A Study of the Performance Predictive Features of Battery Cells

Rafael Abbariao

Introduction

URBAN ELECTRIC POWER

- UEP is a New York based company manufacturing and deploying high-capacity, rechargeable and inexpensive batteries
- Assures stackable power during outages to provide power for hours to days

Solar Micro Grid

Grid Stabilization

CUNY ENERGY INSTITUTE

Introduction

UEP Cell

- Uses zinc-manganese dioxide (Zn-MnO_2) chemistry as traditional AA batteries
- Can be recharged hundreds of times
- High-capacity, safe, easy to transport, and inexpensive alternative to lithium-ion and lead-acid batteries

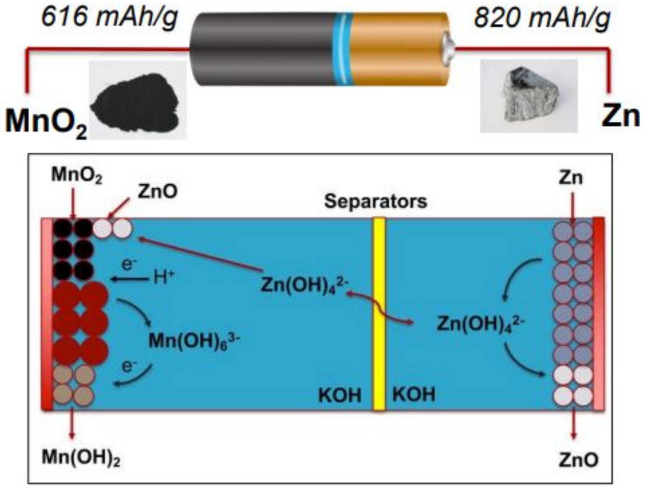
The Monoblock

- Building block for UEP's systems are blocks of four UEP cells in series
- Each monoblock has capacity of 2 kWh energy

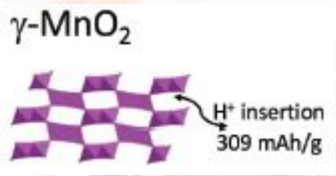
https://www.urbanelectricpower.com/

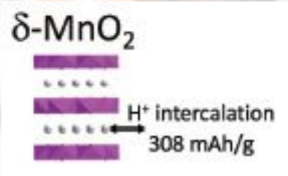
CUNY ENERGY INSTITUTE

Understanding Battery Chemistry

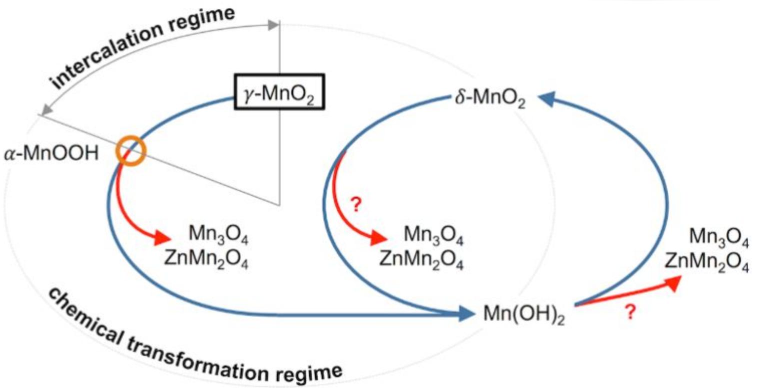


Chemistry:
- Zn and MnO_2 deliver capacities through a two-electron reaction in alkaline electrolyte
- Zn delivers through a dissolution-precipitation reaction
- MnO_2 exhibits proton insertion when EMD is used (γ-MnO_2) but goes through dissolution-precipitation when birnessite is used (δ-MnO_2)



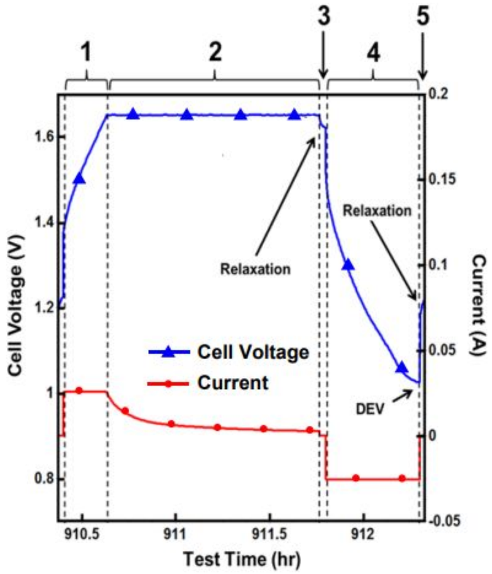


Failure Mechanisms:
- Instability of Mn(III) resulting in formation of irreversible Mn_3O_4
- Zn poisoning forming irreversible $ZnMn_2O_4$



CUNY ENERGY INSTITUTE

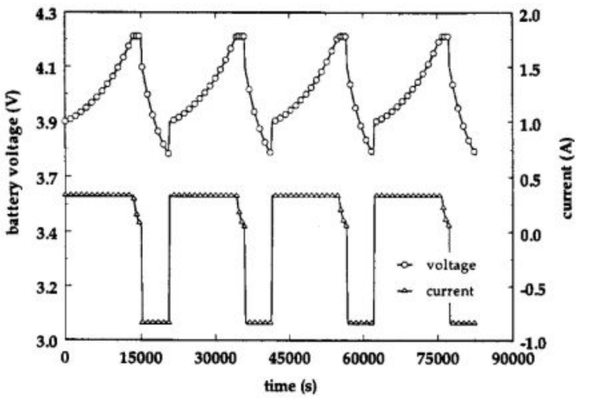
Understanding Battery Specifications



1. Constant current charge
2. Constant voltage charge
3. Rest step
4. Constant current discharge
5. Rest step

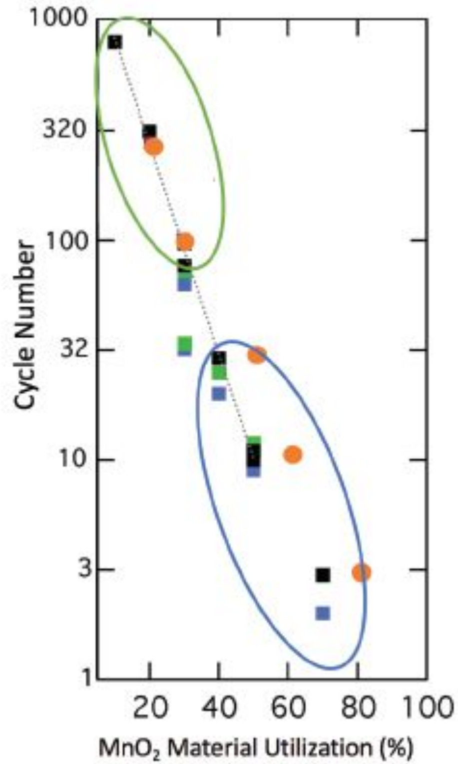
Battery Specifications:

- <u>C-rate</u>: Or discharge current which is a measure of the rate at which a battery is discharged relate to its max capacity
- <u>State of Charge (SOC, %)</u>: Expression of the present battery capacity as a percentage of max capacity
- <u>Depth of Discharge (DOD, %)</u>: Percentage of battery capacity that has been discharged as a percentage of max capacity
- <u>Nominal Capacity</u>: The total Amp-hours available when the battery is discharged at a certain discharge current (C-rate) from 100% SOC to the cut-off voltage
- <u>Cycle life:</u> defined as number of complete charge-discharge cycles performed before nominal capacity falls below 80% initial rated capacity.



CUNY ENERGY INSTITUTE

Understanding Battery Performance and Research Purpose



Number of Cycles as a function of MnO_2 and Zn utilization. The green are represents low MU while the blue represents large capacity

CUNY Energy Institute observed that the cells could:
- Surpass 300 cycles at 50 Ah (or 20% utilization of EMD) for cylindrical cell
- Deliver 150 Ah (or 60% utilization of EMD) where 10 cycles are required

Recall

Cycle life: defined as number of complete charge-discharge cycles performed before nominal capacity falls below 80% initial rated capacity.

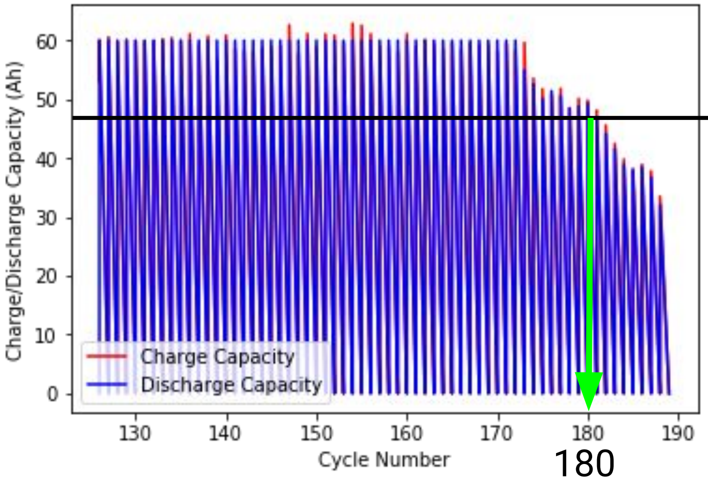
Operating Specs?
Manufacture?
Both?
Neither?

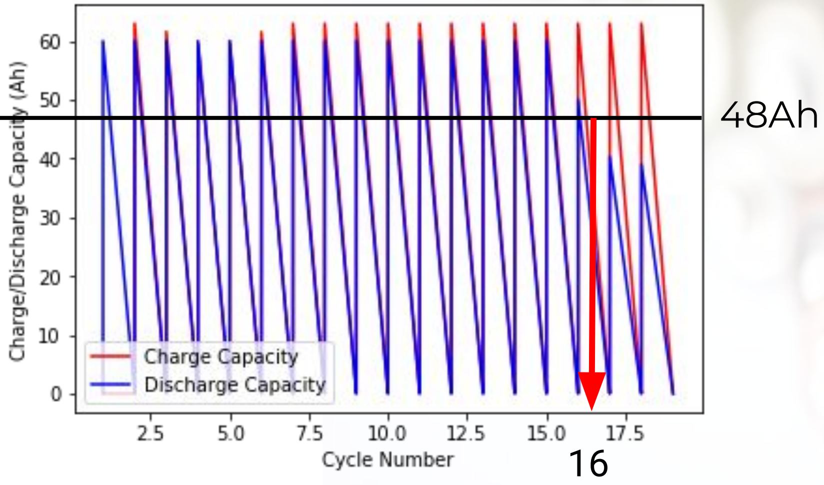
CUNY ENERGY INSTITUTE

Data Processing (for validation)

Time Series Data of a 60Ah cell (30% MU)

	Datapoint Number	Test Time (s)	Cycle Number	Current (A)	Potential (V)	Step Index	Step Time (s)	Charge Capacity (Ah)	Discharge Capacity (Ah)	Charge Energy (Wh)	Discharge Energy (Wh)	Timestamp	dV/dt (V/s)	dQ/dV (Ah/V)
0	1	60.002232	1	0.0	1.511634	1	59.907716	0.0	0.0	0.0	0.0	1525358950000	NaN	NaN
1	2	120.003670	1	0.0	1.511938	1	119.909155	0.0	0.0	0.0	0.0	1525359010000	0.000005	0.0
2	3	180.036363	1	0.0	1.511938	1	179.941848	0.0	0.0	0.0	0.0	1525359070000	0.000000	NaN
3	4	240.053127	1	0.0	1.511634	1	239.958612	0.0	0.0	0.0	0.0	1525359130000	-0.000005	-0.0
4	5	300.085811	1	0.0	1.511634	1	299.991296	0.0	0.0	0.0	0.0	1525359190000	0.000000	NaN



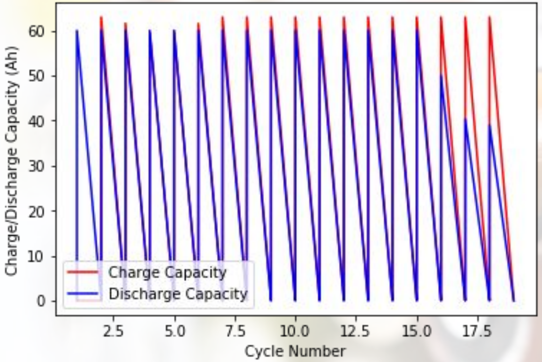


CUNY ENERGY INSTITUTE

Data Processing (continued)

Time Series Data - Processing Script

```
df = pd.read_csv("C:/path/timeseriesfile.csv")
df.head()
X = df.iloc[:, 2].values
y_1 = df.iloc[:, [7]].values     # charge cap
y_2 = df.iloc[:, [8]].values     # discharge cap
Y_1 = plt.plot(X, y_1, color = 'red', label = 'Charge Capacity')
Y_2 = plt.plot(X, y_2, color = 'blue', label = 'Discharge Capacity')
plt.xlabel('Cycle Number')
plt.ylabel('Charge/Discharge Capacity (Ah)')
plt.legend(loc='lower left')
print('Recorded Cycles: %d' % (df.iloc[:, 2].max()))  # print max number of cycles recorded
print('C-rate: %d Ah' % (df.iloc[:, 8].max()))  # print discharge capacity
max_DC = df['Discharge Capacity (Ah)'].max()
abovefade_DC = df['Discharge Capacity (Ah)'] < 0.805 * max_DC
belowfade_DC = df['Discharge Capacity (Ah)'] > 0.795 * max_DC
fade_df = df[abovefade_DC & belowfade_DC]
fade_df.tail(5)
print(fade_df['Cycle Number'].tail(1))
print(fade_df['Discharge Capacity (Ah)'].tail(1))
```



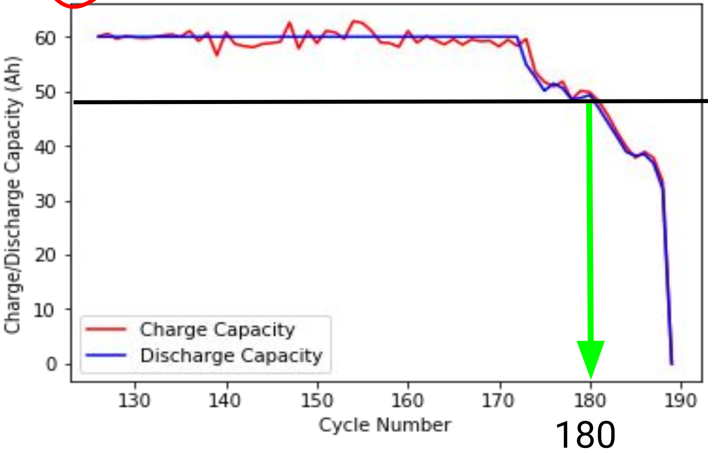

```
Recorded Cycles: 19
C-rate: 60 Ah
```

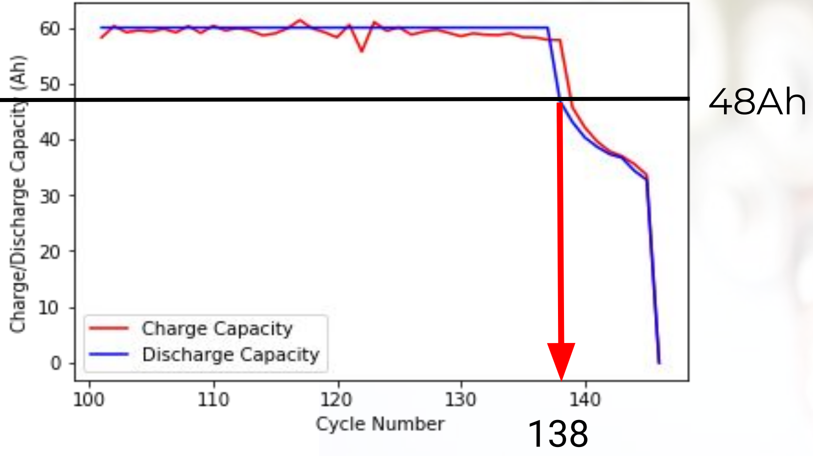
```
40774     16
Name: Cycle Number, dtype: int64
40774     48.293544
Name: Discharge Capacity (Ah), dtype: float64
```

Data Processing (continued)

Cycle Series Data of a 60Ah cell (30% MU)

	Cycle Number	Charge Capacity (Ah)	Discharge Capacity (Ah)	Minimum Test Net Capacity (Ah)	Maximum Test Net Capacity (Ah)	Maximum Cumulative Capacity (Ah)	Charge Energy (Wh)	Discharge Energy (Wh)	Minimum Test Net Energy (Wh)	Maximum Test Net Energy (Wh)	Maximum Cumulative Energy (Wh)	Coulombic Efficiency
0	1	1.788448	55.476014	-1.788448	53.687566	57.264461	3.048885	61.910691	-3.048885	58.861805	64.959576	31.0190871612
1	2	51.645682	57.443367	2.041884	59.485250	166.353510	87.613006	69.007224	-28.751200	58.861805	221.579806	1.11225883349
2	3	67.000002	60.000170	-7.514752	59.485250	293.353683	115.103221	72.471346	-74.847197	40.256024	409.154374	0.895524897151
3	4	63.000324	60.000116	-10.514906	52.485418	416.354123	110.358926	73.534881	-112.734776	-2.375851	593.048180	0.952377896801
4	5	63.000052	60.000081	-13.514842	49.485210	539.354256	110.411489	74.137267	-149.611384	-39.199895	777.596936	0.952381445449



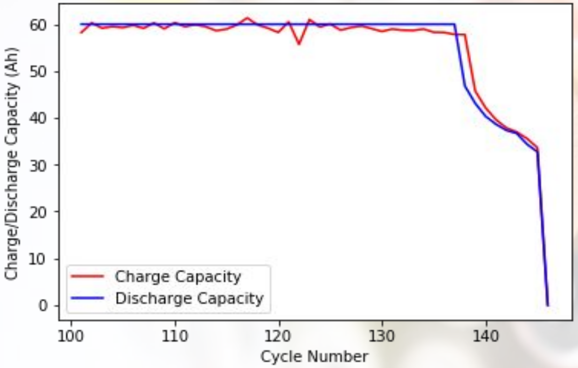


CUNY ENERGY INSTITUTE

Data Processing (continued)

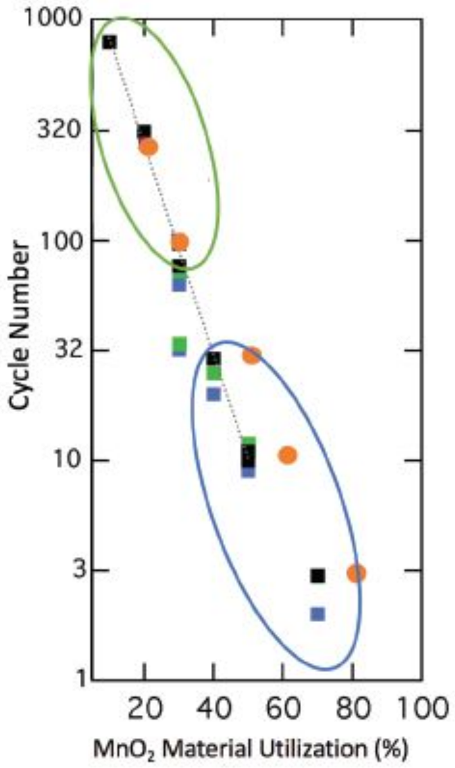
Cycle Series Data - Processing Script

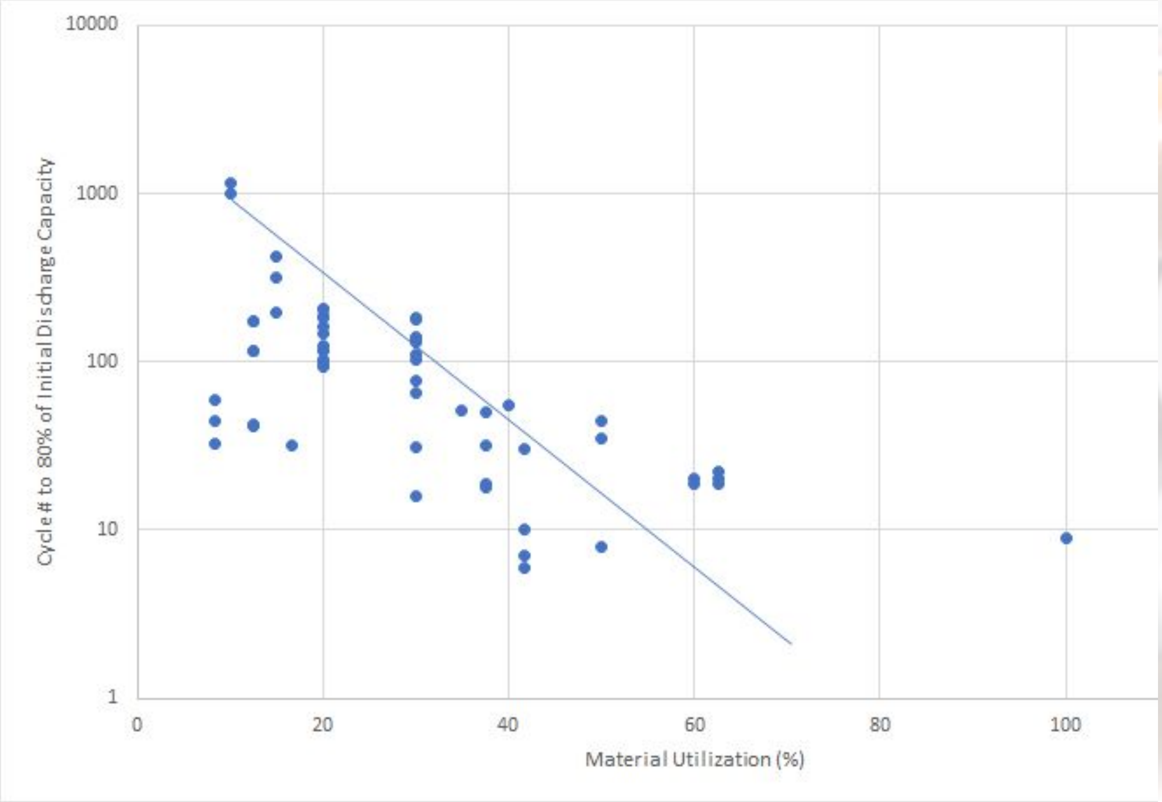
```
df = pd.read_csv("C:/path/cycleseriesfile.csv")
df.head()
X = df.iloc[:, 0].values
y_1 = df.iloc[:, [1]].values     # charge capacity
y_2 = df.iloc[:, [2]].values     # discharge capacity
y1 = plt.plot(X, y_1, color = 'red', label = 'Charge Capacity')
y2 = plt.plot(X, y_2, color = 'blue', label = 'Discharge Capacity')
plt.xlabel('Cycle Number')
plt.ylabel('Charge/Discharge Capacity (Ah)')
plt.legend(loc='lower left')
def fadecycle():
    initial_DC = df.iloc[[1], [2]].values
    for _, row in df.iloc[1:, :].iterrows():
        if row[2] <= 0.8 * initial_DC:
            print(row[0], row[2])
print(fadecycle())
```




```
138 46.81879304
139 43.10925148
140 40.33167306
141 38.64781188
142 37.33765924
143 36.67580096
144 34.37053032
145 32.73379815
146 0.0
```

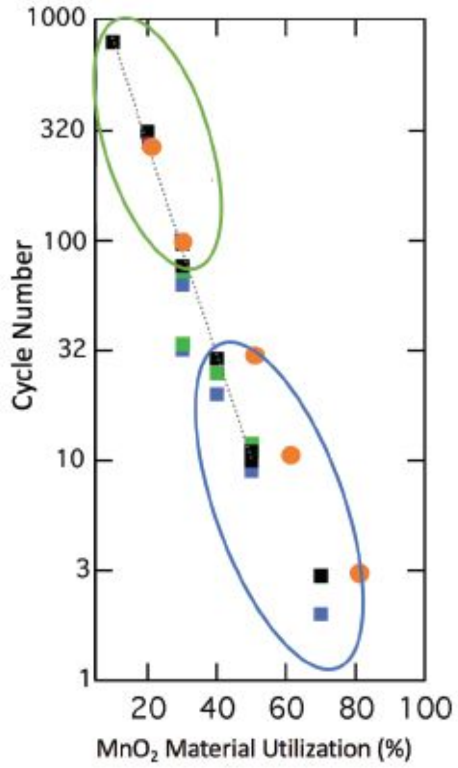
Data Processing (continued)

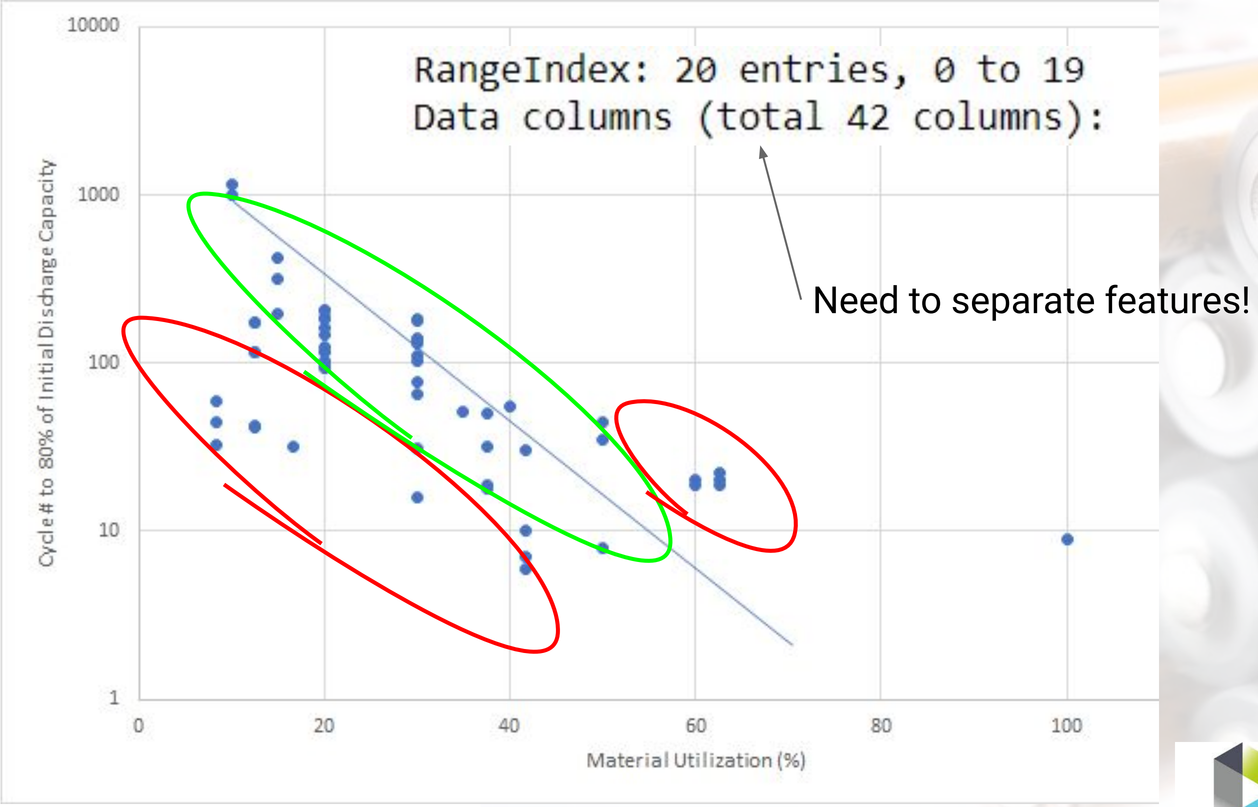




CUNY ENERGY INSTITUTE

Data Processing (continued)





Component Analysis

A cell at 10% MU with a max discharge current of 10A

CUNY ENERGY INSTITUTE

Component Analysis

A cell at 10% MU with a max discharge current of 10A

CUNY ENERGY INSTITUTE

Component Analysis

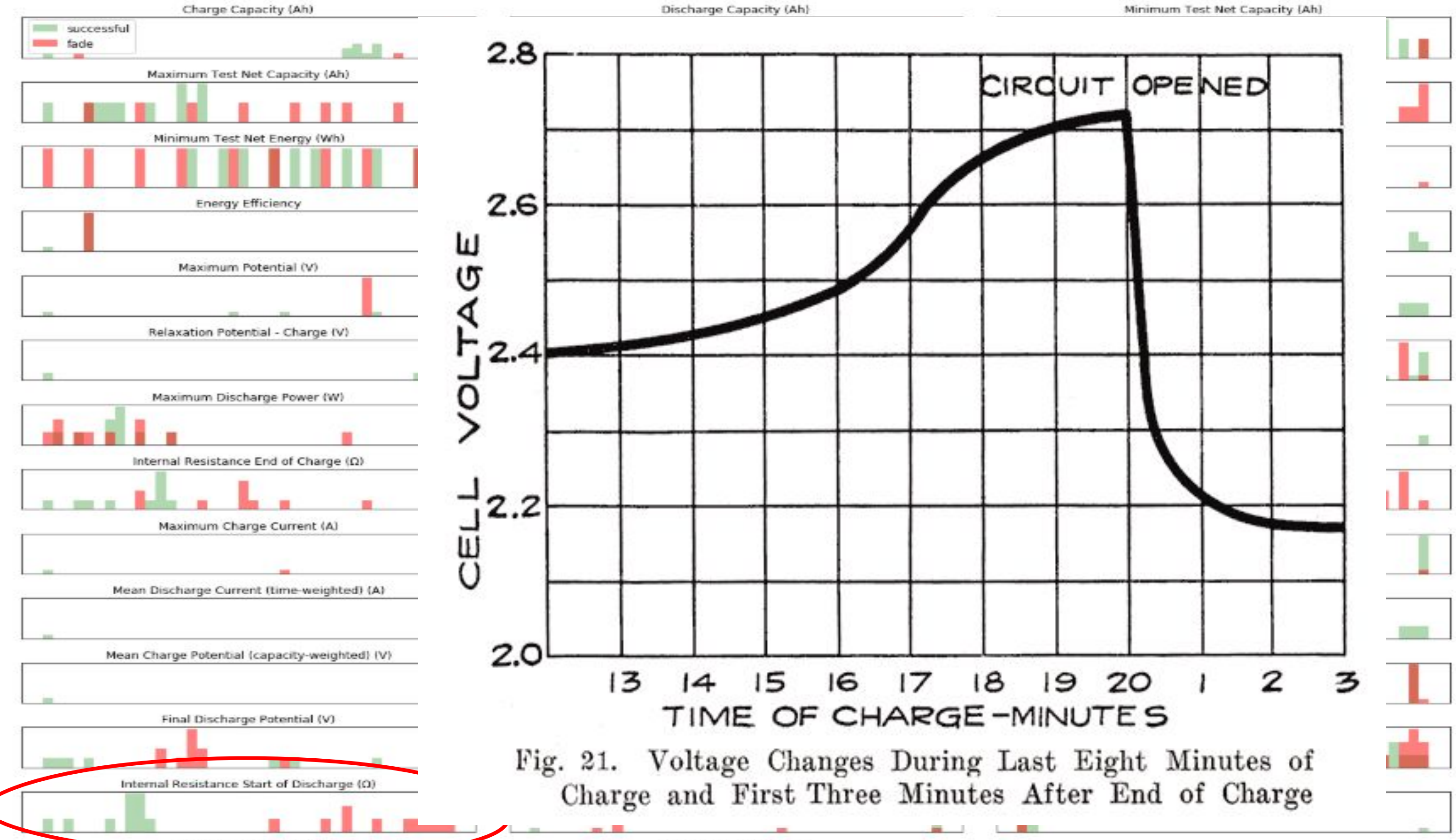


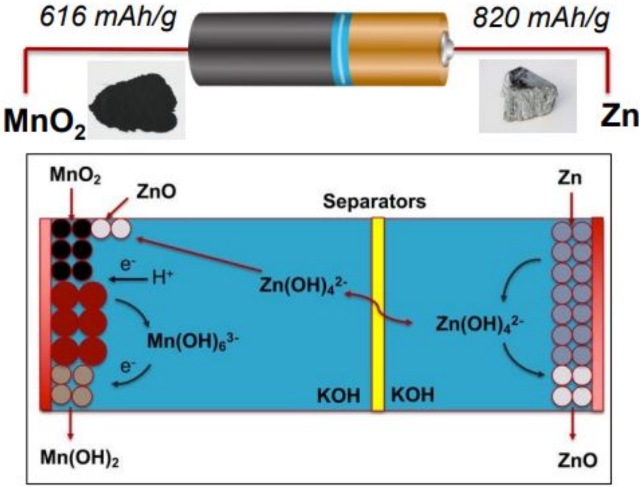
Fig. 21. Voltage Changes During Last Eight Minutes of Charge and First Three Minutes After End of Charge

A cell at 10% MU with a max discharge current of 10A

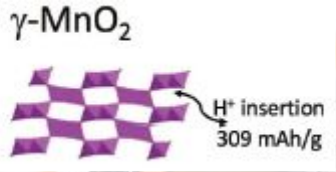
Component Analysis

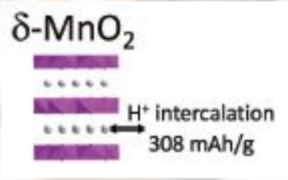
```
successful_df1 = pd.read_csv(“C:/path/file.csv”)
unsuccessful_df1 = pd.read_csv("C:/path/file.csv”)
merged_df = pd.concat([successful_df1.iloc[:10, :], unsuccessful_df1.iloc[:10, :]])
merged_df.replace('None', np.nan, inplace=True)
imputer = SimpleImputer(missing_values = np.nan, strategy = "constant", fill_value = 0)
imputed_df = imputer.fit_transform(merged_df.iloc[:, :])
imputed_df = pd.DataFrame(imputed_df, columns = merged_df.columns)
clean_df = imputed_df[['selectfeatures']]
clean_df = clean_df.convert_objects(convert_numeric=True)
fig, axes = plt.subplots(14, 3, figsize = (20, 16))
ax = axes.ravel()

for i in range(41):
    _, bins = np.histogram(clean_df.iloc[:, i], bins = 50)
    ax[i].hist(clean_df.iloc[:30, i], bins = bins, color = 'g', alpha = 0.3)
    ax[i].hist(clean_df.iloc[30:, i], bins = bins, color = 'r', alpha = 0.5)
    ax[i].set_title(clean_df.columns[i], fontsize = 12)
    ax[i].axes.get_xaxis().set_visible(True)
    ax[i].set_yticks(())

ax[0].legend(['successful', 'fade'], loc = 'best', fontsize = 12)
plt.tight_layout()
plt.show()
```

Understanding Battery Chemistry (Recall)

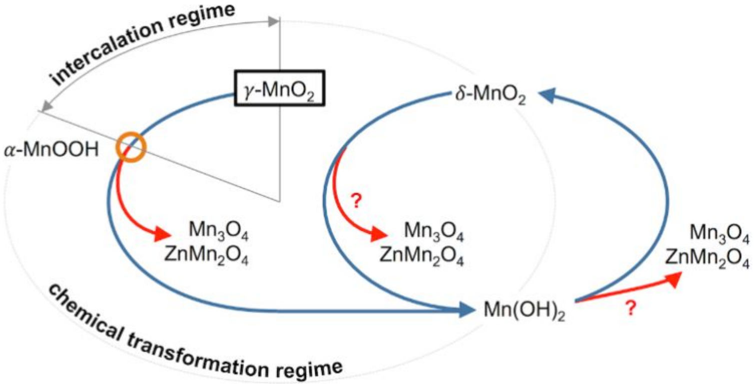


Chemistry:
- Zn and MnO_2 deliver capacities through a two-electron reaction in alkaline electrolyte
- Zn delivers through a dissolution-precipitation reaction
- MnO_2 exhibits proton insertion when EMD is used (γ-MnO_2) but goes through dissolution-precipitation when birnessite is used (δ-MnO_2)



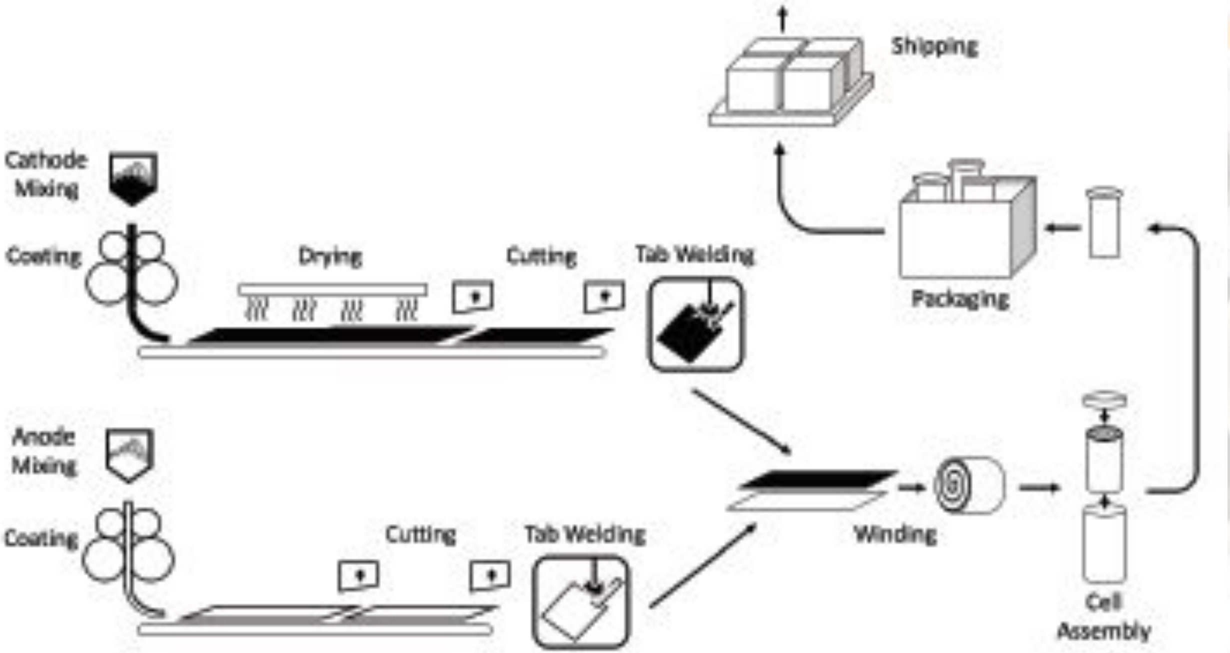


Failure Mechanisms:
- Instability of Mn(III) resulting in formation of irreversible Mn_3O_4
- Zn poisoning forming irreversible $ZnMn_2O_4$



Component Analysis (continued)

Structure of a primary D cell



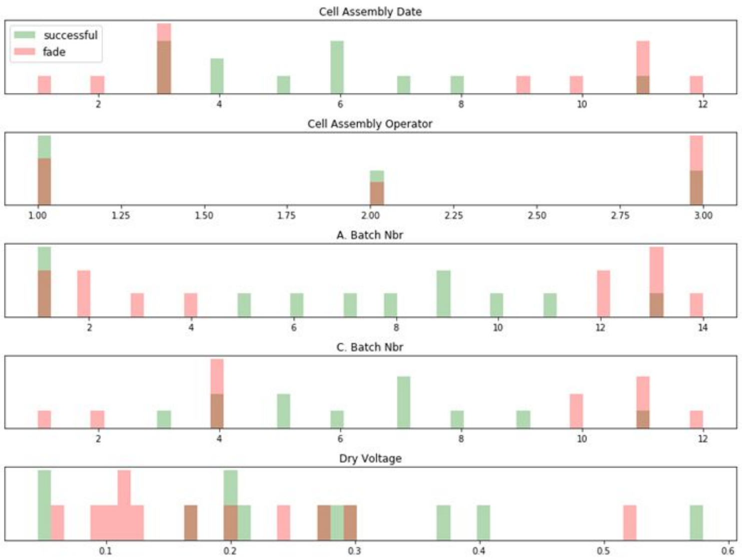
Roll to roll manufacturing process used by UEP

Component Analysis (continued)

	Cell Assembly Date	Cell Assembly Operator	A. Batch Nbr	C. Batch Nbr	Dry Voltage
0	4/26/2018	Franck	A - 040618	C - 042518	0.200
1	4/26/2018	Franck	A - 040618	C - 042518	0.300
2	4/26/2018	Franck	A - 040618	C - 042418	0.270
3	5/10/2018	Franck	A - 050318	C - 050718	0.200
4	5/10/2018	Franck	A - 043018	C - 050718	0.290

↓

	Cell Assembly Date	Cell Assembly Operator	A. Batch Nbr	C. Batch Nbr	Dry Voltage
0	3	1	1	4	0.200
1	3	1	1	4	0.300
2	3	1	1	3	0.270
3	4	1	7	5	0.200
4	4	1	5	5	0.290



CUNY ENERGY INSTITUTE

What's Next?

→ Use a feature based deep learning model to classify the cells to be able to predict battery cycle life before degradation

Data-driven prediction of battery cycle life before capacity degradation

Kristen A. Severson[1,5], Peter M. Attia[2,5], Norman Jin[2], Nicholas Perkins[2], Benben Jiang[1], Zi Yang[2], Michael H. Chen[2], Muratahan Aykol[3], Patrick K. Herring[3], Dimitrios Fraggedakis[1], Martin Z. Bazant[1], Stephen J. Harris[2,4], William C. Chueh[2*] and Richard D. Braatz[1*]

	Classification accuracy (%)		
	Train	Primary test	Secondary test
Variance classifier	82.1	78.6	97.5
Full classifier	97.4	92.7	97.5

Train and primary/secondary test refer to the data used to learn the model and evaluate model performance, respectively.

CUNY ENERGY INSTITUTE

Thank you!

Questions?

CUNY ENERGY INSTITUTE