Leveraging MATLAB-Simulink in Building Battery State-of-Health Estimation Pipelines for Electric Vehicles

Matthew Daigle, Principal Scientist, Artificial Intelligence
Nilesh Kulkarni, Director, Artificial Intelligence

NIO USA

Artificial Intelligence Team Overview

Team

- About 25 full-time employees
- Average experience 12 years

Projects

- Autonomous Driving: Trajectory Planning, Decision Making
- Vehicle AI: Smart Controls, Diagnostics & Prognostics

Motivation

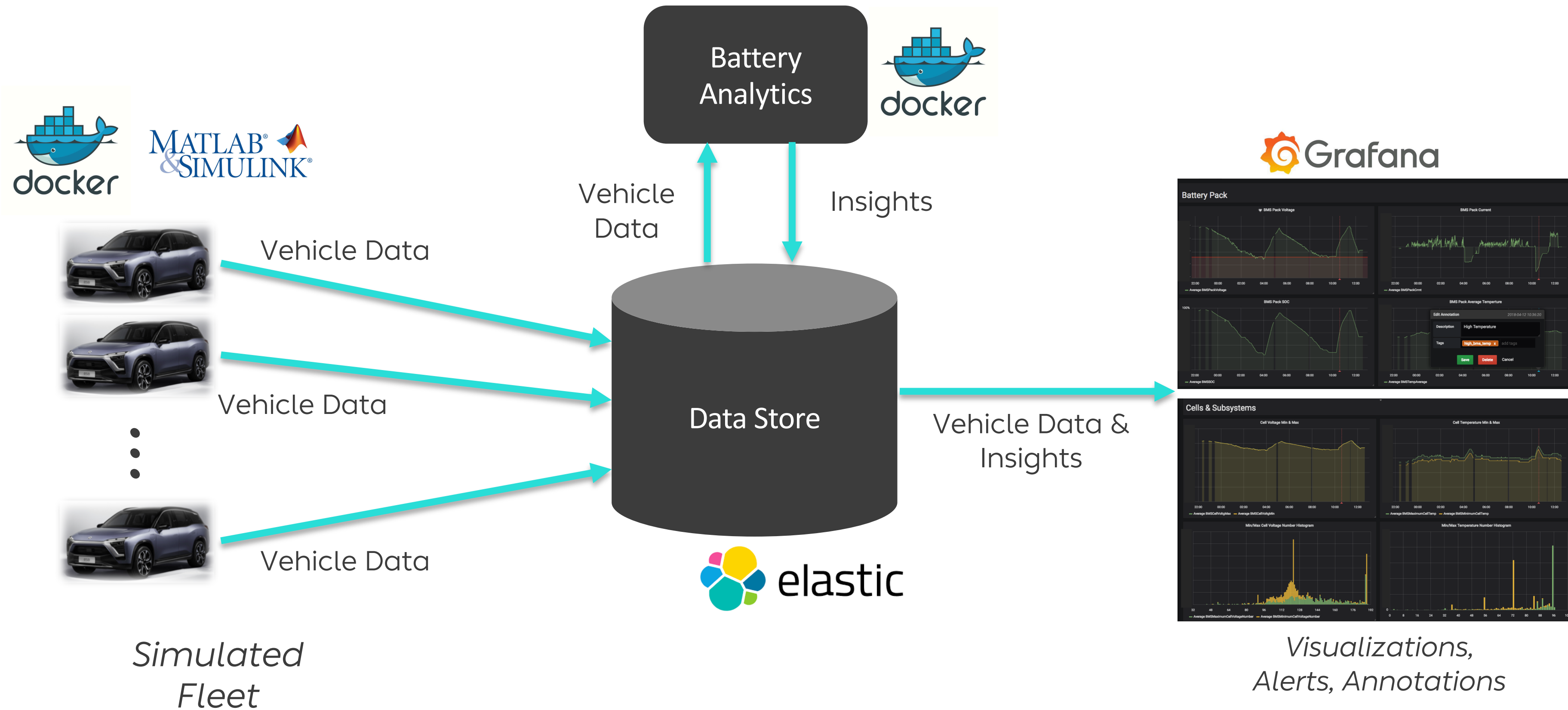
- In electric vehicles, understanding battery State-of-Health (SOH) is critical
  - Powertrain performance
  - Range estimation
  - Fleet management
  - Service operations

Challenges

- In the product design phase, battery data is available only under laboratory and limited driving conditions
  - No existing fleet
  - Limited in-vehicle data collection
  - Data for only specific driving conditions
- However, to build an analytics stack focused on monitoring battery SOH and predicting battery life, we need lots of data

Solution: Scalable simulation-based data generation deployed in the cloud

Cloud-based Architecture



Outline

Vehicle Simulation

- Li-ion Cell Models
- Powertrain
- Vehicle Dynamics
- Driving Conditions

Battery Analytics

- SOH Estimation
- SOH Prediction

Cloud Deployment

- Code Generation
- Simulation as a Service
- Batch Simulations
- Analytics
- Dashboards

NIO

Vehicle Simulator

Vehicle Simulation Summary

Driver Model

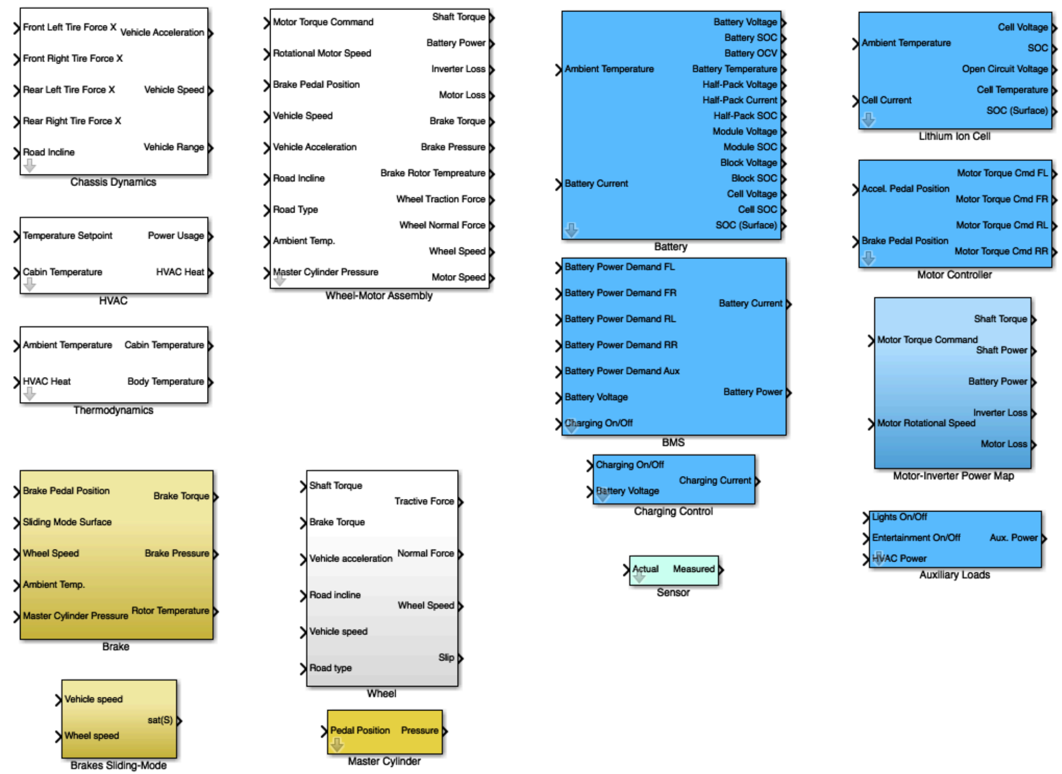
- Inputs
  - Desired waypoint (latitude, longitude)
  - Desired speed
- Outputs
  - Accelerator pedal position
  - Brake pedal position
  - Steering wheel angle

Vehicle Model

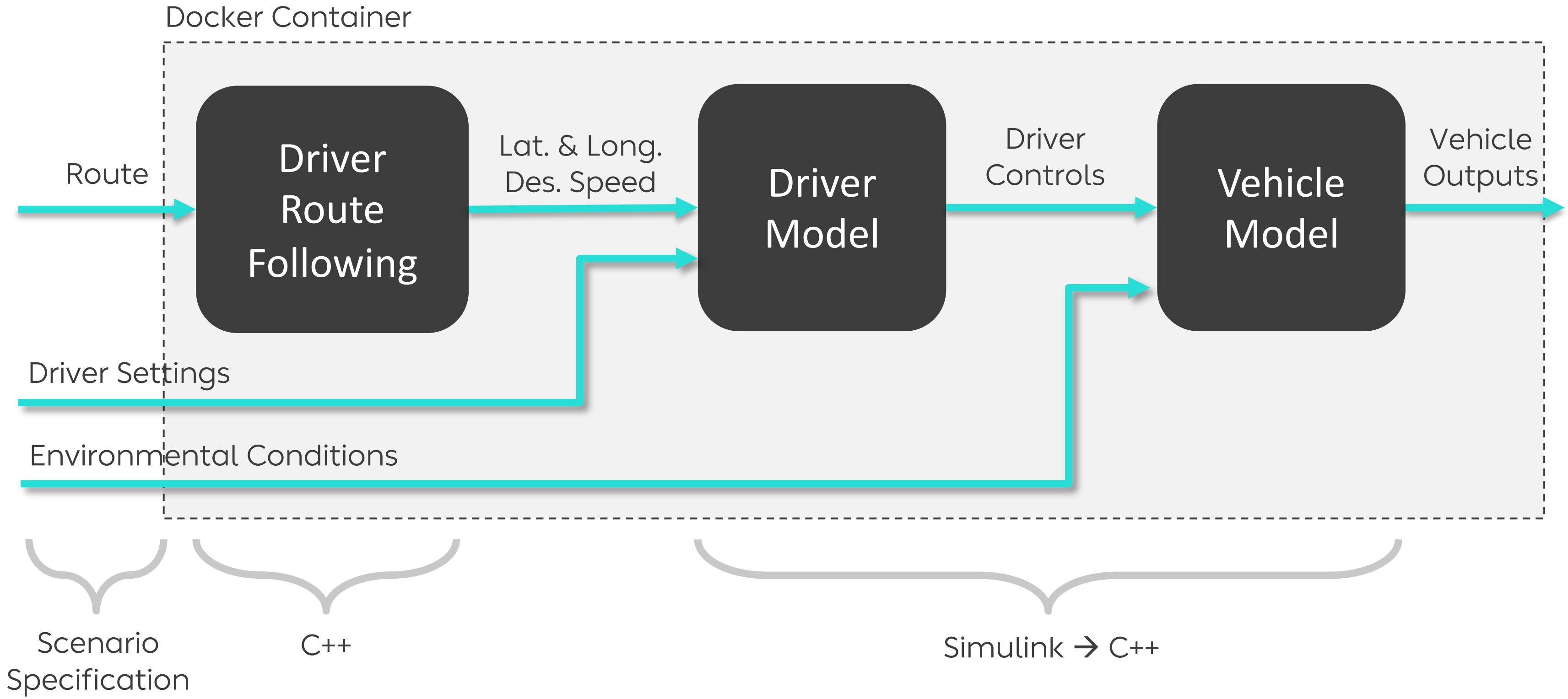
- Inputs
  - Ambient temperature
  - Accelerator pedal position
  - Brake pedal position
  - Steering wheel angle
  - Road type and angle
  - HVAC controls
  - Accessory controls
- Outputs
  - Voltages, currents, temperatures
  - Speed, acceleration
  - Latitude, longitude
  - Etc.

Vehicle Simulation Summary

- Electrical
  - Cell models and battery configuration
  - Cell/battery health degradation
  - Motor and inverter efficiency and losses
  - Torque/current limits
  - Regenerative braking
  - Charging
  - HVAC
  - Thermal management
- Mechanical
  - Drag
  - Road type
  - Brakes
  - Driver model
- Sensors
  - Speed, brake pressures, voltages, currents, etc.
- Fault injection via parameter changes

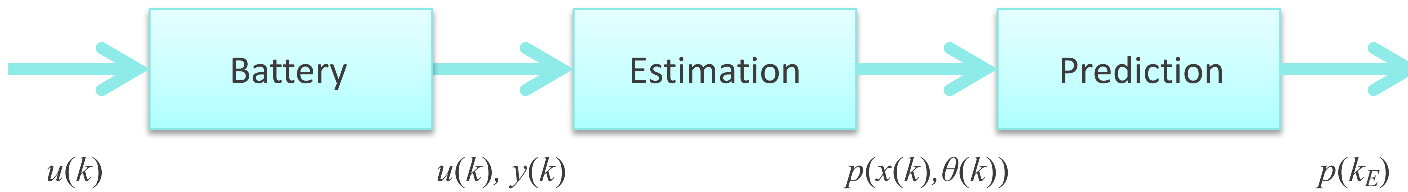


Simulation Architecture



Battery Analytics

Prognostics Architecture



System gets input and produces output

Estimation module estimates the states and parameters, given system inputs and outputs

- Must handle sensor noise
- Must handle process noise

For some event E, e.g., end of discharge (EOD) or end of life (EOL), prediction module predicts k_E

- Must handle state-parameter uncertainty at time of prediction
- Must handle future process noise trajectories
- Must handle future input trajectories (battery loads)

In model-based approaches, require a dynamic model of the battery

Lithium Ion Chemistry

Discharge

Positive electrode is cathode, negative electrode is anode

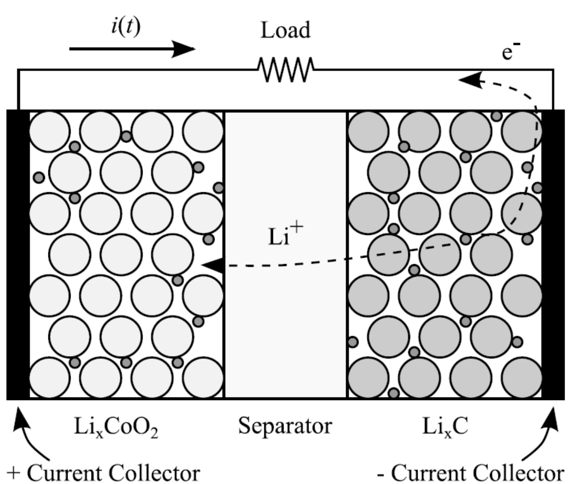
Reduction at pos. electrode:

$Li_{1-n}CoO_2 + nLi^+ + ne^- \rightarrow LiCoO_2$

Oxidation at neg. electrode:

$Li_nC \rightarrow nLi^+ + ne^- + C$

Current flows + to –, electrons flow – to +, lithium ions flow – to +



Charge

Positive electrode is anode, negative electrode is cathode

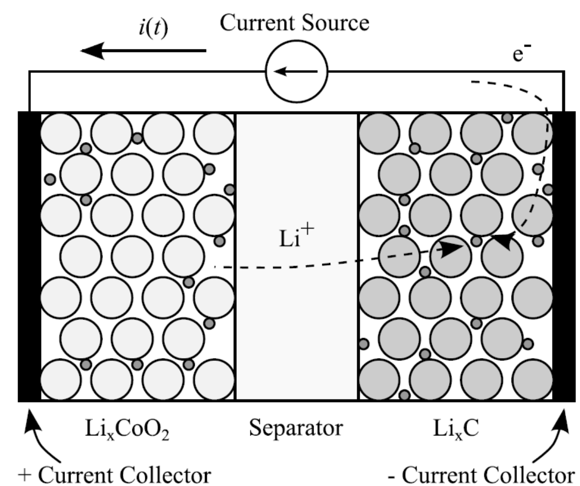
Oxidation at pos. electrode:

$LiCoO_2 \rightarrow Li_{1-n}CoO_2 + nLi^+ + ne^-$

Reduction at neg. electrode:

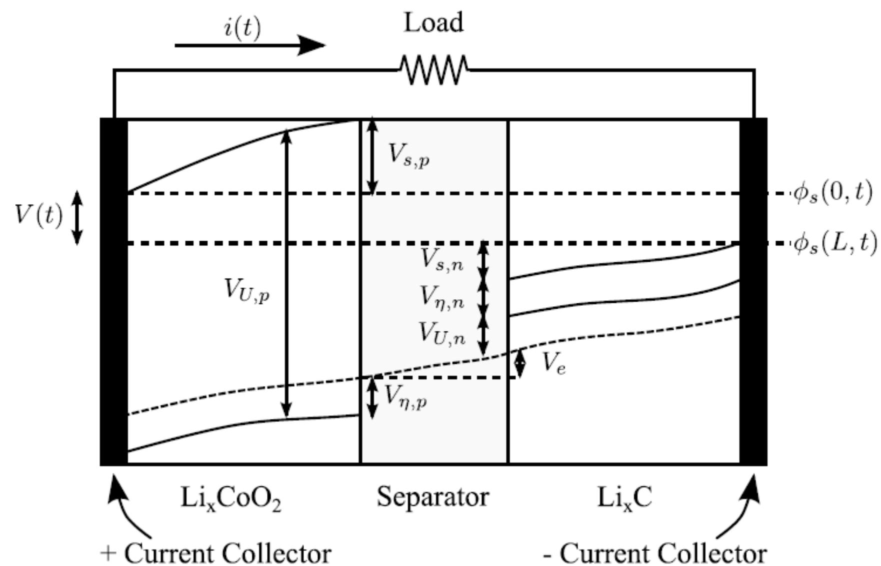
$nLi^+ + ne^- + C \rightarrow Li_nC$

Current flows – to +, electrons flow + to –, lithium ions flow + to –



Cell Discharge Modeling

- Lumped-parameter, ordinary differential equations
- Capture voltage contributions from different sources
  - Equilibrium potential →Nernst equation with Redlich-Kister expansion
  - Concentration overpotential → split electrodes into surface and bulk control volumes
  - Surface overpotential → Butler-Volmer equation applied at surface layers
  - Ohmic overpotential → Constant lumped resistance accounting for current collector resistances, electrolyte resistance, solid-phase ohmic resistances



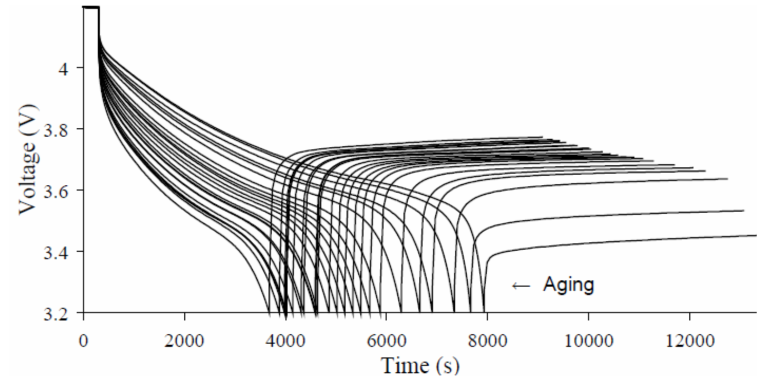
Battery Aging Modeling

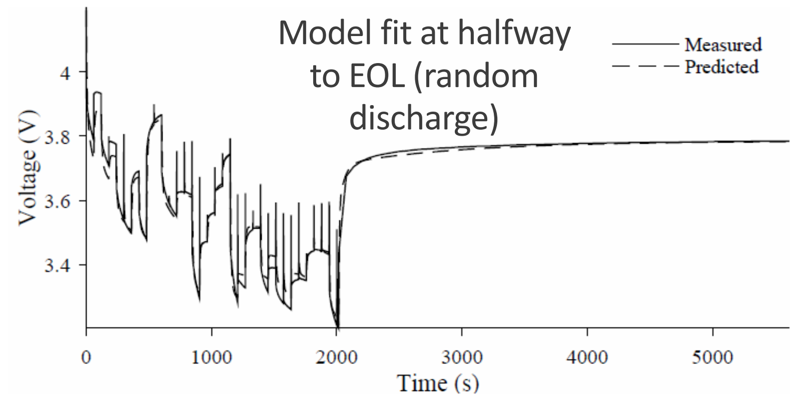
Aging results in two major qualitative effects on dynamics:

- Loss of capacity (due to diffusion stress, irreversible parasitic side reactions)
- Increase in internal resistance (due to solid electrolyte interface layer growth)

Capture with changes in three age-related parameters:

- q_{max} (max available charge)
- R_o (Ohmic resistance)
- D (diffusion rate parameter)

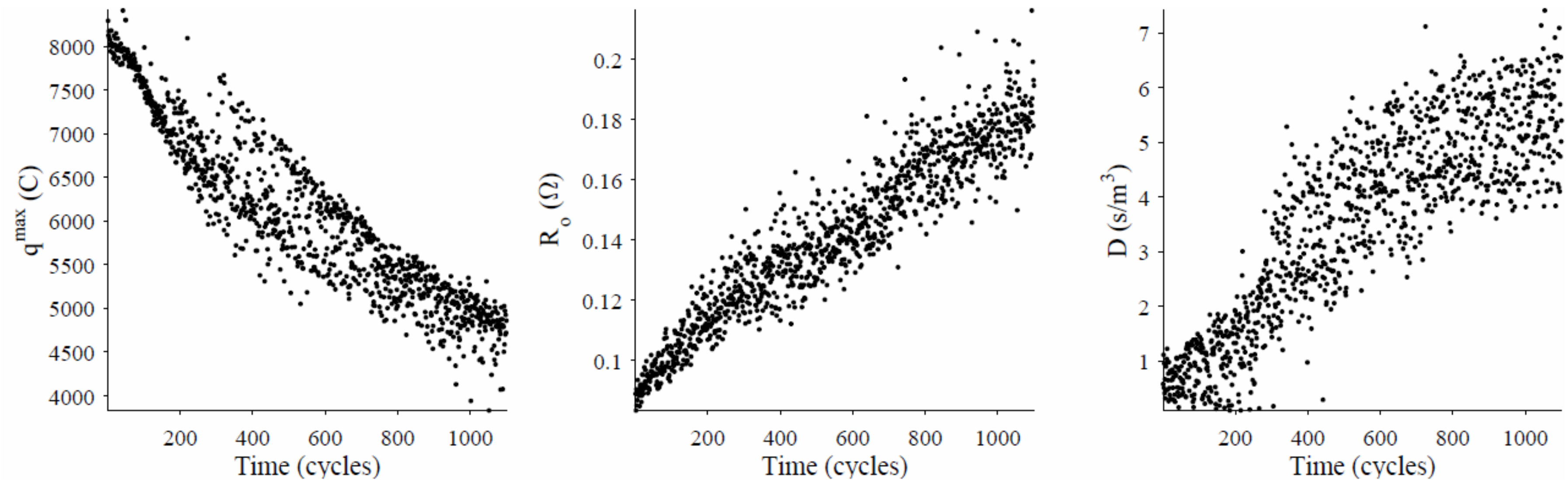




Battery Aging Modeling

Given a discharge cycle, can estimate age-related parameters and determine how they change over time

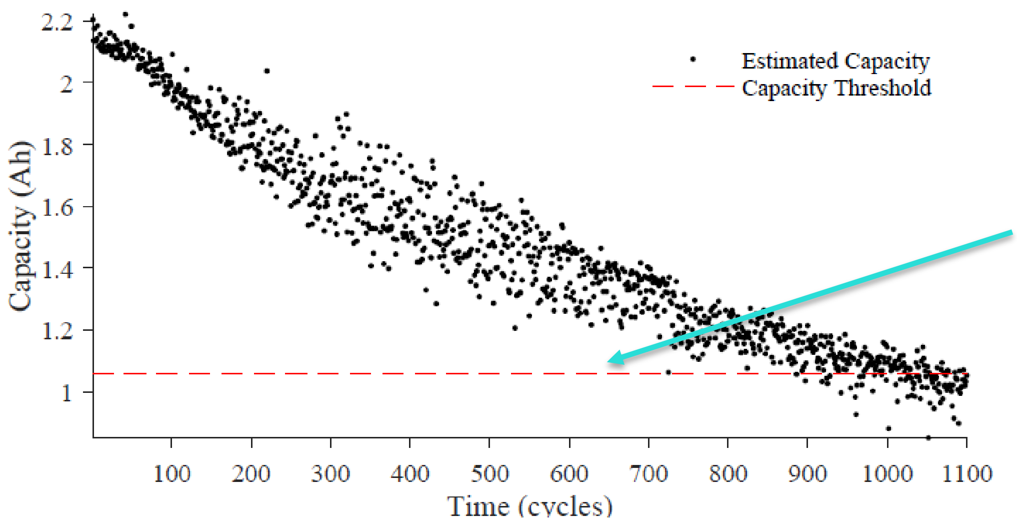
- Assume rate of change of age parameters is of form $w^*|i_{applied}|$
- w is aging rate parameter, $i_{applied}$ is applied current



Defining End of Life (EOL)

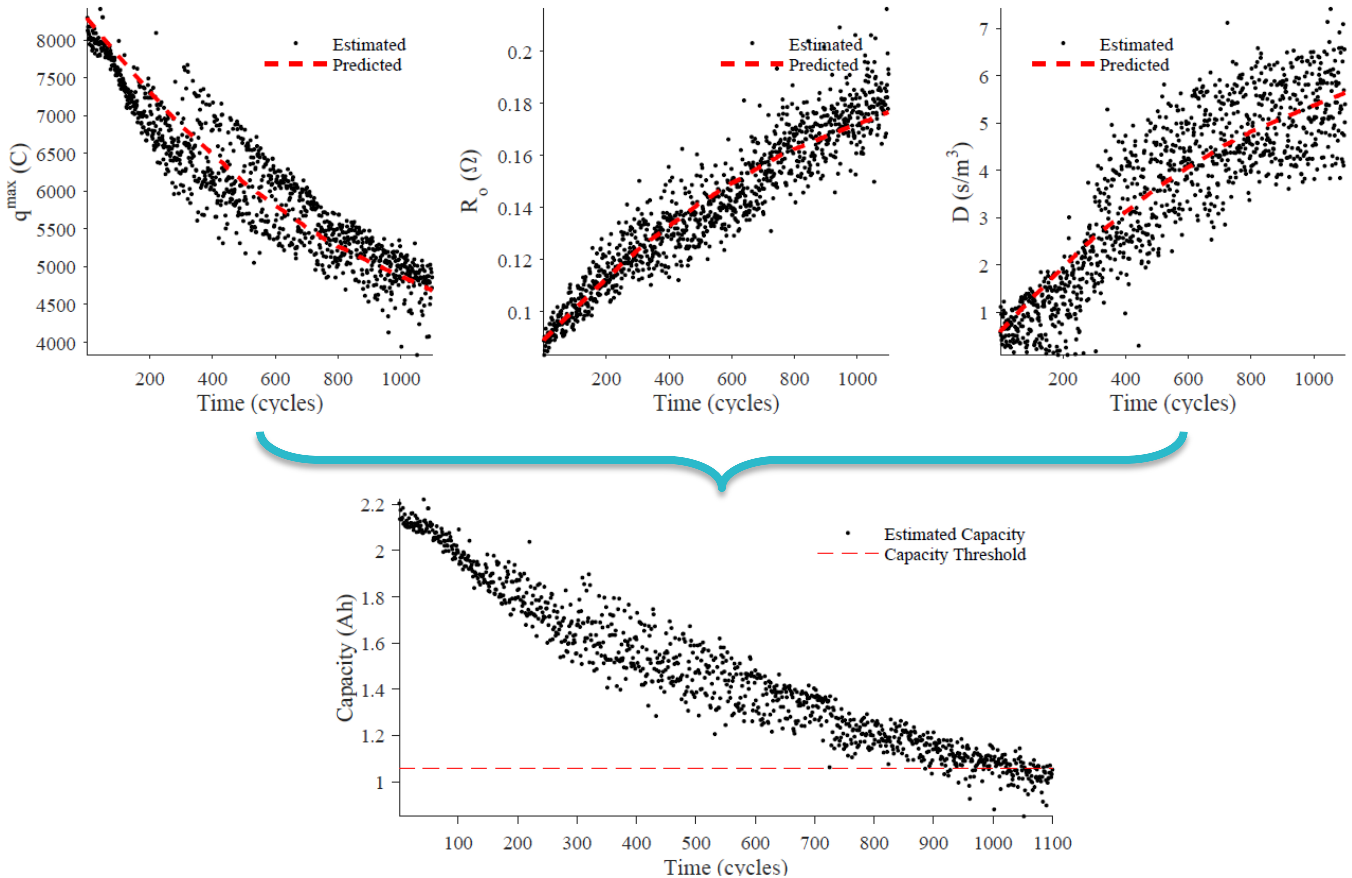
Capacity is measured in Ah for a given discharge cycle

- But, end of discharge (EOD) is dependent on the load, so capacity measurement will be different depending on how battery is used
- Only meaningful to measure capacity w/r/t reference conditions
- For given age parameters, can use model to simulate a reference discharge and compute corresponding capacity



EOL defined as 50% capacity as measured at reference conditions.

State of Health Estimation



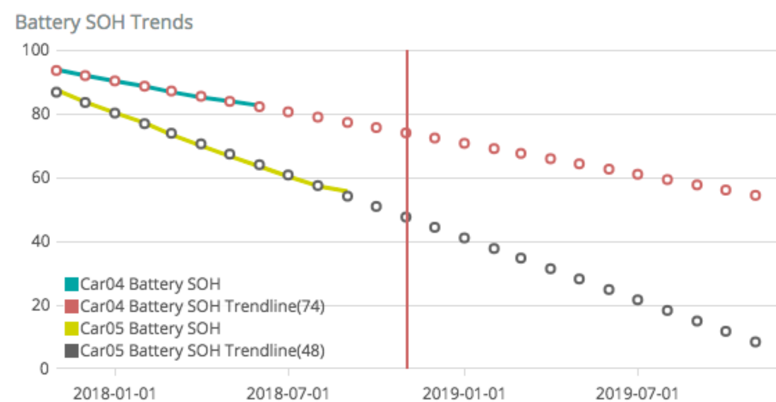
Battery Analytics

Health Estimation

- For each trip, take pack voltage, current, and temperature
- Estimate battery state of charge (SOC) using unscented Kalman filter (UKF)
- Estimate current values of aging parameters over the trip
- Map aging parameters to current battery capacity/state of health (SOH)

State of Health Prediction

- Obtain SOH estimates for all previous trips
- Determine expected future battery loads
- Fit aging model (e.g., linear regression)
- Predict time/miles at which SOH will fall below 80%



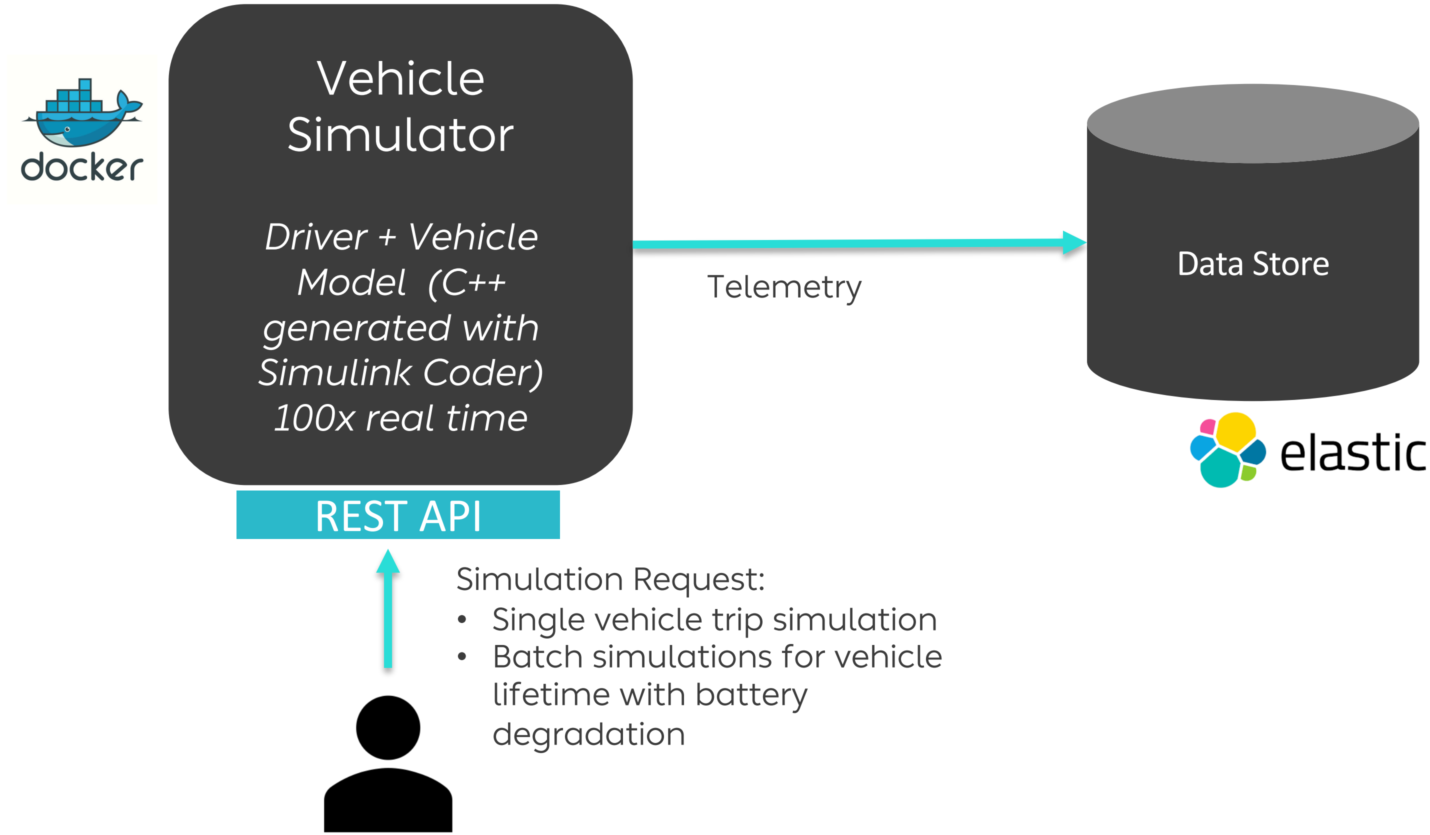
Other Metrics

- Powertrain efficiency, trip energy regenerated, etc.

NIO

Cloud Deployment

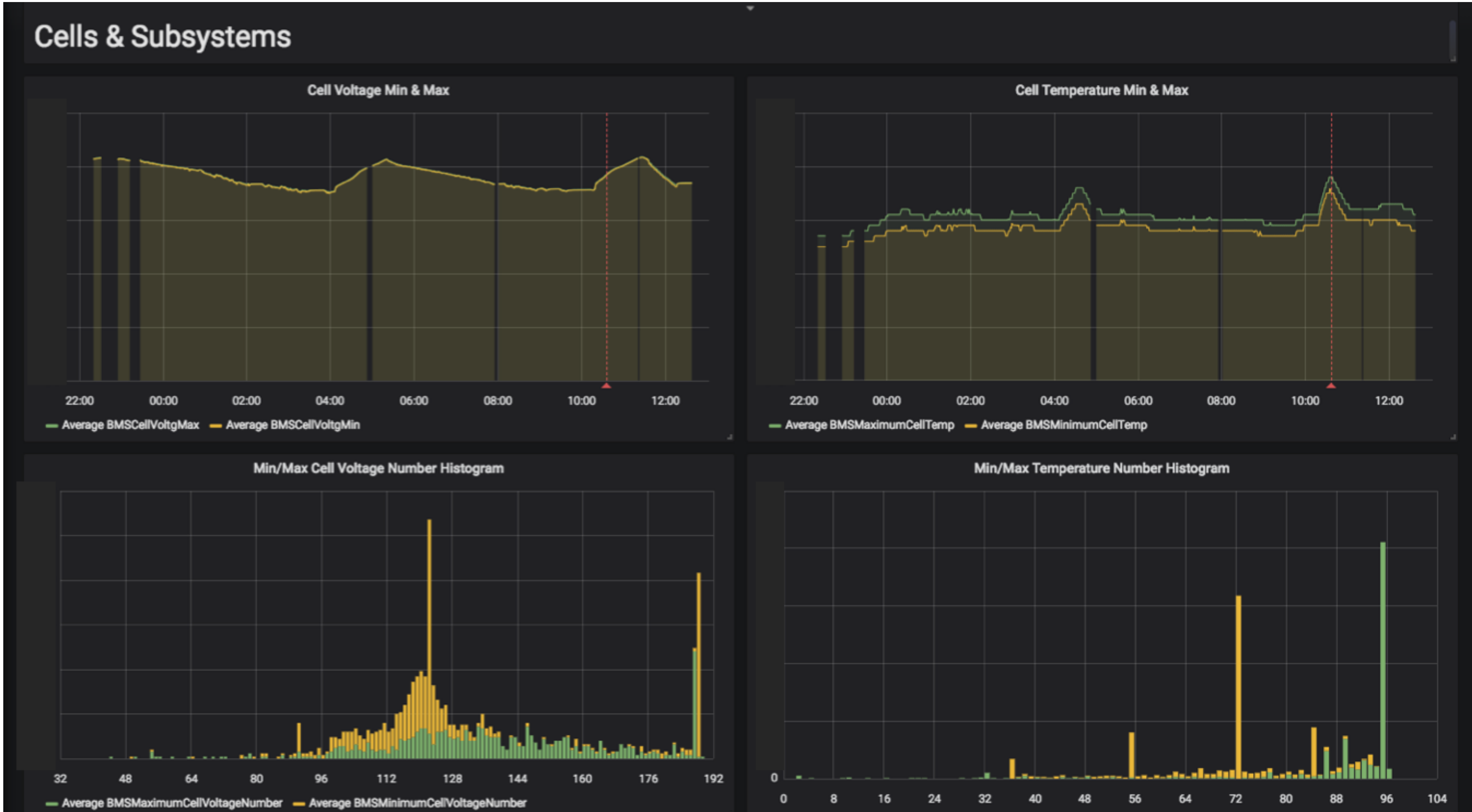
Simulation Deployment



Battery Analytics Deployment

- Algorithms are prototyped in MATLAB implemented in Python
    - For each vehicle
        - Queries data for a trip from Elasticsearch
        - Runs analytics algorithms on the trip data
        - Pushes results back to Elasticsearch
    - Results include time-based analytics (e.g., state-of-charge) and trip summary metrics (e.g., SOH)
- Implemented as a batch job that is Dockerized


Battery Pack
BMS Pack Voltage
22:00
00:00
02:00
04:00
06:00
08:00
10:00
12:00
Average BMSPackVoltage
BMS Pack Current
Average BMSPackCrrnt
BMS Pack SOC
100%
Average BMSSOC
BMS Pack Average Temperture
Edit Annotation
2018-04-12 10:36:20
Description
High Temperature
Tags
high_bms_temp x
add tags
Save
Delete
Cancel
Average BMSTempAverage



Cells & Subsystems
Cell Voltage Min & Max
22:00
00:00
02:00
04:00
06:00
08:00
10:00
12:00
Average BMSCellVoltgMax
Average BMSCellVoltgMin
Cell Temperature Min & Max
22:00
00:00
02:00
04:00
06:00
08:00
10:00
12:00
Average BMSMaximumCellTemp
Average BMSMinimumCellTemp
Min/Max Cell Voltage Number Histogram
32
48
64
80
96
112
128
144
160
176
192
Average BMSMaximumCellVoltageNumber
Average BMSMinimumCellVoltageNumber
Min/Max Temperature Number Histogram
0
0
8
16
24
32
40
48
56
64
72
80
88
96
104

NIO

Conclusions

Summary

- Simulations executed on request basis or in batch mode
- Dashboards combining vehicle- and fleet-level metrics built from Elasticsearch data source
- Automated pipelines running on test vehicles
- Deployment to production in progress

Next Steps

- Scalability
- Validation with customer data
- Connection to service operations