frac{I(t)}{C_1}$$

$$\dot{V}_{r2} = -\frac{V_{r2}}{R_2 C_2} + \frac{I}{C_2} \xrightarrow{discretisation} V_{r2}(t+1) - V_{r2}(t) = -\frac{V_{r2}(t)}{R_2 C_2} + \frac{I(t)}{C_2}$$

- The fitting algorithm used in BatPar is fmincon (find a minimum of a constrained nonlinear multivariable function)
- Best match is evaluated by the gap (RMSE) between experimental voltage and fitting model voltage, if the fitting model is given the same current as the experiment.
- To this end, the nature of BatPar is a curve fitting program.

Post-processing

How to collect the parameters (results), and is there any visualisation of the parameters?

- A folder of the same name as your datasheet will be created by BatPar to save parameterisation results.

PUd-cell4-R601-700 - 001.csv_FullCell_DISCHARGE

- Inside the folder:

Log of historical status of parameterisation → Command window log.txt

Data extraction of discharge segments.fig

Comparison between test and fitting model → Global comparison - Experiment VS Paramterised Model.fig

Input variables → INPUT.mat

The sorted parameter look-up table → LookUpTable_AllInOne.xlsx

Intermediate and output variables → OUTPUT.mat

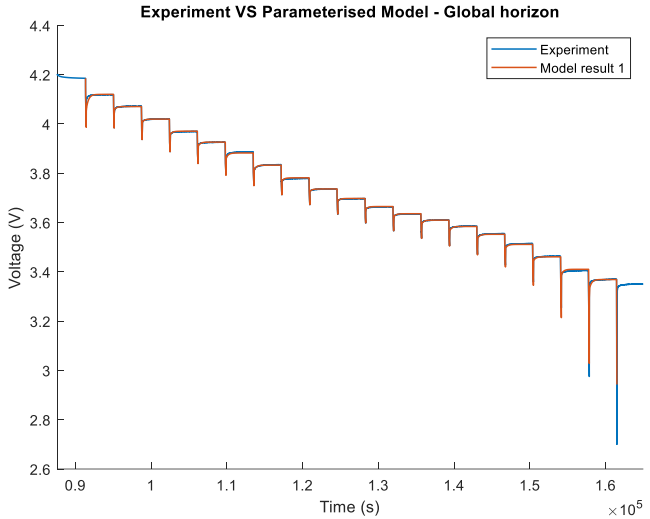
Visualisation (plots) of parameters → Parameterised Ci.fig

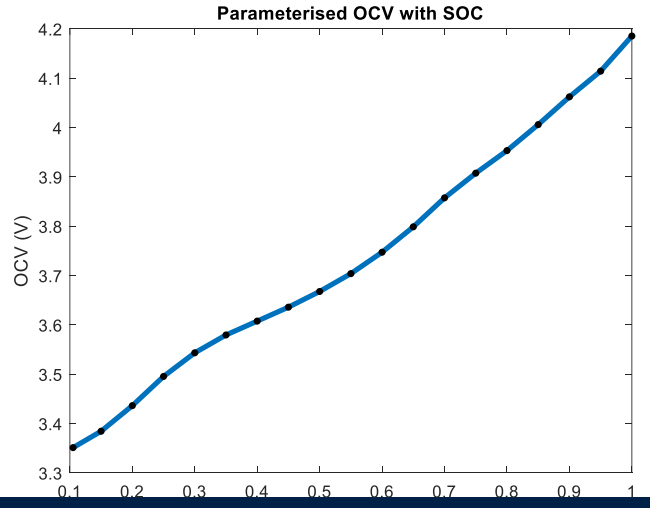
Parameterised OCV.fig

Parameterised R0.fig

Parameterised Ri.fig

Parameterised tau_i.fig





	A	B	C	D	E	F	G
1	SOC	OCV	R0	R1	R2	C1	C2
2	0.013787	2.860692796	0.003934135	0.161784309	0.161416353	148.2749022	8720.250554
3	0.05	3.220176513	0.003934135	0.148656373	0.150835751	161.3691504	9331.945697
4	0.1	3.396689543	0.003473767	0.190723964	0.156181796	125.7762901	9012.516687
5	0.15	3.438578653	0.003473767	0.088136379	0.209447006	272.1753817	6720.511635
6	0.2	3.49891243	0.00302554	0.108295281	0.229995166	221.5105998	6120.089672
7	0.25	3.552129566	0.002804208	0.163034183	0.16614686	147.1381781	8471.968939
8	0.3	3.588069143	0.002570741	0.134774002	0.152890283	177.9909496	9206.543521
9	0.35	3.610625759	0.002266572	0.073770173	0.146933999	325.179562	9579.750433
10	0.4	3.652219529	0.002253149	0.100961122	0.1791013	237.6018831	7859.189428
11	0.45	3.688088997	0.002205914	0.148452947	0.165624631	161.5902758	8498.681826
12	0.5	3.712644198	0.002190235	0.12155729	0.151298328	197.3435942	9303.414382
13	0.55	3.75323033	0.002141747	0.079914397	0.207802356	300.1781076	6773.701064
14	0.6	3.802008064	0.002136793	0.155412133	0.164108609	154.3544388	8577.191948
15	0.65	3.83682805	0.002172069	0.097248385	0.161853841	246.6730188	8696.679892
16	0.7	3.855158806	0.002190222	0.019081755	0.086890711	1257.146035	16199.55725
17	0.75	3.9295541	0.002193411	0.101016409	0.142720971	237.4718416	9862.538288
18	0.8	3.995159919	0.002275834	0.158287674	0.158340965	151.5503509	8889.620184
19	0.85	4.013641155	0.002383425	0.011382786	0.094546368	2107.441164	14887.83839
20	0.9	4.10814595	0.002582714	0.14343172	0.145029891	167.2471937	9705.523681
21	0.95	4.148569874	0.002848653	0.110425718	0.106767556	217.2370074	13183.6964
22	1	4.192	0.002858747	0.150345253	0.165755128	159.5564349	8491.99089

Post-processing

How to collect the parameters (results), and is there any visualisation of the parameters?

- If T and/or C-rate dependences are considered in parameterisation, the parameter look-up table and plots will be 3D.
- Below is an example considering 2 Ts and 4 C-rates.

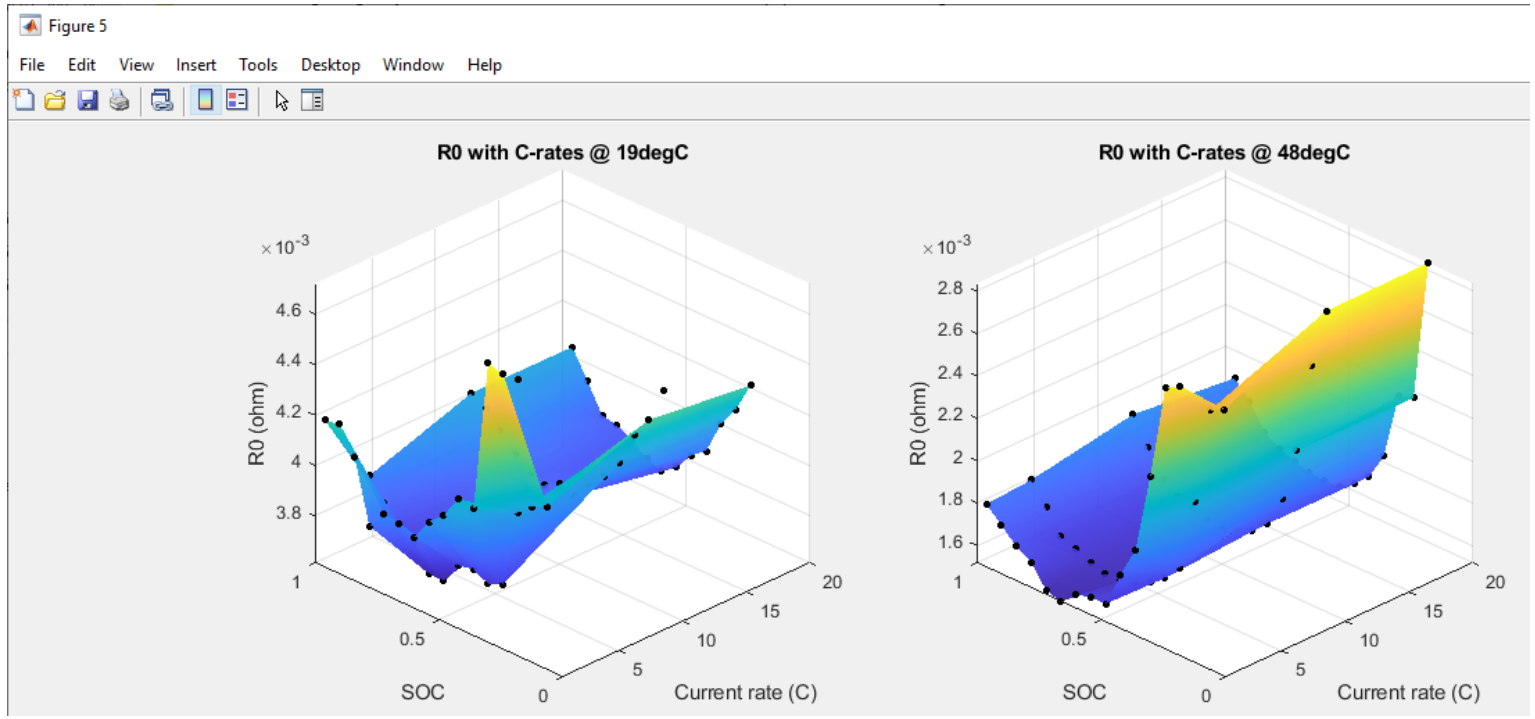
AutoSave Off — LUT_R0_with_C-rates @ All Temperatures.x

File Home Insert Page Layout Formulas Data Review View

N12

	A	B	C	D	E	F
1		0.5	4	12	20	C-rate
2	0.18	0.00471069		0.004403765		
3	0.24	0.004707222	0.004099053	0.004258733	0.004208722	
4	0.3	0.004722059	0.004073176	0.004172267	0.004087097	
5	0.36	0.004116812	0.004020645	0.004034859	0.004000747	
6	0.42	0.00413088	0.003709968	0.003951974	0.003867397	
7	0.48	0.004036377	0.003685159	0.003886407	0.00382286	
8	0.54	0.003985042	0.003714667	0.003822526	0.003747966	
9	0.6	0.003897687	0.00370372	0.00384206	0.003705255	
10	0.66	0.003925502	0.003613646	0.003812046	0.003729676	
11	0.72	0.00393618	0.003616478	0.003820997	0.003775281	
12	0.78	0.003859098	0.003709972	0.003875167	0.003805096	
13	0.84	0.004106669	0.003695621	0.003940855	0.003820261	
14	0.9	0.004209596	0.003813033	0.004005596	0.003928206	
15	0.96	0.004197509	0.003896435	0.004036485	0.00403548	
16	SOC					

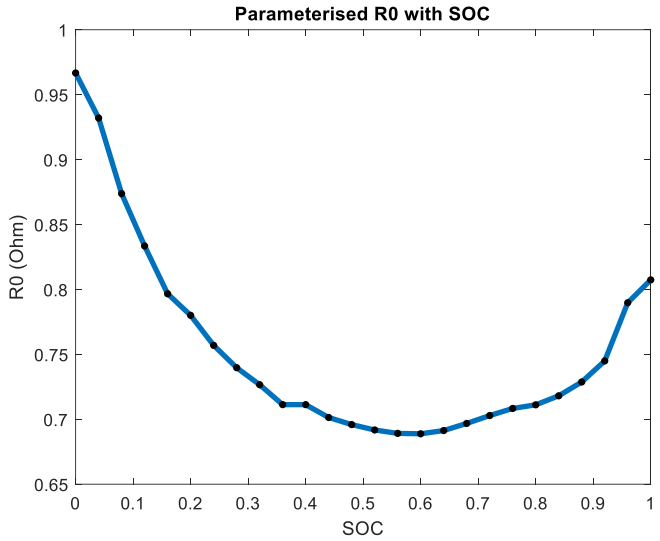
19degC 48degC

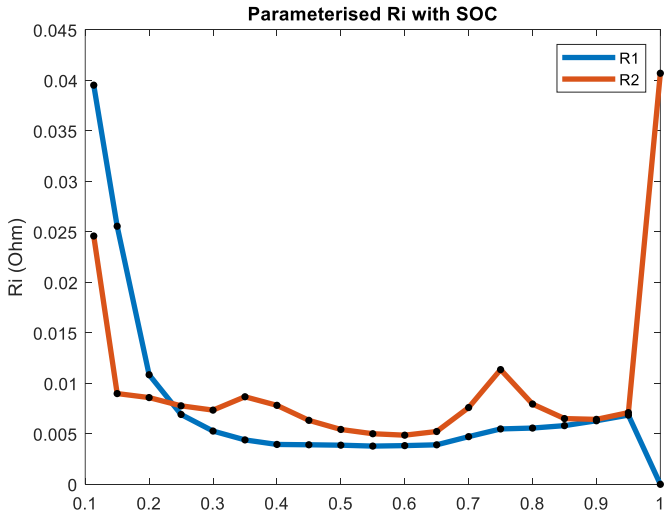


Result check

How to check the quality of the parameters, or shall we be confident with the parameters?

- The first way to check parameter quality is to look at the global comparison plot – does fitting model match well with experiment?
- Also check the plots of OCV and R0:
  - OCV should be strictly non-decreasing with SOC increasing.
  - R0 with SOC should approximately present a 'L' shape.
- The other Rs and Cs can have very random shapes, but there should not be too many spikes.
- You may also check the log of parametersation, as it provides RMSE of fitting as well as warning messages of unusual status happened in parameterisaiton.





Command window log.txt - Notepad

File Edit Format View Help

```
Directing to PUd-cell4-R801-900 - 001.csv
Initialising...
1 continuous segment(s) of DISCHARGE extracted from your datasheet.
SOC range considered in parameterisation is between 0.12944 and 1.
Initialisation done - Paramterisation (with constant tau, without I or T dependence) in progress, including 2 STAGES
--> Each STAGE further includes 19 STEPS.
*********************************************************************************************
--> Proceeding to Stage 1 - solving parameters with VARIABLE tau.
     Step 1 of 19, parameterising at SOC 0.12944
     ======CAUTION======CAUTION======CAUTION======CAUTION======
     Step 1 failed. --> Parameterisation cannot happen at SOC = 0.12944 because of insufficient experimental data.
The codes have to jump over this SOC. If you keep seeing this message, please make the SOC window size larger.
     ======CAUTION======CAUTION======CAUTION======CAUTION======
     Step 2 of 19, parameterising at SOC 0.15
     Step 2 done. Root-mean-square error of voltage is 0.014356
     Step 3 of 19, parameterising at SOC 0.2
```

Thank you !