`

Advance Energy Storage System
ELEC8900-30-R-2022S

University of Windsor

Project Report

on

"Modelling Approaches for Computing Open Circuit Voltage in Li-ion Batteries "

Submitted by

Christino Jacob (110060886)

Submission date: August 15, 2022

Instructor:

Dr. Balakumar Balasingam

`

Table of Contents

`

List of Figures

Abbrevations

OCV – Open Circuit Voltage
SOC – Sales of charge
Li-ion – Lithium-ion

`

1. Abstract

Open Circuit Voltage (OCV) is one of the key factors used to evaluate the battery's state of charge (SOC), which aids in estimating the battery's health and longevity. To compute the OCV-SOC curve and conclude correct OCV modelling, various methods are applied. This assignment involves evaluating four battery cells using various OCV modelling approaches, determining the value of a parameter for modelling, comparing error with given data, ranking the best modelling approach, and improving the accuracy of battery state estimate.

2. Introduction

Battery management system is brain for battery operated equipment, as battery is the primary source of energy for driving it. To increase battery's life and safety, it is essential to monitor the state of charge of the battery in a real time.

SOC of the battery cannot be measured directly while the battery is in operation at a certain time, hence it must be calculated, and several approaches have been presented to estimate SOC. We have SOC and OCV data for four different Li-ion battery cells in this project: C1209, C1210, C1211, and C1212. Plotting and analysing data from an OCV-SOC (open circuit voltage-state of charge) curve for a certain battery cell can be used to explain the battery's properties and compute various parameters for the subsequent modelling approach [2].

- Unnewehr Universal / Linear model
- Shepherd model
- Nernst model
- Combined model
- Combined+3 model
- Polynomial model
- Exponential model

These generally used methods impose or are combined with a nonlinear curve characterization of a Li-ion battery's open circuit voltage (OCV) and state of charge (SOC) [1]. The SOCs for the given batteries are calculated using Coulomb Counting method making use of the current readings taken in each time interval. The generated SOC values are then used to compare modeled OCV-SOC curve date and the data of given OCV-SOC of batteries, then finding the different error, ranking each model, and calculating Borda ranking to decide the best modeling approach [2]. A smart battery management system should be designed using an accurate model and must be reliable, as it provides data for batteries performance and predicting the life.

3. Modelling Approach

3.1 Fetching given data of OCV and generating SOC using Coulomb Counting

The battery data files are imported to the MATLAB workspace using the ‘load’ command. Then the Voltage readings, Current readings and Time intervals are separated assigned as column matrices making use of the structure operations in MATLAB.

For each battery, the SOC generations are done using the coulomb counting method. The seed value for the SOC (s(0)) is taken as 1 and all other SOC values are calculated based on the following equation.

$$s[k+1] = s[k] + \frac{\Delta_k i[k]}{3600\mathrm{C}_{\mathrm{batt}}}$$

Where, Cbatt is the battery capacities given in the instructions *Figure 1* shows the MATLAB code for fetching data of OCV and generating SOC from given file.

```
C1209_data=load('C1209.mat');
I_C1209=[C1209_data.I];
T_C1209=[C1209_data.Time];
OCV_C1209=[C1209_data.V];
C_batt_C1209=2.9765;

%SOC calculation using coulombs counting method
SOC_C1209(1)=1;
for i=2:1:length(OCV_C1209)
    SOC_C1209(i)=SOC_C1209(i-1)+abs((T_C1209(i-1)-T_C1209(i))).*I_C1209
end
SOC_C1209_s=SOC_C1209*(1-2*E)+E; %Scaled SOC
```

Figure 1:MATLAB code for fetching OCV and generating SOC data from given file

3.2 Plotting Curve of OCV vs SOC data of given battery cell

Using plot command to plot OCV vs SOC curve for given battery. *Figure 2* shows command used to plot OCV and SOC data of battery-cell C1209 in subplot-1.

```
figure(1)
sgtitle('OCV-SOC plot for the four cells(Measured)')
subplot(2,2,1)
plot(SOC_C1209,OCV_C1209); % Ploting OCV-SOC curve of the battery C1209
grid on;
title('C1209','OCV v/s SOC')
xlabel('SOC');
ylabel('OCV');
```

Figure 2:MATLAB code for plotting OCV vs SOC curve for given battery

Executing above code for all battery cell which plot OCV-SOC data, with the main title ‘OCV-SOC curve for given Battery-Cell’, and individual title of Battery-Cell number that are C1209, C1210, C1211 and C1212. *Figure 3* is output of MATLAB code command to plot OCV-SOC data for all four battery cells, where OCV data is plotted on Y-axis and SOC data on X-axis.

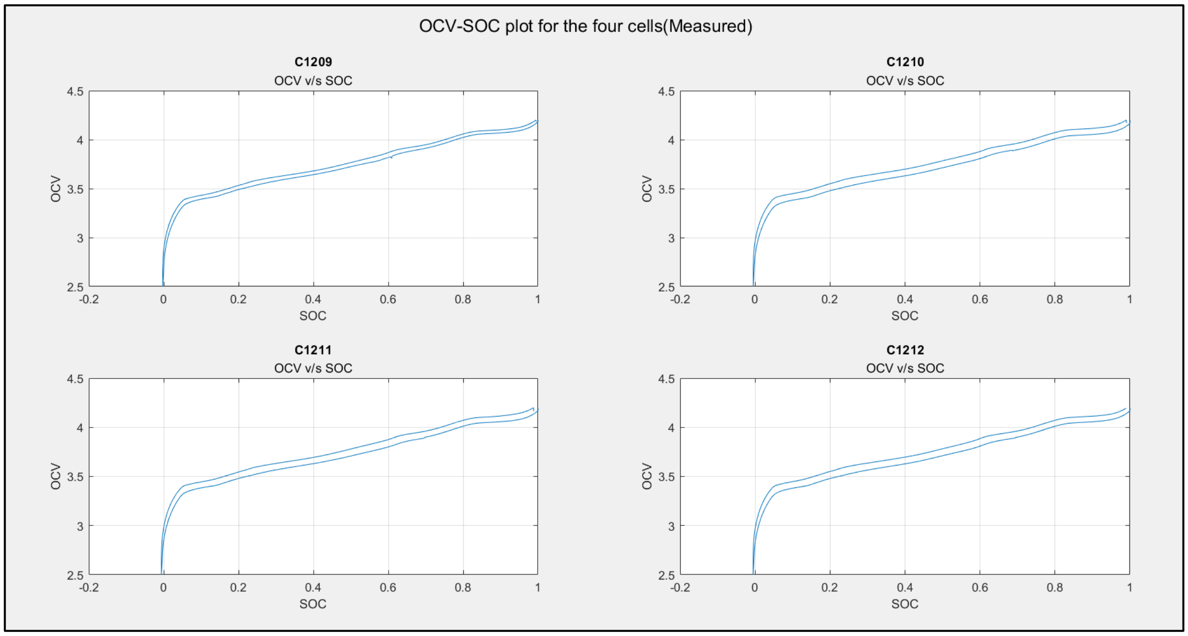

Figure 3:OCV v/s SOC curve for given Batteries

3.3 Observation of Battery Data

Examining data graphs Figure 3 shows that in a Li-ion battery, as the open circuit voltage (OCV) across the terminal decreases, so does the state of charge (SOC), but they are not directly related because the plot computes that the open circuit voltage (OCV) and state of charge (SOC) of a Li-ion battery-cell have non-linear characteristics. Figure 3 also shows that this linear OCV model fulfils the equation [2].

$$v = V_o(s) = p(s)^T\mathbf{k}, \quad (1)$$

where $V_o(s)$ denotes OCV at $s\in[0,1]$ denotes SOC and $p(s)^T$ is a row vector of linear functions s and **k** is the OCV parameter vector [2].

In this project, we are using 7 different linear models listed below [2]:

1. Unnewehr Universal Model (Linear model): $p(s)^T = [1 \quad s]$ (2)
2. Shepherd model: $p(s)^T = [1 \quad \frac{1}{s}]$ (3)
3. Nernst model: $p(s)^T = [1 \quad \ln(s) \quad \ln(1-s)]$ (4)
4. Combined model: $p(s)^T = [1 \quad s \quad \frac{1}{s} \quad s \quad \ln(s) \quad \ln(1-s)]$ (5)
5. Combined+3 model: $p(s)^T = [1 \quad s \quad \frac{1}{s} \quad s \quad \ln(s) \quad \ln(1-s)]$ (6)

6. Polynomial model: $p(s)^T = [1 \quad s \quad \ldots \quad s^m \quad s^{-1} \ldots \quad s^{-n}]$ (7)
7. Exponential model: $p(s)^T = [1 \quad e^s \quad \ldots \quad e^{s^m} \quad e^{-s} \ldots \quad s^{-s^n}]$ (8)

It can be observed that SOC of the battery is range from 0 to 1, if this range in put in the above model, then it will only pass for Linear and Exponential model.

The reason for the model failing is '1/s' and 'ln(1-s)'; if either of these parameters is 0, we cannot compute the actual k parameters. And because the SOC range is 0 to 1, the model fails. To avoid the model failing, we scale the SOC for the remaining five models with a safety margin using the equation below.

Consider safety margin: Epsilon = E = 0.175.

Now new scaled SOC will be SOC_new = SOC x (1 – 2x(E)) + E. (9)

Figure 4 shows MATLAB code to compute new scaled-SOC which is required to compute the k parameters. These parameters are then used to generate plot for modeled OCV and then plot OCV-SOC for all different models.

```
E = 0.175; %Epsilon for Safety Margin
S=0:0.001:1;
S_s=S*(1-2*E)+E; %Scaled SOC
```

Figure 4:MATLAB code for scaling SOC

3.4 Modeled OCV and OCV parameter estimation using different OCV models

In this project, we have 32 numbers of data for each battery cell that is OCV as Vo(s) and SOC as p(s). So, using least square estimation we can compute **k** parameters for 7 different models [2].
General equation of linear OCV model: $V_o(s) = p(s)^T \mathbf{k}$ (10)

Equation of list square estimation of k parameter for above equation is:

$\hat{k}_{LSE} = (p(s)^T p(s))^{-1} p(s)^T V_o(s)$ (11)

Where k will be a (nx1) matrix for given n number of OCV parameters in each model. So, parameters

1. Unnewehr universal model (Linear model) = $\hat{k}_{LSE} = [ko \quad k1]$
2. Sepherd model = $\hat{k}_{LSE} = [ko \quad k1]$
3. Nernst model = $\hat{k}_{LSE} = [ko \quad k1 \quad k2]$
4. Combined model = $\hat{k}_{LSE} = [k0 \quad k1 \quad k2 \quad k3 \quad k4 \quad k5]$
5. Combined+3 model = $\hat{k}_{LSE} = [k0 \quad k1 \quad k2 \quad k3 \quad k4 \quad k5 \quad k6 \quad k7]$
6. Polynomial model = $\hat{k}_{LSE} = [k0 \quad k1 \quad k2 \quad k3 \quad k4 \quad k5]$
7. Exponential model = $\hat{k}_{LSE} = [k0 \quad k1 \quad k2 \quad k3 \quad k4 \quad k5]$

Figure 5 shows the code for finding **k** parameter using List Square Estimation, then re-generating OCV model using **k** parameters and finally sub-plotting curve of all seven models in one single sub-plot under battery-cell title 'C1209'. Firstly, calculating value of **k** parameter using (11). Then using (1), we are calculating OCV of each model for each cell using computed **k** parameter and then plotting curve for modeled OCV vs SOC range of 0:0.001:1 for Unnewehr universal model (Linear Model) and Exponential model; and scaled-SOC range for Shepherd model, Nernst model, Combined model, Combined+3 model and Polynomial model.

```
%Linear model

P_Linear_C1209 = [ones(length(SOC_C1209),1) SOC_C1209']; %Row vector of function of SOC for Linear model for batte
K_Linear_C1209 = inv(P_Linear_C1209'*P_Linear_C1209)*P_Linear_C1209'*OCV_C1209; %OCV parameters vector for Linear m
OCV_model_Linear_C1209 = K_Linear_C1209(1)*ones(size(S')) + K_Linear_C1209(2)*S'; %Computed OCV Linear model for ba
figure(2);
sgtitle('Estimated OCV-SOC of 4 cells - Linear Model')
subplot(2,2,1)
plot(S, OCV_model_Linear_C1209),xlabel('SOC'),ylabel('Estimated OCV');
title('Battery-C1209');

%Shepherd model

P_Shepherd_C1209 = [ones(length(SOC_C1209_s),1) 1./SOC_C1209_s']; %Row vector of function of SOC for Shepherd mode
K_Shepherd_C1209 = inv(P_Shepherd_C1209'*P_Shepherd_C1209)*P_Shepherd_C1209'*OCV_C1209; %OCV parameters vector for
OCV_model_Shepherd_C1209 = K_Shepherd_C1209(1)*ones(size(S_s')) + K_Shepherd_C1209(2)*1./S_s'; %Computed OCV Shephe
figure(3)
sgtitle('Estimated OCV-SOC of 4 cells - Shepherd Model')
subplot(2,2,1)
plot(S_s, OCV_model_Shepherd_C1209),xlabel('SOC'),ylabel('Estimated OCV');
title('Battery-C1209');
```

Figure 5:MATLAB code for plotting all OCV-SOC model

Following figures shows the plot of OCV vs SOC of given data and all computed OCV models for all four battery-cells.

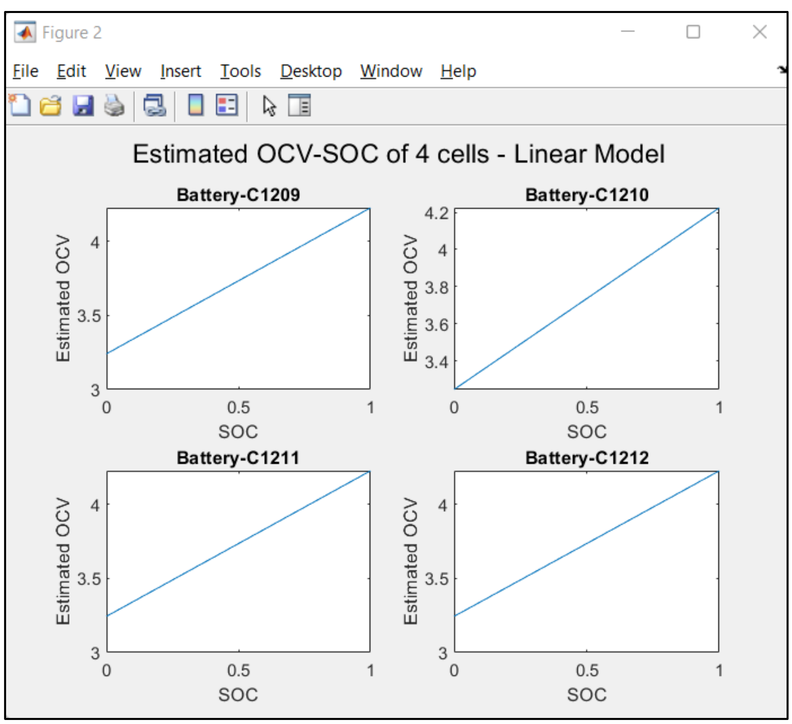

Figure 6:Linear Model

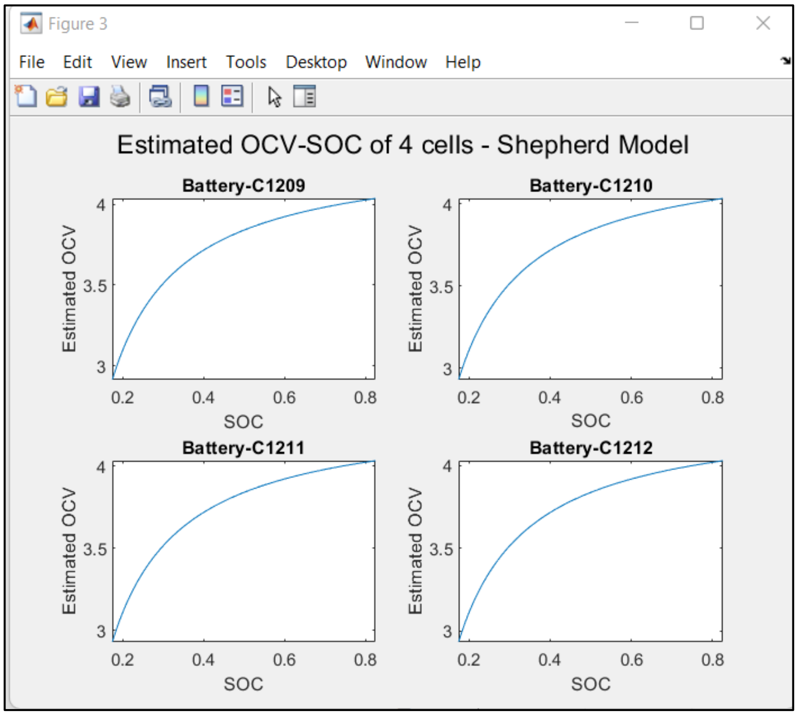

Figure 7:Shepherd Model

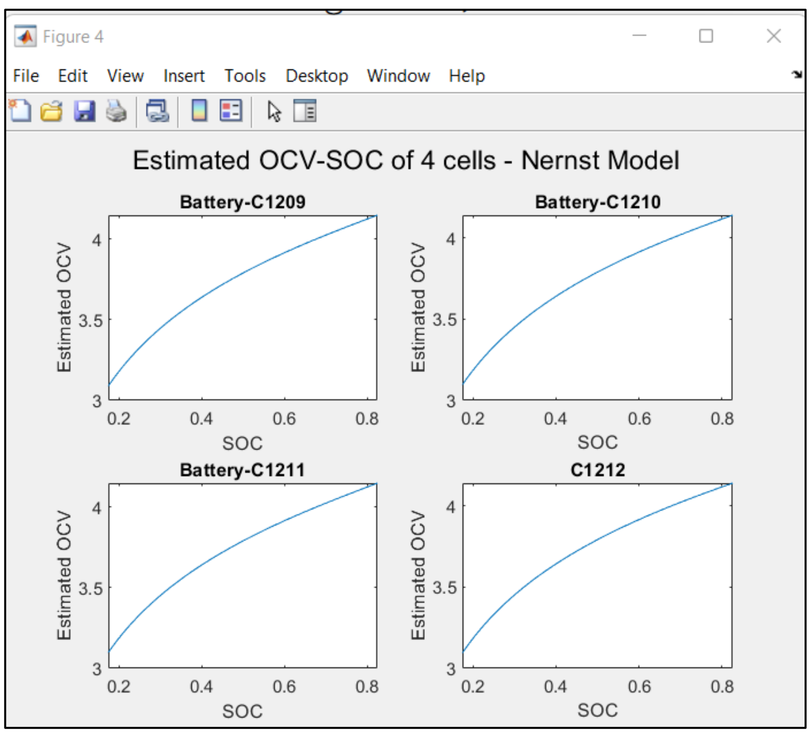

Figure 8:Nernst Model

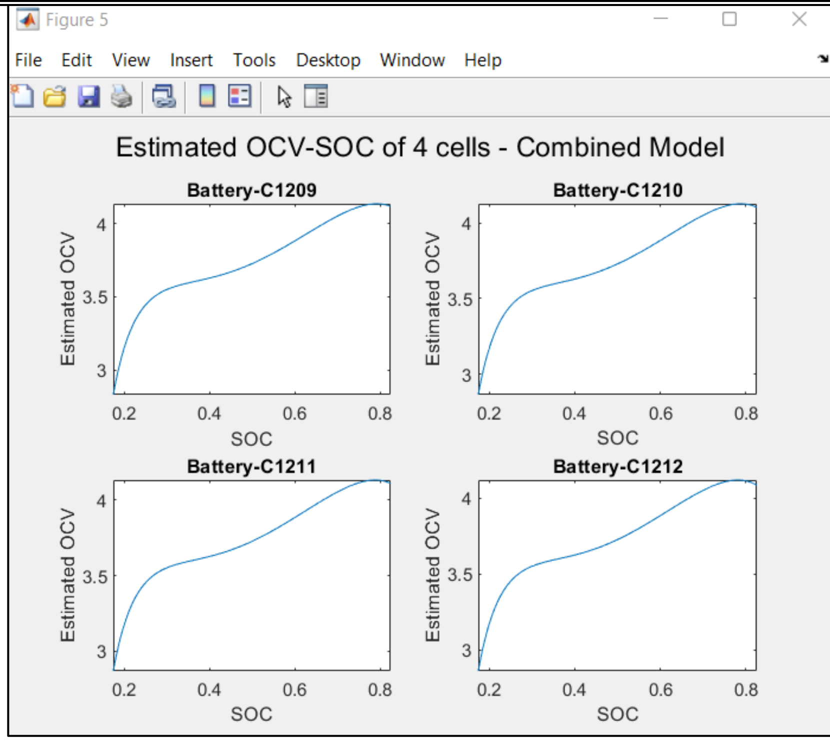

Figure 9:Combined Model

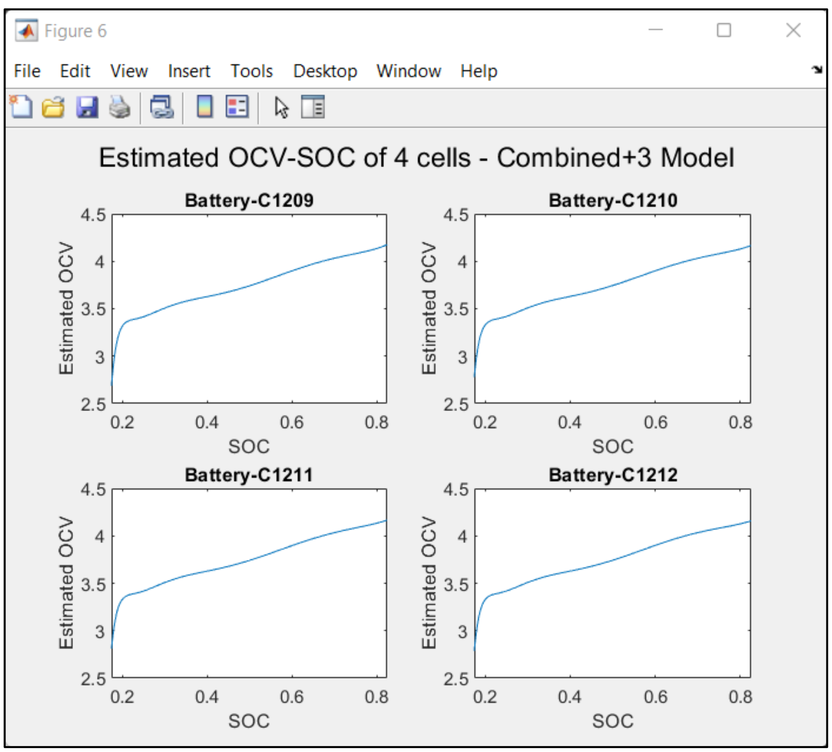

Figure 10:Combined+3 Model

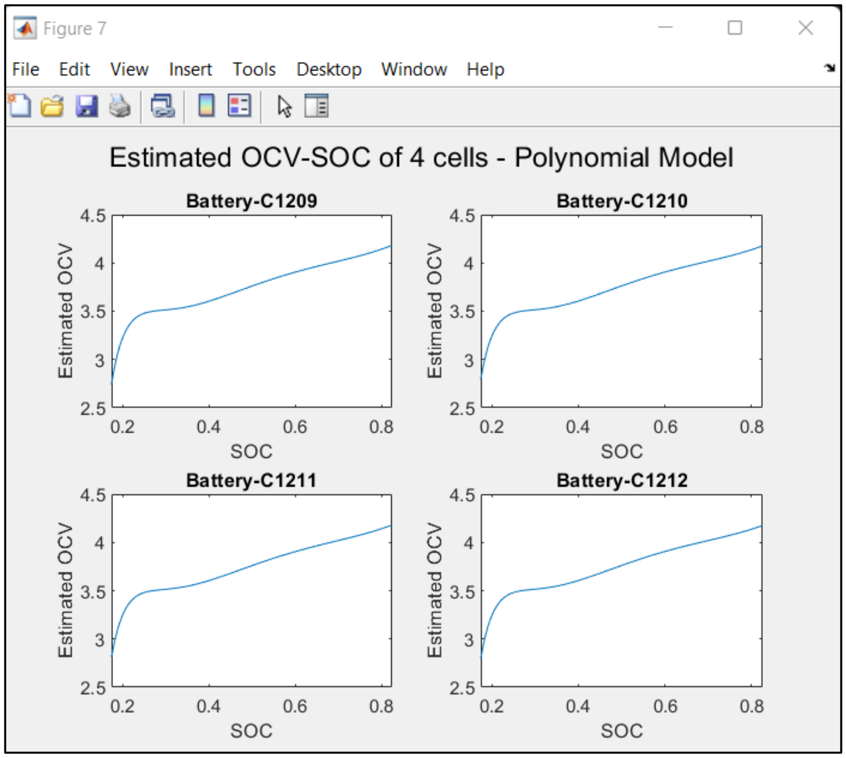

Figure 11:Polynomial Model

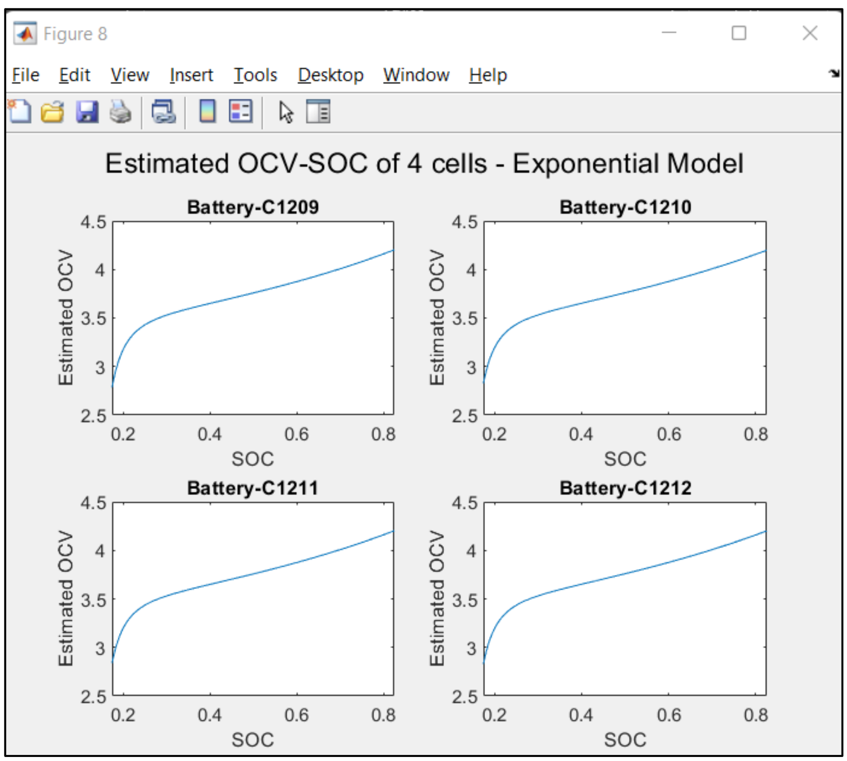

Figure 12:Exponential Model

3.5 Finding error for different model and comparing modeled OCV with OCV of given battery-cell

Finding error between OCV data of battery and OCV of different model and plotting Error vs SOC curve for each model. Figure 13, shows code of error calculation, done by subtracting modeled OCV with OCV of battery C1209. And the plotting the curve for Error vs SOC for individual model and superimposing other modes in same plot.

```
figure(9)
sgtitle('OCV-SOC Error for different OCV models for 4 Cells')
subplot(2,2,1)
Data_error_Linear_C1209 = OCV_C1209 - OCV_model_Linear_C1209_CCSOC;
plot(SOC_C1209,Data_error_Linear_C1209,'DisplayName','Linear OCV model','LineStyle','--'
title('C1209','Error v/s SOC')
xlabel('SOC');
ylabel('Modeling Error');
grid on
hold on
```

Figure 13:MATLAB code for plotting OCV-SOC error for different OCV models

Figure 14 depicts the error vs. SOC output for all seven models and all four battery cells. We can see from the plot that Shepherd OCV has greater inaccuracy at SOC=0.5 and takes a positive peak between SOC=0 and SOC=0.1. This figure can be used to predict which model has the most error and which is better at forecasting the OCV-SOC curve. Furthermore, it is obvious that Combined+3 is a very accurate model with a minimal error that is close to zero. However, in order to accurately quantify model performance, a ranking technique is used to determine which model best fits the OCV-SOC curve of given/measured data.

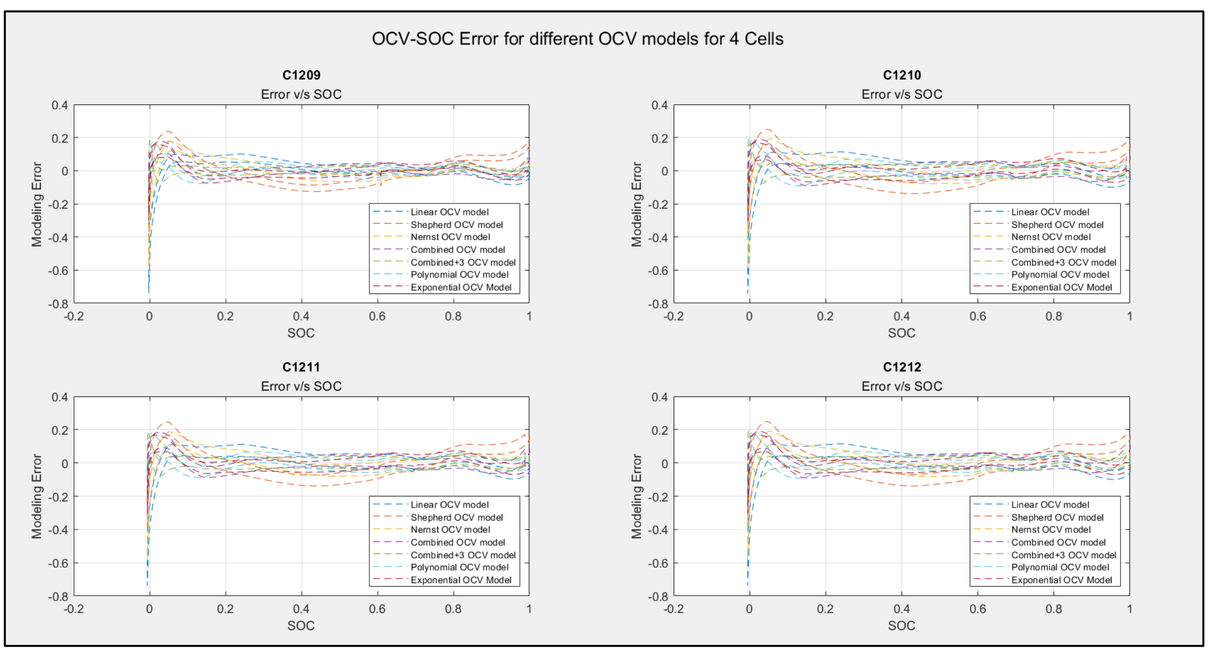

Figure 14:OCV-SOC Error for different models for Batter-Cells

4. Error Prediction and forming Individual Metrices for OCV models

To evaluate accuracy of modeled OCV, different error is calculated for all seven model and added in an individual matrix for each battery-cell. Similarly, making same matrix for another battery-cell. In this paper, different error used for prediction are as follows: [2]

i. Best Fit:

$$\text{BF}(\%) = \left(1 - \frac{\|\hat{v} - v\|}{\|v - \bar{v}\|}\right) \times 100$$

ii. R^2 Fit:

$$R^2(\%) = \left(1 - \frac{\|\hat{v} - v\|^2}{\|v - \bar{v}\|^2}\right) \times 100$$

iii. Max Error:

$$\text{ME} = \max_i \{|v_i - \hat{v}_i|\}$$

iv. Root Mean Square Error:

$$\text{RMS} = \frac{\|v - \hat{v}\|}{\sqrt{N-M}} \text{ or } \sqrt{MSE}$$

N = Number of data points

M = Number of estimated parameter (that is **k** parameters of OCV model)

$\hat{v}$ it the predicted valve of $\bar{v}$ using the estimated parameters, for linear models $\hat{v} = P\hat{k}$ and $\bar{v} = \frac{1}{N} \sum_i^N \hat{v}(i)$.

Model Evaluation Metrics: It's a tradeoff between the number of model parameters and number of data points. If models are fitted using Least Squares, then following analog of Akaike's information criterion (AIC) can be used as a model evaluation metric [2].

$$\text{AIC} = N \ln\left(\frac{S_2}{N}\right) + 2(M + 1)$$

where $S_2 = \sum_{i=1}^{N} {e_i}^2$ and with $e = v - \bar{v}$

In the above S_2 is sum of the square of Errors (SSE), e_i is the i^{th} element of the residual vector e and M it the number of parameters of OCV model. The better the model lower is the AIC [2]. *Figure 15* shows the code to implement above equations in MATLAB.

```
%Error for Linear Model

OCV_model_Linear_C1209_CCSOC= P_Linear_C1209*K_Linear_C1209;
N_Linear_C1209=numel(P_Linear_C1209);
M_Linear_C1209=numel(K_Linear_C1209);
V_bar_Linear_C1209 = (1/N_Linear_C1209)*(sum(OCV_model_Linear_C1209_CCSOC));
BF_Linear_C1209 = (1-(norm(OCV_model_Linear_C1209_CCSOC - OCV_C1209)/norm(OCV_C1209 - V_bar_Linear_C1209)))*100;
RF_Linear_C1209 = (1-(norm(OCV_model_Linear_C1209_CCSOC - OCV_C1209).^2/norm(OCV_C1209 - V_bar_Linear_C1209).^2))*
ME_Linear_C1209 = max(abs(OCV_C1209 - OCV_model_Linear_C1209_CCSOC));
RMS_Linear_C1209 = norm(OCV_C1209 - OCV_model_Linear_C1209_CCSOC) / sqrt(N_Linear_C1209-M_Linear_C1209);
AIC_Linear_C1209 = N_Linear_C1209*log(sum((OCV_C1209 - OCV_model_Linear_C1209_CCSOC).^2)/N_Linear_C1209) + 2*(M_Li
```

Figure 15:MATLAB code to find the error matrices

Error matrices are calculated for all the seven models for each of the 4 batteries- C1209, C1210, C1211 and C1212 and they are stored in table format. These tables are then displayed in the command window of MATLAB. *Figure 16*, *Figure 17*, *Figure 18* and *Figure 19* show the error matrices for batteries C1209, C1210, C1211 and C1212 respectively.

Err_Metrix_C1209_Table =

7×6 table

OCV Model Metrix Table for C1209	AIC	RMSE	R_SQ	BF	Max Error
{'Linear Model' }	-37537	0.079245	99.649	94.077	0.73809
{'Shepherd Model' }	-17030	0.10019	89.868	68.168	0.39913
{'Nernst Model' }	-17634	0.092331	91.398	70.67	0.5825
{'Combined Model' }	-21167	0.057279	96.691	81.81	0.30492
{'Combined+3 Model' }	-26958	0.02619	99.309	91.686	0.18233
{'Polynomial Model' }	-23393	0.042403	98.187	86.536	0.19379
{'Exponential Model'}	-22240	0.049558	97.522	84.26	0.24158

Figure 16:Error Matrix for Battery C1209

Err_Metrix_C1210_Table =

7×6 table

OCV Model Metrix Table for C1210	AIC	RMSE	R_SQ	BF	Max Error
{'Linear Model' }	-36733	0.081155	99.632	93.931	0.74115
{'Shepherd Model' }	-16660	0.10245	89.298	67.287	0.40405
{'Nernst Model' }	-17185	0.095344	90.733	69.558	0.58414
{'Combined Model' }	-20229	0.06287	95.973	79.932	0.31108
{'Combined+3 Model' }	-23892	0.038086	98.523	87.848	0.192
{'Polynomial Model' }	-21852	0.050347	97.418	83.932	0.20323
{'Exponential Model'}	-21049	0.056209	96.78	82.056	0.24825

Figure 17:Error Matrix for battery C1210

Err_Metrix_C1211_Table =

7×6 table

OCV Model Metrix Table for C1211	AIC	RMSE	R_SQ	BF	Max Error
{'Linear Model' }	-36835	0.080477	99.638	93.981	0.73782
{'Shepherd Model' }	-16598	0.1032	89.211	67.154	0.39949
{'Nernst Model' }	-17209	0.094913	90.877	69.796	0.58358
{'Combined Model' }	-20378	0.061503	96.171	80.433	0.30795
{'Combined+3 Model' }	-24184	0.036525	98.651	88.385	0.188
{'Polynomial Model' }	-21987	0.049343	97.536	84.304	0.20273
{'Exponential Model'}	-21209	0.054905	96.948	82.53	0.24507

Figure 18:Error Matrix for battery C1211

```
Err_Metrix_C1212_Table =

  7×6 table
```

OCV Model Metrix Table for C1212	AIC	RMSE	R_SQ	BF	Max Error
{'Linear Model' }	-36412	0.081658	99.626	93.884	0.73887
{'Shepherd Model' }	-16575	0.10217	89.282	67.261	0.40484
{'Nernst Model' }	-17030	0.09596	90.548	69.256	0.58147
{'Combined Model' }	-20158	0.062378	96.008	80.021	0.30597
{'Combined+3 Model' }	-23794	0.037812	98.534	87.894	0.19084
{'Polynomial Model' }	-21712	0.050369	97.398	83.869	0.20158
{'Exponential Model'}	-20899	0.056346	96.742	81.95	0.2456

Figure 19:Error Matrix for battery C1212

5. Ranking Metrics for OCV models

Sorting the best data in a cell with the lowest ranking in the table and constructing the Borda ranking by adding all the data of individual models with the data from the error matrix table [2]. The criteria for sorting the data for each error are listed below.

a. **AIC:** Is **best** when the value is the **lowest**. So, sorting the column in ascending order and ranked 1 for first row and increasing it till rank 6 for last row.
b. **RMSE:** Is **best** when the value is **near to zero**. So, sorting the column in ascending order and ranked 1 for first row and increasing it till rank 6 for last row.
c. **R^2:** Is the **best** when the value is **near to one**. So, sorting the column in descending order and ranked 1 for first row and increasing it till rank 6 for last row.
d. **BF:** Is **best** when value is **near to 100**. So, sorting the column in ascending order and ranked 1 for first row and increasing it till rank 6 for last row.
e. **Max Error**: Is **best** when the value is **near to zero**. So, sorting the column in ascending order and ranked 1 for first row and increasing it till rank 6 for last row.

Computing data with above-described sorting and adding data of all columns in last row to know the Borda ranking. Then sorting Borda ranking row in ascending order to know which is the best rank and best model will come in first row and other in ascending order.

Figure 20, Figure 21, Figure 22 and *Figure 23* shows the Borda ranking for batteries C1209, C1210, C1211 and C1212 respectively.

```
Ranking_C1209_Table =

  7×8 table
```

OCV Model Ranking for C1209	AIC	RMSE	R_SQ	BF	Max Error	Sum	Borda Ranking
{'Combined+3 Model' }	2	1	2	2	1	8	1
{'Polynomial Model' }	3	2	3	3	2	13	2
{'Linear Model' }	1	5	1	1	7	15	3
{'Exponential Model'}	4	3	4	4	3	18	4
{'Combined Model' }	5	4	5	5	4	23	5
{'Nernst Model' }	6	6	6	6	6	30	6
{'Shepherd Model' }	7	7	7	7	5	33	7

Figure 20:Borda ranking for C1209

`

```
Ranking_C1210_Table =

  7×8 table
```

OCV Model Ranking for C1210	AIC	RMSE	R_SQ	BF	Max Error	Sum	Borda Ranking
{'Combined+3 Model' }	2	1	2	2	1	8	1
{'Polynomial Model' }	3	2	3	3	2	13	2
{'Linear Model' }	1	5	1	1	7	15	3
{'Exponential Model'}	4	3	4	4	3	18	4
{'Combined Model' }	5	4	5	5	4	23	5
{'Nernst Model' }	6	6	6	6	6	30	6
{'Shepherd Model' }	7	7	7	7	5	33	7

Figure 21:Borda ranking for C1210

```
Ranking_C1211_Table =

  7×8 table
```

OCV Model Ranking for C1211	AIC	RMSE	R_SQ	BF	Max Error	Sum	Borda Ranking
{'Combined+3 Model' }	2	1	2	2	1	8	1
{'Polynomial Model' }	3	2	3	3	2	13	2
{'Linear Model' }	1	5	1	1	7	15	3
{'Exponential Model'}	4	3	4	4	3	18	4
{'Combined Model' }	5	4	5	5	4	23	5
{'Nernst Model' }	6	6	6	6	6	30	6
{'Shepherd Model' }	7	7	7	7	5	33	7

Figure 22:Borda ranking for C1211

```
Ranking_C1212_Table =

  7×8 table
```

OCV Model Ranking for C1212	AIC	RMSE	R_SQ	BF	Max Error	Sum	Borda Ranking
{'Combined+3 Model' }	2	1	2	2	1	8	1
{'Polynomial Model' }	3	2	3	3	2	13	2
{'Linear Model' }	1	5	1	1	7	15	3
{'Exponential Model'}	4	3	4	4	3	18	4
{'Combined Model' }	5	4	5	5	4	23	5
{'Nernst Model' }	6	6	6	6	6	30	6
{'Shepherd Model' }	7	7	7	7	5	33	7

Figure 23:Borda ranking for C1212

`

6. Observations

Finding the various statistics of inaccuracy, individual metrics, and Borda ranking from the given data of each battery and analysing them by approaching different models. Referring to all the data, it is evident that the Combined+3 model is the most correct for the supplied data of battery-cells.

According to the ranking tables, the Combined+3 model is ranked first, followed by the Polynomial model, the Combined model, the Exponential model, the Nernst model, the Shepherd model, and the Unnewehr Universal model, which is ranked last and is the most inaccurate model (Linear model). This data may also be seen in the model comparison's OCV vs SOC curve and the Error vs SOC curve. Considering all these aspects, Combined+3 model is the best option to choose to model the given set of batteries(Samsung-30T INR21700).

OCV characterization is crucial for a battery management system in-order to calculate various parameters of a battery. Accurate knowledge of the nonlinear relationship between the OCV and the state of charge (SOC) is required for adaptive SOC tracking during battery usage. Battery Fuel Gauges (BFGs) are one of the main sections in battery management systems and it uses these characterization techniques to determine the SOC, time to shutdown, time to full0charge, battery health, etc. which are extensively used in portable electronic gadgets like smartphones. The emergence of battery powered electric vehicles over conventional IC engine powered vehicles has exploited these characterization techniques in developing robust and efficient battery management systems.

For performing real-time SOC estimation in a battery management system which is configured to use Samsung-30T INR21700 batteries, it would be best to take the average of the parameters of all the given cells-C1209, C1210, C1211 and C1212. All these batteries belong to the same series and taking the arithmetic mean of their data will provide us with a more accurate data for real-time SOC estimation.

7. Conclusion

This project focus on the study of finding best and accurate model, and from the observations there is clearly justification that the Combined+3 model should be used to estimate **k** parameter for given cell of Li-ion battery. As OCV vs SOC modeled curve of Combined+3 is showing negligible difference; also, in Error vs SOC curve for Combined+3 model is near to zero only, which means there is negligible error using this method for estimating various parameter of Li-ion battery cells. Hence, smart battery management system must be designed with accurate model like Combine+3 as it is very reliable in estimating SOC of battery with respect to the terminal voltage and as it provides accurate data for predicting battery performance and life.

`

8. References

[1] Zhang, R., Xia, B., Li, B., Cao, L., Lai, Y., Zheng, W., Wang, H., Wang, W., and Wang, M. "A Study on the Open Circuit Voltage and State of Charge Characterization of High-Capacity Lithium-Ion Battery Under Different Temperature." *Energies*, vol.11, no.9, September 2018. [Online]. Available: https://doi.org/10.3390/en11092408 [Accessed: Aug. 10, 2022].

[2] B. Pattipati, B. Balasingam, G.V. Avvari, K.R. Pattipati, and Y. Bar-Shalom, "Open circuit voltage characterization of lithium-ion batteries," *Journal of Power Sources Science Direct*, vol. 269, p. 317-333, December 10, 2014. [Online]. Available: https://doi.org/10.1016/j.jpowsour.2014.06.152 [Accessed: Aug. 10, 2022].