



Physics and Hybrid Model Approaches for Prognostics and Decision Making

Chetan S. Kulkarni, PhD

Technical Fellow, KBR

Diagnostics and Prognostics Group
NASA Ames Research Center



Acknowledgement

Diagnostics and Prognostics Group
NASA Ames Research Center

SWS Team
NASA Langley Research Center

Collaborators

Prof. Olga Fink, Manuel Chao – ETH Zurich

Dr. Kai Goebel – PARC

Prof. Felipe Viana, Renato Nascimento -
University of Central Florida



Credit: www.nasa.gov



Credit: www.nasa.gov



Credit: www.nasa.gov

Outline

- Introduction to Prognostics
- Introduction to Model-based Prognostics
- Research Approach
- Architecture
- Case Study : Prognostics of Li-Ion Batteries
- Hybrid Modeling for Prognostics
- Closing Remarks

INTRODUCTION TO PROGNOSTICS

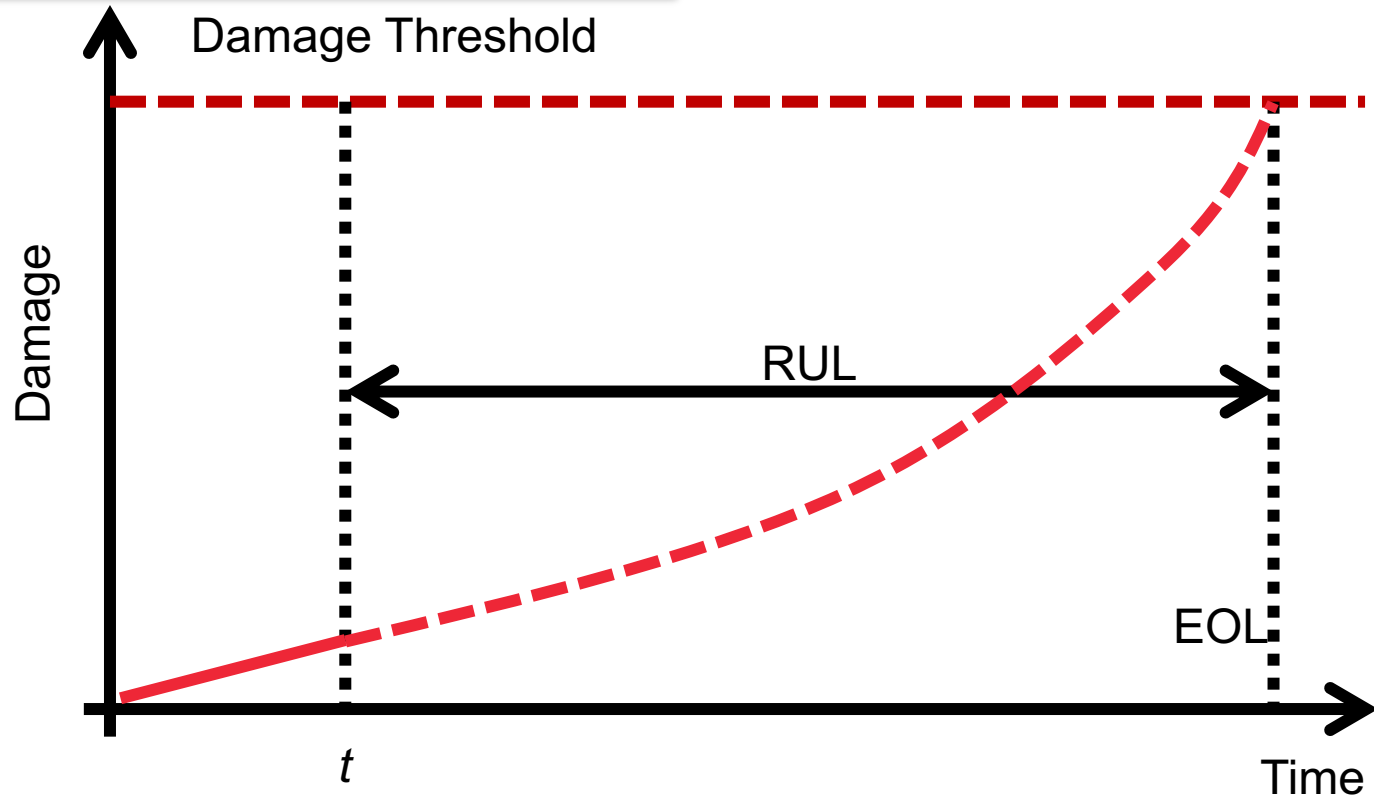
Why Diagnostics

- di·ag·nos·tic
 - a distinctive symptom or characteristic.
 - a program or routine that helps a user to identify errors.
 - the practice or techniques of diagnosis.
 - "advanced medical diagnostics"
 - PHM Community – “Detect and Isolate”
 - Fault Magnitude
 - System/Component

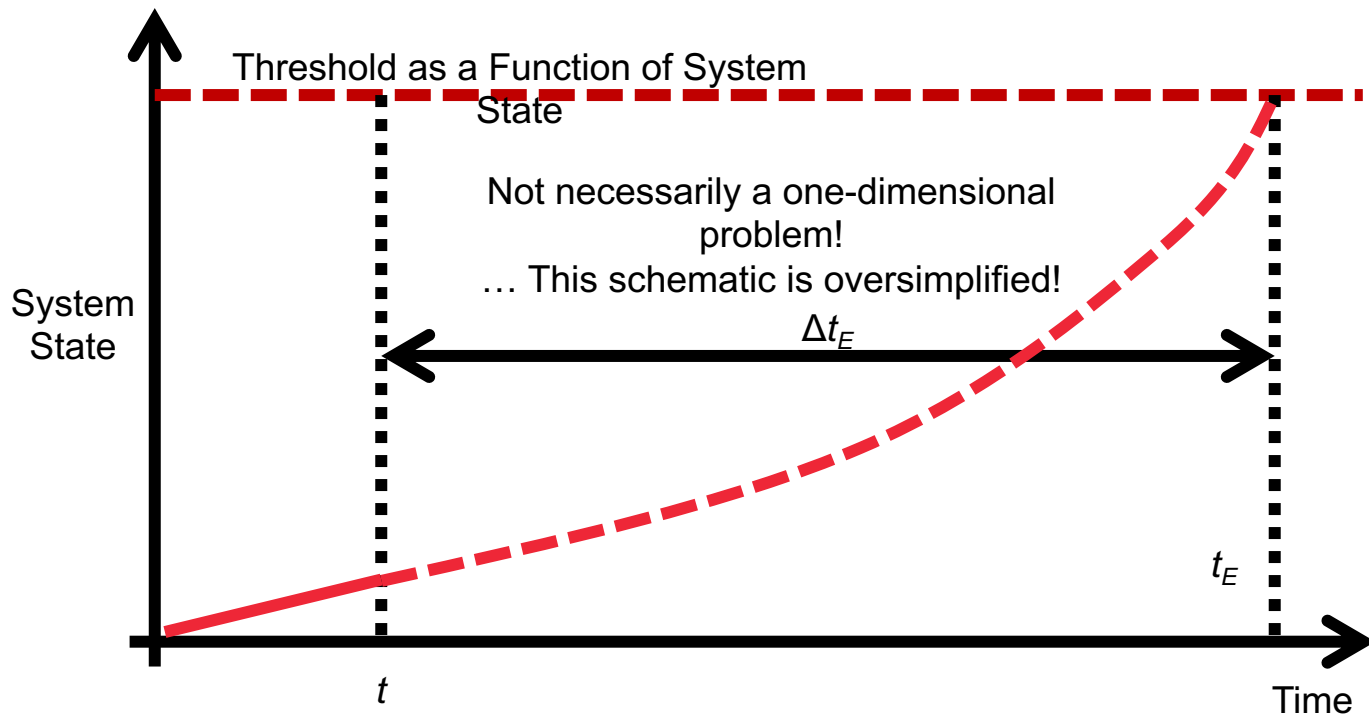
Why Prognostics

- Safety and Decision Making
- Reliability & Performance
 - product reputation reduced safety factors
- Operational Optimization
 - Prolonging component life by modifying how the component is used (e.g., load shedding/distribution)
 - Optimally plan or replan a mission

Basic Idea

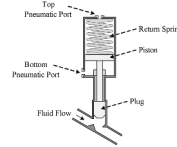


Basic Idea



- **RUL: Remaining Useful Life**

- Model underlying physics of a component/subsystem



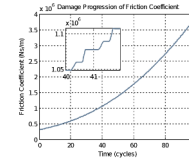
$$f_g(p_1, p_2) = \begin{cases} C_s A_s p_1 \sqrt{\frac{\gamma}{Z R_g T} \left(\frac{2}{\gamma+1} \right)^{(\gamma+1)/(\gamma-1)}}, & p_1 \geq p_2 \wedge p_1/p_2 \geq \left(\frac{\gamma+1}{2} \right)^{\gamma/(\gamma-1)} \\ C_s A_s p_1 \sqrt{\frac{2}{Z R_g T} \left(\frac{\gamma}{\gamma+1} \right) \left(\left(\frac{p_2}{p_1} \right)^{2/\gamma} - \left(\frac{p_2}{p_1} \right)^{(\gamma+1)/\gamma} \right)}, & p_1 \geq p_2 \wedge p_1/p_2 < \left(\frac{\gamma+1}{2} \right)^{\gamma/(\gamma-1)} \\ C_s A_s p_2 \sqrt{\frac{\gamma}{Z R_g T} \left(\frac{2}{\gamma+1} \right)^{(\gamma+1)/(\gamma-1)}}, & p_1 < p_2 \wedge p_2/p_1 \geq \left(\frac{\gamma+1}{2} \right)^{\gamma/(\gamma-1)} \\ C_s A_s p_2 \sqrt{\frac{2}{Z R_g T} \left(\frac{\gamma}{\gamma+1} \right) \left(\left(\frac{p_1}{p_2} \right)^{2/\gamma} - \left(\frac{p_1}{p_2} \right)^{(\gamma+1)/\gamma} \right)}, & p_1 < p_2 \wedge p_2/p_1 < \left(\frac{\gamma+1}{2} \right)^{\gamma/(\gamma-1)} \end{cases}$$

$$f_t(t) = f_g(p_t(t), u_t(t))$$

$$f_b(t) = f_g(p_b(t), u_b(t))$$

$$f_v(t) = \frac{x(t)}{L_s} C_v A_v \sqrt{\frac{2}{\rho} |p_{f1} - p_{fr}| \text{sign}(p_{f1} - p_{fr})}$$

- Model physics of damage propagation mechanisms



$$\dot{r}(t) = w_r |F_f(t) v(t)|$$

- Determine criteria for End-of-Life threshold



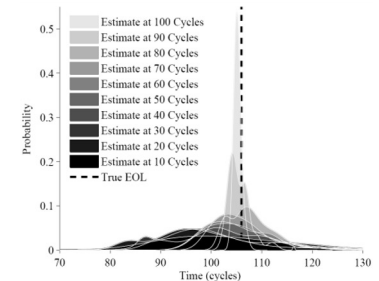
$$EOL(t_P) \triangleq \inf\{t \in \mathbb{R} : t \geq t_P \wedge T_{EOL}(\mathbf{x}(t), \boldsymbol{\theta}(t)) = 1\}$$

- Develop algorithms to propagate damage into future



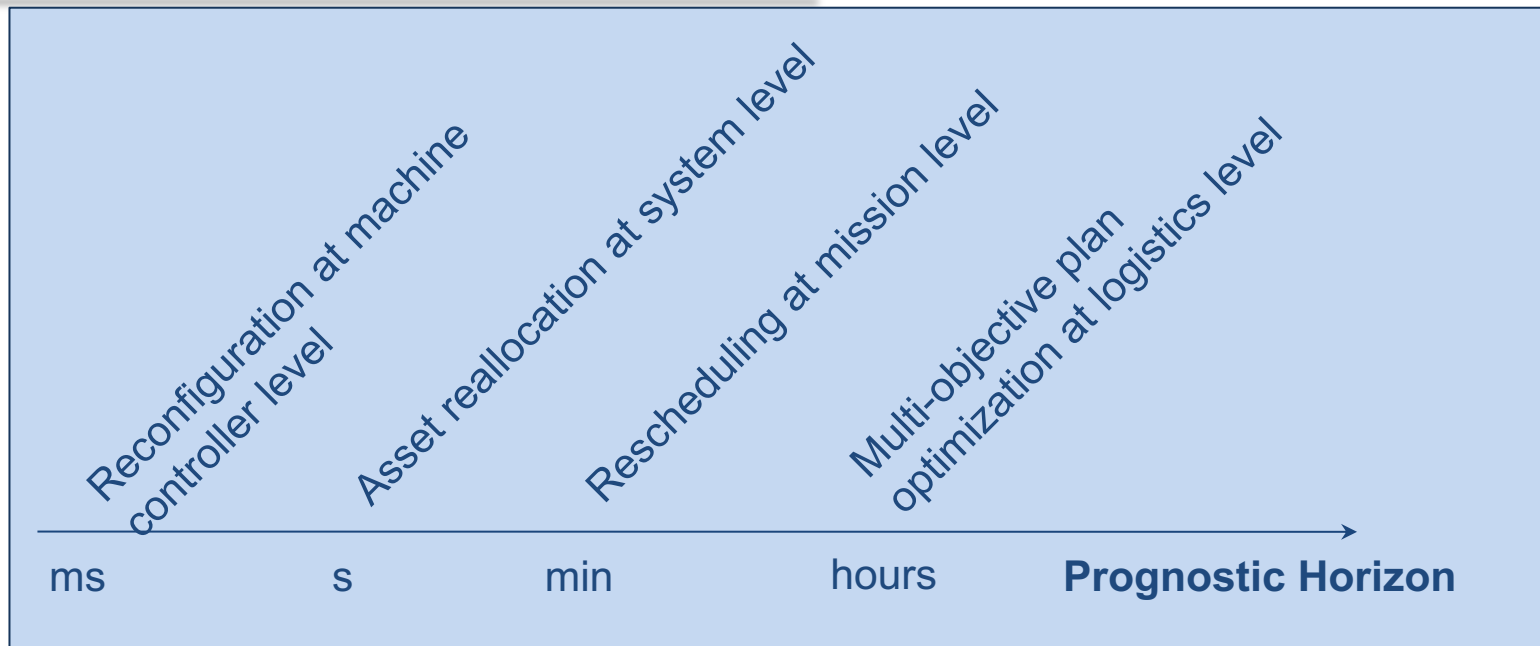
Algorithm 2 EOL Prediction

Inputs: $\{\mathbf{x}_{k_P}^i, \boldsymbol{\theta}_k^i, w_{k_P}^i\}_{i=1}^N$
Outputs: $\{EOL_{k_P}^i, w_{k_P}^i\}_{i=1}^N$
for $i = 1$ **to** N **do**
 $k \leftarrow k_P$
 $\mathbf{x}_k^i \leftarrow \mathbf{x}_{k_P}^i$
 $\boldsymbol{\theta}_k^i \leftarrow \boldsymbol{\theta}_{k_P}^i$
 while $C_{EOL}(\mathbf{x}_k^i, \boldsymbol{\theta}_k^i) = 0$ **do**
 Predict $\mathbf{u}_k$
 $\boldsymbol{\theta}_{k+1}^i \sim p(\boldsymbol{\theta}_{k+1}^i | \boldsymbol{\theta}_k^i)$
 $\mathbf{x}_{k+1}^i \sim p(\mathbf{x}_{k+1}^i | \mathbf{x}_k^i, \boldsymbol{\theta}_k^i, \mathbf{u}_k)$
 $k \leftarrow k + 1$
 $\mathbf{x}_k^i \leftarrow \mathbf{x}_{k+1}^i$
 $\boldsymbol{\theta}_k^i \leftarrow \boldsymbol{\theta}_{k+1}^i$
 end while
 $EOL_{k_P}^i \leftarrow k$
end for



- Deal with uncertainty





Pilots

Air Traffic Control

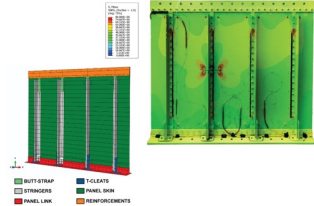
Autonomy.

Airline Operators

Operators

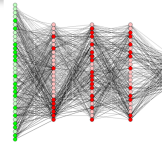
Maintainers

State of the Art



- Results tend to be intuitive
- Models can be reused
- If incorporated early enough in the design process, can drive sensor requirements
- Computationally efficient to implement
- Model development requires a thorough understanding of the system
- High-fidelity models can be computationally intensive

- Paris-Erdogan Crack Growth Model
- Taylor tool wear model
- Corrosion model
- Abrasion model



- Easy and Fast to implement
- May identify relationships that were not previously considered
- Requires lots of data and a “balanced” approach”
- Results may be counter(or even un-)intuitive
- Can be computationally intensive, both for analysis and implementation

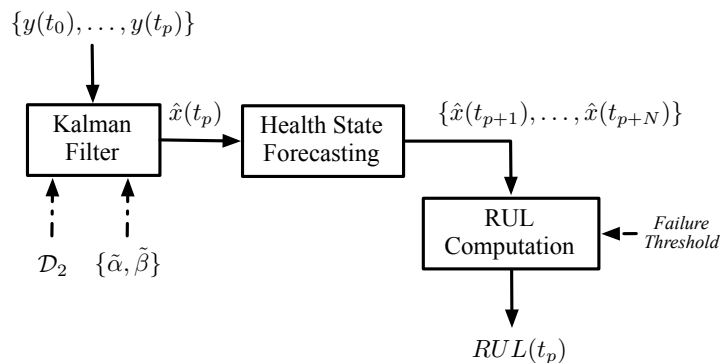
- Regression analysis
- Neural Networks (NN)
- Bayesian updates
- Relevance vector machines (RVM)

Model-based prognostics

- State vector includes dynamics of normal and degradation process

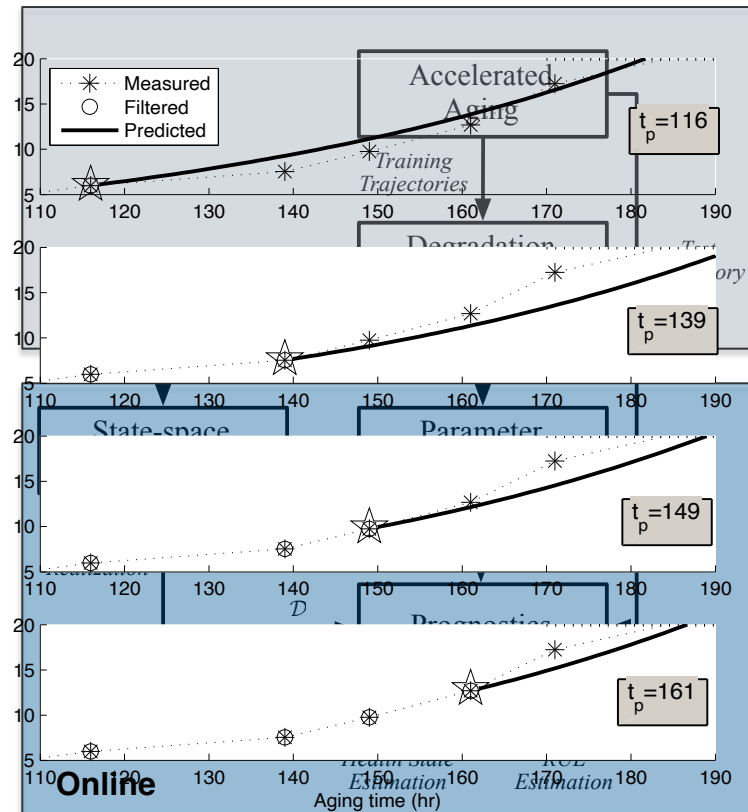
$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}$$

$$y_k = Hx_k + v_k$$

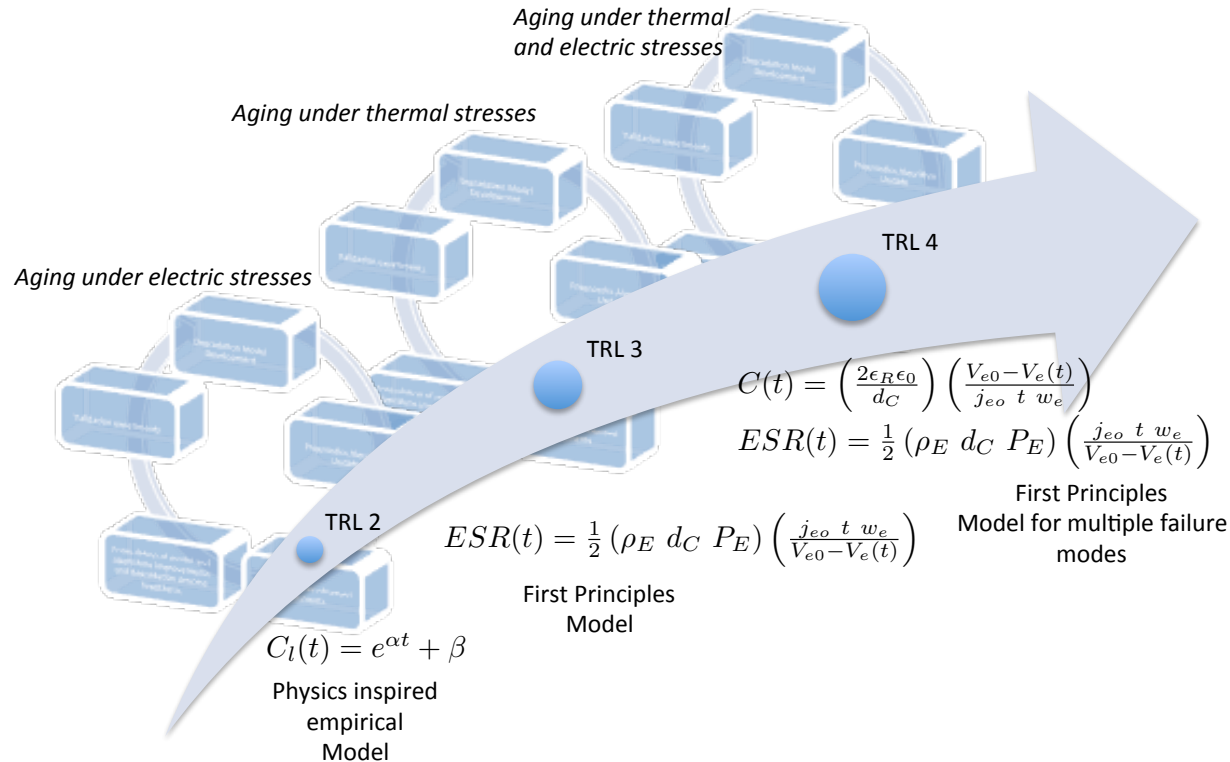


- EOL defined at time in which performance variable cross failure threshold

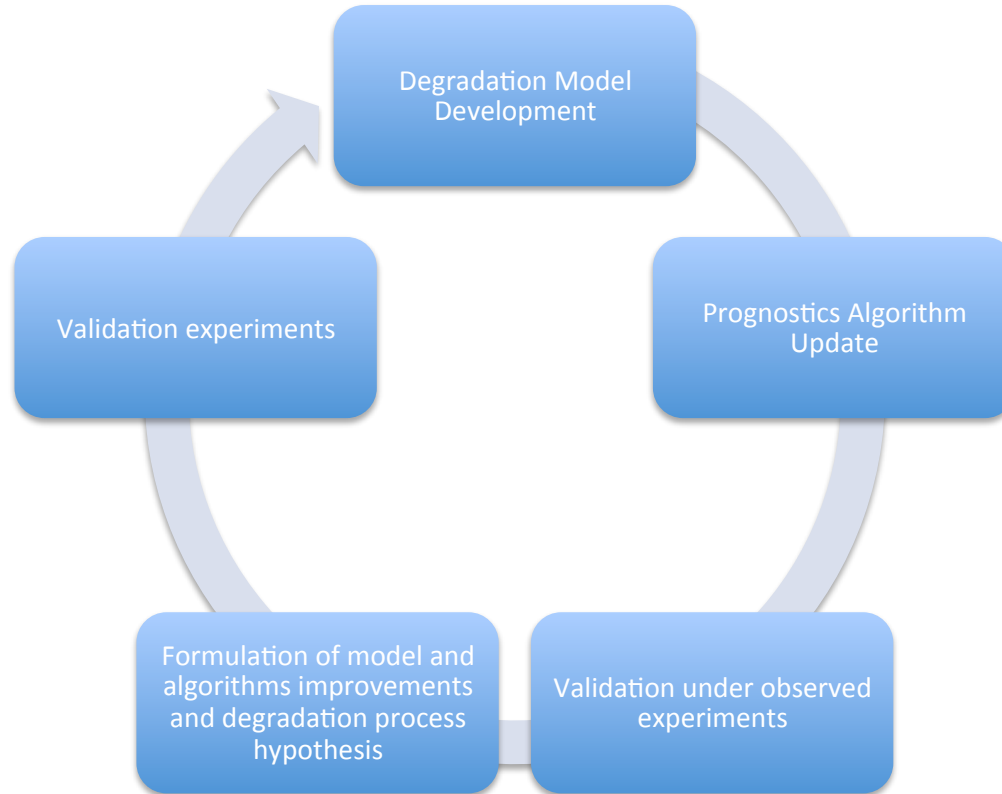
$$R(t_p) = t_{EOL} - t_p$$



Model and Algorithm Maturation

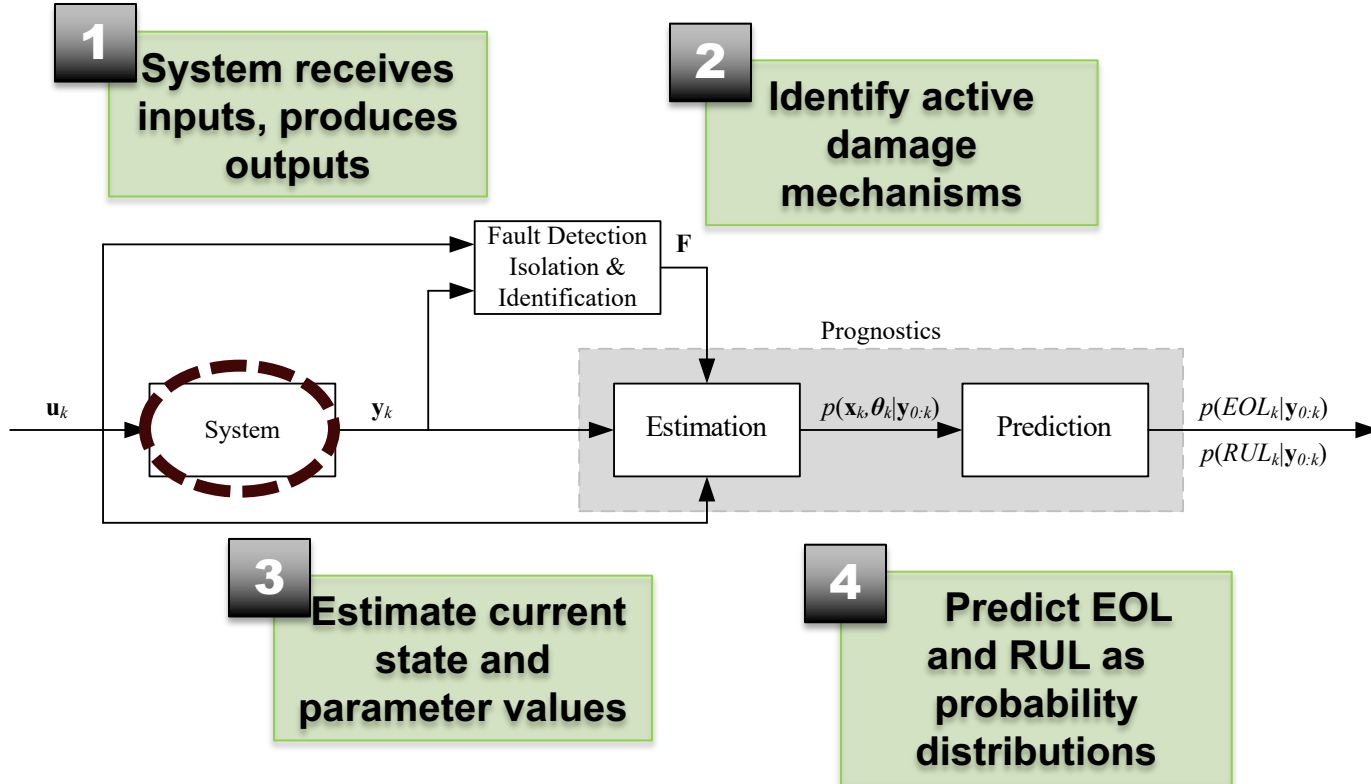


Algorithm Maturation - Validation



Architecture

Model-Based Architecture



Initial Problem Formulation

- Assume we know
 - Initial state, $\mathbf{x}(k_o)$
 - Future input trajectory, $\mathbf{U}_{k_o, k_h} = [\mathbf{u}(k_o), \mathbf{u}(k_o + 1), \dots, \mathbf{u}(k_h)]$
 - Process noise trajectory, $\mathbf{V}_{k_o, k_h} = [\mathbf{v}(k_o), \mathbf{v}(k_o + 1), \dots, \mathbf{v}(k_h)]$
- Problem definition
 - Given $k_o, k_h, \mathbf{x}(k_o), \mathbf{U}_{k_o, k_h}, \mathbf{V}_{k_o, k_h}$
 - Compute EOL
 - $\text{EOL}(k) = \inf\{k' : k' \geq k \text{ and } T_f(\mathbf{x}(k))\}$

Computational Algorithm

ComputeEOL($k_o, k_h, \mathbf{x}(k_o), \mathbf{U}_{k_o, k_h}, \mathbf{V}_{k_o, k_h}$)

1. $\mathbf{X}_{k_o, k_h}(k_o) \leftarrow \mathbf{x}(k_o)$

2. **for** $k = k_o$ **to** $k_h - 1$ **do**

3. **if** $T_f(\mathbf{X}_{k_o, k_h})(k)$

4. **return** k

5. **end if**

6. $\mathbf{X}_{k_o, k_h}(k+1) \leftarrow f(\mathbf{X}_{k_o, k_h}(k), \mathbf{U}_{k_o, k_h}(k), \mathbf{V}_{k_o, k_h}(k))$

7. **end for**

8. **if** $T_f(\mathbf{X}_{k_o, k_h})(k)$

9. **return** k

10. **else**

11. **return** ∞

12. **end if**

// Set initial state

// Check if failure state

// Return current time as EOL

// Update state

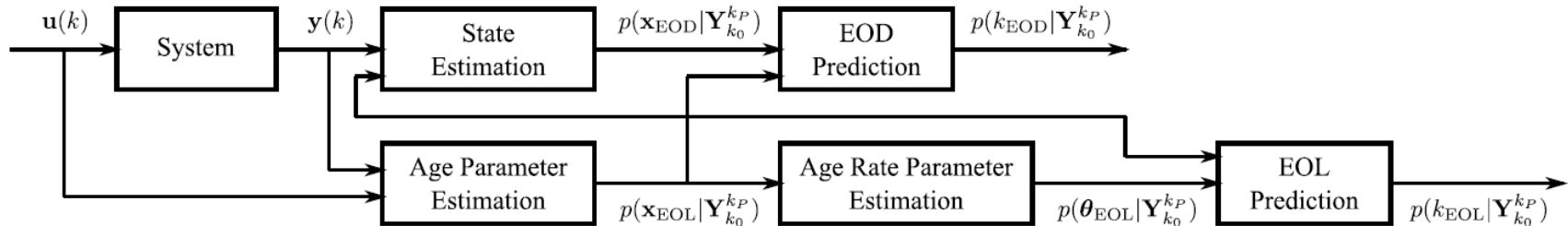
// Check if failure state

// Return current time (k_h) as EOL

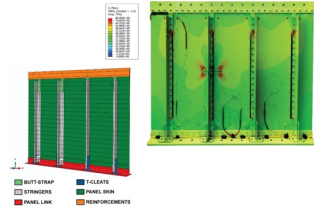
// Return infinity

Integrated Prognostics Architecture

- System (battery) gets inputs (current) and produces outputs (voltage)
- State estimation computes estimate of state given estimates of age parameters
- EOD prediction computes prediction of time of EOD, given state and age parameter estimates
- Age parameter estimation computes estimates of age parameters
- Age rate parameter estimation computes parameters defining aging rate progression
- EOL prediction computes prediction of time of EOL, given age parameter and age rate parameter estimates

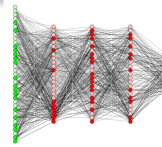


State of the Art



- Results tend to be intuitive
- Models can be reused
- If incorporated early enough in the design process, can drive sensor requirements
- Computationally efficient to implement
- Model development requires a thorough understanding of the system
- High-fidelity models can be computationally intensive

- Paris-Erdogan Crack Growth Model
- Taylor tool wear model
- Corrosion model
- Abrasion model



- Easy and Fast to implement
- May identify relationships that were not previously considered
- Requires lots of data and a “balanced” approach”
- Results may be counter(or even un-)intuitive
- Can be computationally intensive, both for analysis and implementation

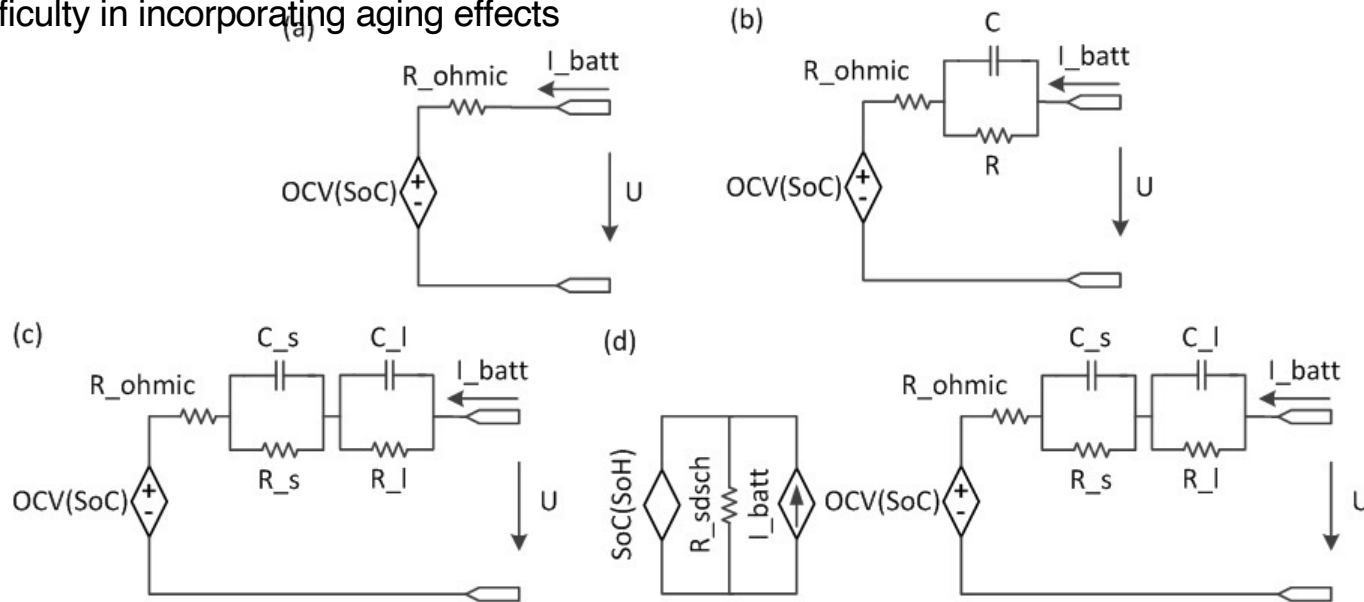
- Regression analysis
- Neural Networks (NN)
- Bayesian updates
- Relevance vector machines (RVM)

Case Study : Prognostics of Li-Ion Batteries

Battery Modeling

- Equivalent Circuit Empirical Models

- Most common approach
- Various model complexities used
- Difficulty in incorporating aging effects



Battery Model – Tuned using Lab Data

- An equivalent circuit battery model is used to represent the battery terminal voltage as a function of current and the charge stored in 3 capacitive elements

$$x = [q_b \ q_{cp} \ q_{Cs}]^T$$

$$\dot{x} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\frac{1}{R_{cp}C_{cp}} & 0 \\ 0 & 0 & -\frac{1}{R_sC_s} \end{bmatrix} x + \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} i + \xi$$

$$y = V = \left[\frac{1}{C_b} - \frac{1}{C_{cp}} - \frac{1}{C_s} \right] \cdot x$$

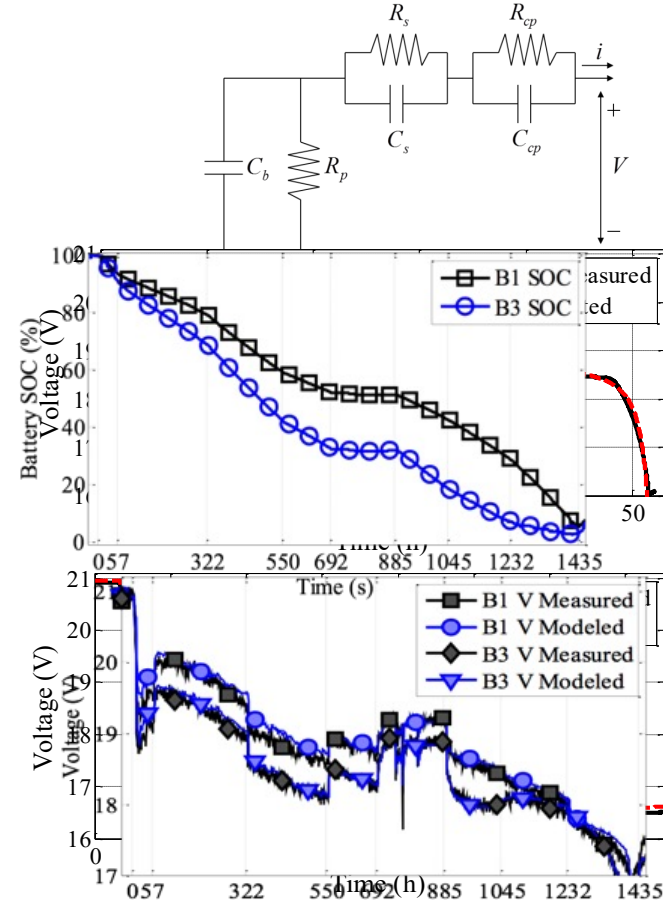
- Two laboratory loading experiments are used to fit the following parameterization coefficients

$$SOC = 1 - \frac{q_{max} - q_b}{C_{max}}$$

$$C_b = C_{Cb0} + C_{Cb1} \cdot SOC + C_{Cb2} \cdot SOC^2 + C_{Cb3} \cdot SOC^3$$

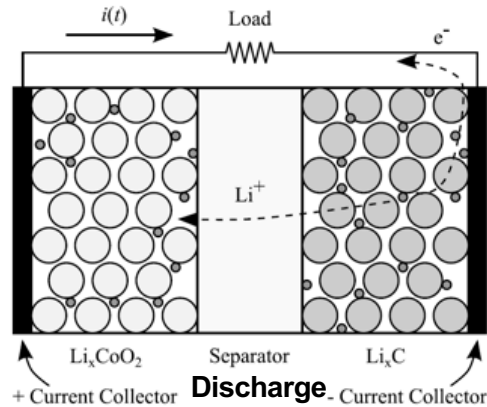
$$C_{cp} = C_{cp0} + C_{cp1} \cdot \exp(C_{cp2}(1 - SOC))$$

$$R_{cp} = R_{cp0} + R_{cp1} \cdot \exp(R_{cp2}(1 - SOC))$$

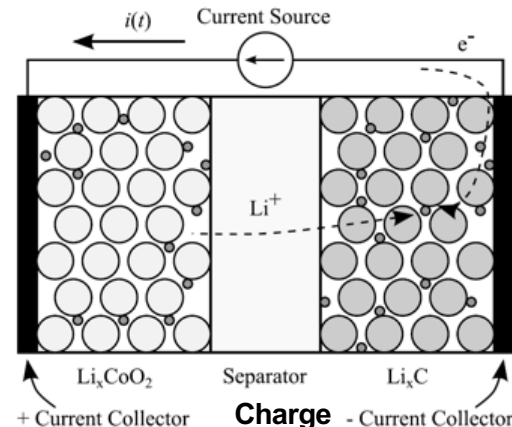


Battery Modeling

- Electrochemical Models vs. Empirical Models
 - Battery physics models enable more direct representation of age-related changes in battery dynamics than empirical models
 - Typically have a higher computational cost and more unknown parameters



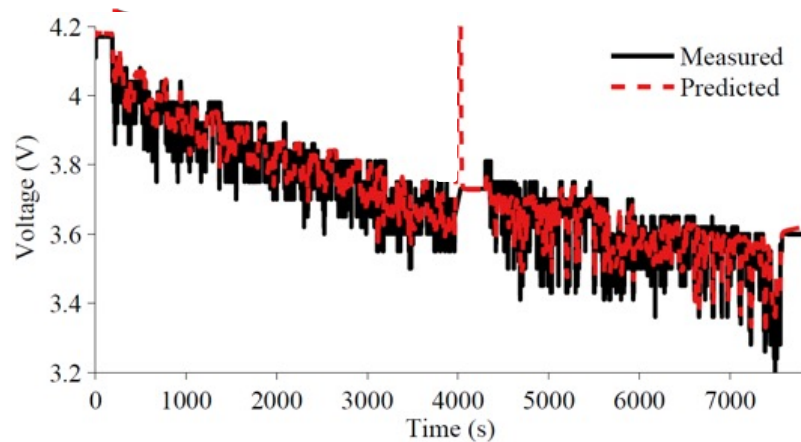
Reduction at pos. electrode:
 $\text{Li}_{1-n}\text{CoO}_2 + n\text{Li}^+ + n\text{e}^- \rightarrow \text{LiCoO}_2$
 Oxidation at neg. electrode:
 $\text{Li}_n\text{C} \rightarrow n\text{Li}^+ + n\text{e}^- + \text{C}$
 Current flows + to -
 Electrons flow - to +
 Lithium ions flow - to +



Oxidation at pos. electrode:
 $\text{LiCoO}_2 \rightarrow \text{Li}_{1-n}\text{CoO}_2 + n\text{Li}^+ + n\text{e}^-$
 Reduction at neg. electrode:
 $n\text{Li}^+ + n\text{e}^- + \text{C} \rightarrow \text{Li}_n\text{C}$
 Current flows - to +
 Electrons flow + to -
 Lithium ions flow + to -

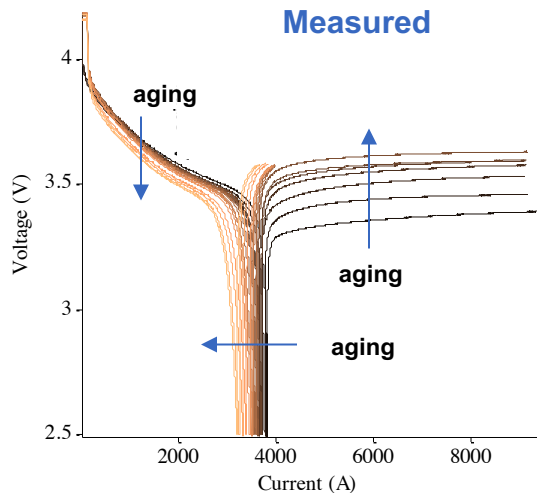
Electrochemical Li-ion Model

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



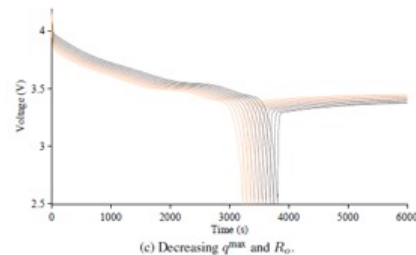
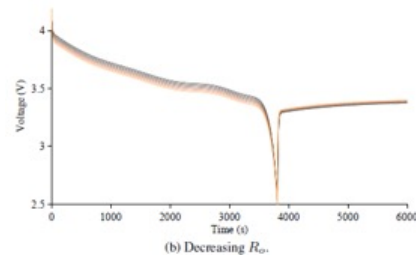
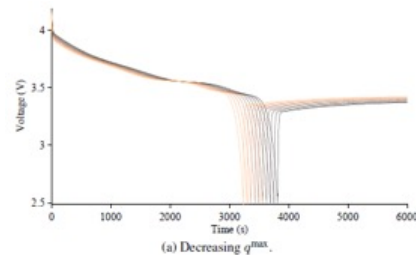
Battery Aging

- Contributions from both decrease in mobile Li ions (lost due to side reactions related to aging) and increase in internal resistance
 - Modeled with decrease in " $q^{\max}$ " parameter, used to compute mole fraction
 - Modeled with increase in " R_o " parameter capturing lumped resistances



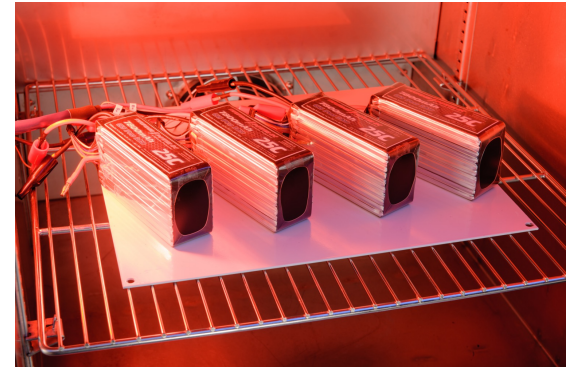
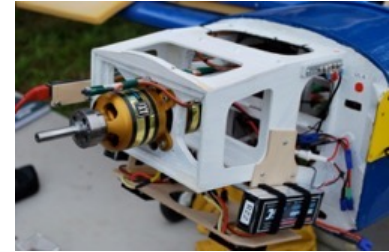
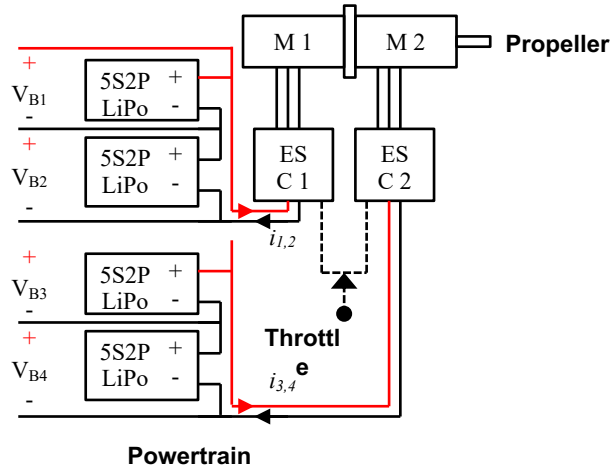
— Cycle 16
— Cycle 26
— Cycle 36
— Cycle 46
— Cycle 56
— Cycle 66
— Cycle 76
— Cycle 86
— Cycle 96
— Cycle 106
— Cycle 116
— Cycle 126
— Cycle 136
— Cycle 146
— Cycle 156
— Cycle 166
— Cycle 176
— Cycle 186

Simulated



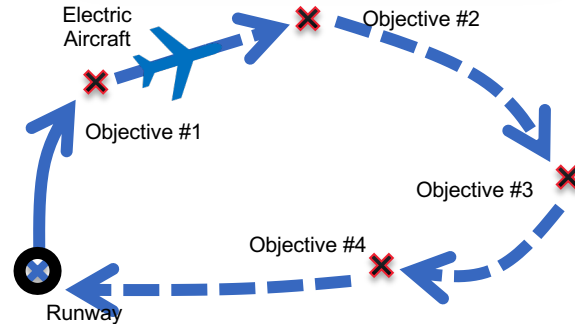
Edge 540-T

- Subscale electric aircraft operated at NASA Langley Research Center
- Powered by four sets of Li-polymer batteries
- Estimate SOC online and provide EOD and remaining flight time predictions for ground-based pilots



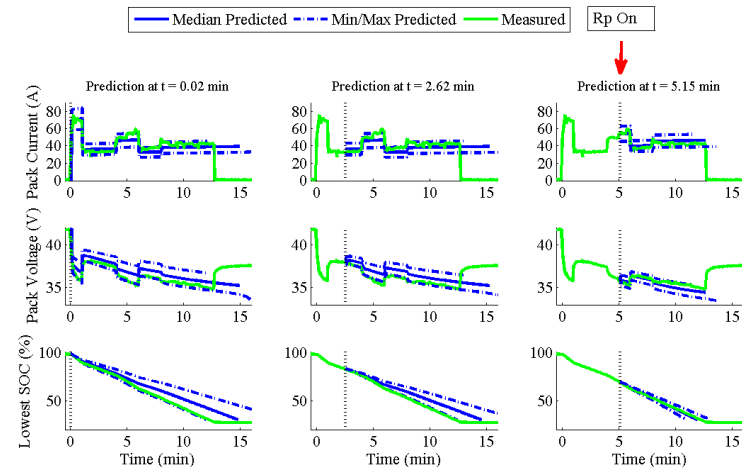
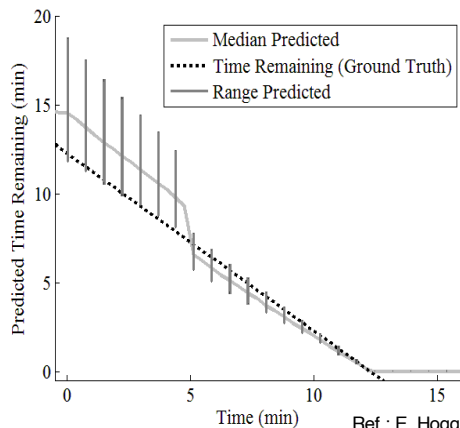
Edge UAV Use Case

- Piloted and autonomous missions, visiting waypoints
- Require 2-minute warning for EOD so pilot/autopilot has sufficient time to land safely
 - This answer depends on battery age
 - Need to track both current level of charge and current battery age
 - Based on current battery state, current battery age, and expected future usage, can predict EOD and correctly issue 2-minute warning



Predication over Flight Plan

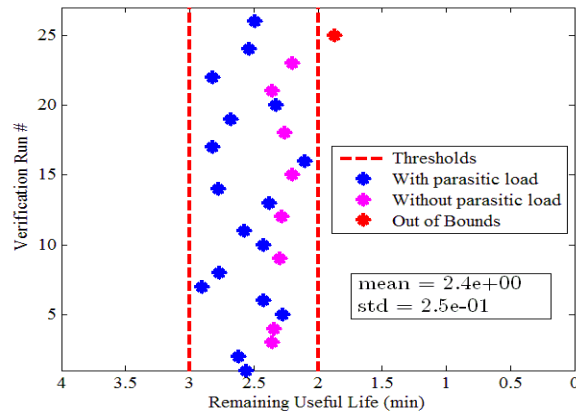
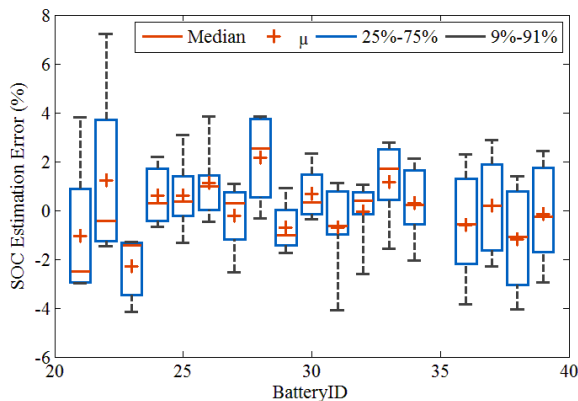
- Measured and predicted battery current, voltage and SOC different time steps
- The min, max and median predictions are plotted from each sample time until the predicated SOC reaches 30%



- Predictions for remaining flight time for entire flight plan
- Overestimate till parasitic load is injected
- Once the parasitic load is detected the remaining flying time prediction shifts down.

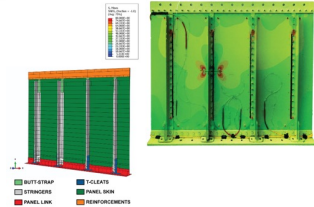
Performance Requirements

- Accuracy requirements for the two minute warning were specified as:
 - *The prognostic algorithm shall raise an alarm no later than two minutes before the lowest battery SOC estimate falls below 30% for at least 90% of verification trial runs.*
 - *The prognostic algorithm shall raise an alarm no earlier than three minutes before the lowest battery SOC estimate falls below 30% for at least 90% of verification trial runs.*
 - *Verification trial statistics must be computed using at least 20 experimental runs*



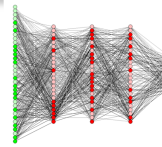
Hybrid Approaches

State of the Art



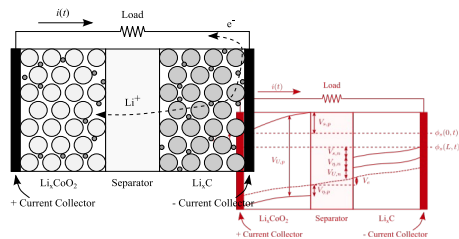
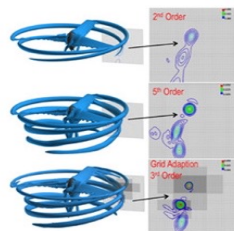
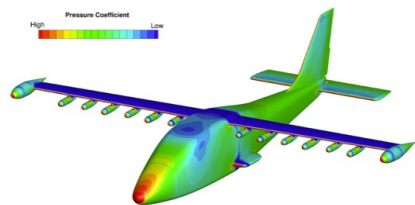
- Results tend to be intuitive
- Models can be reused
- If incorporated early enough in the design process, can drive sensor requirements
- Computationally efficient to implement
- Model development requires a thorough understanding of the system
- High-fidelity models can be computationally intensive

- Paris-Erdogan Crack Growth Model
- Taylor tool wear model
- Corrosion model
- Abrasion model

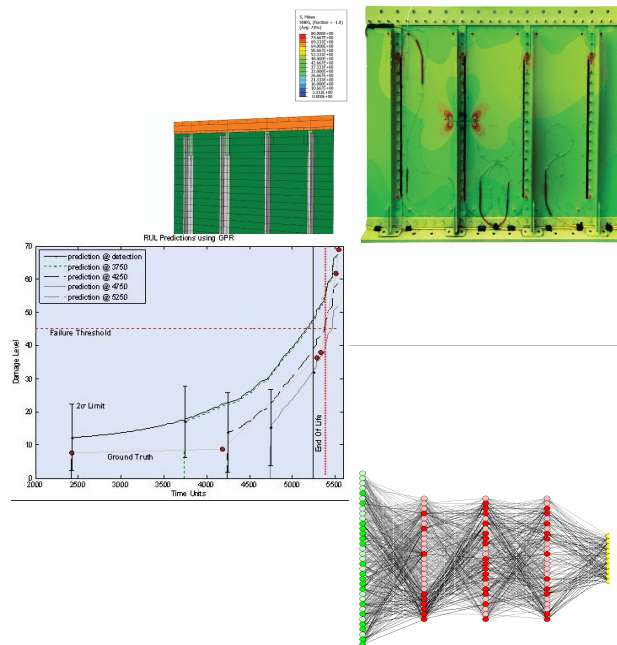


- Easy and Fast to implement
- May identify relationships that were not previously considered
- Requires lots of data and a “balanced” approach”
- Results may be counter(or even un-)intuitive
- Can be computationally intensive, both for analysis and implementation

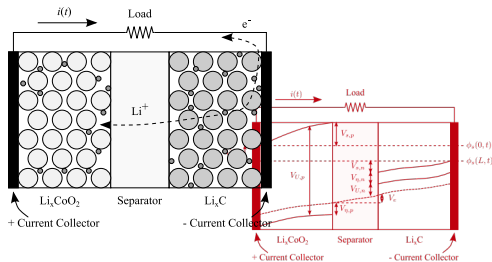
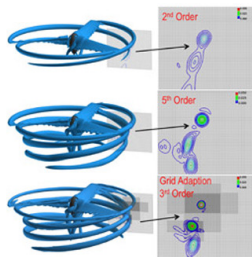
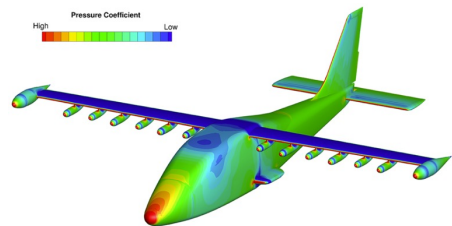
- Regression analysis
- Neural Networks (NN)
- Bayesian updates
- Relevance vector machines (RVM)



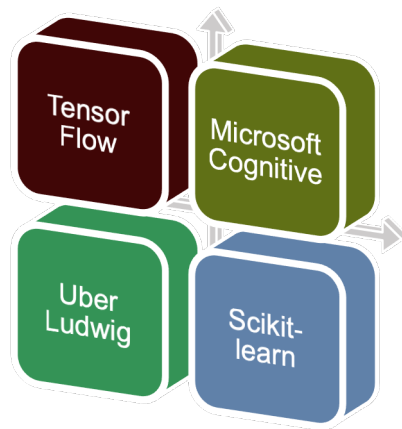
+



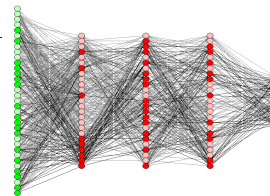
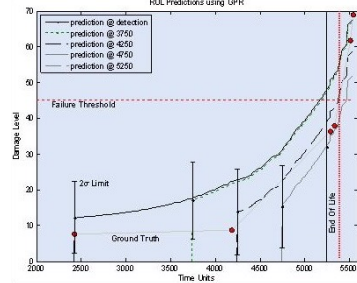
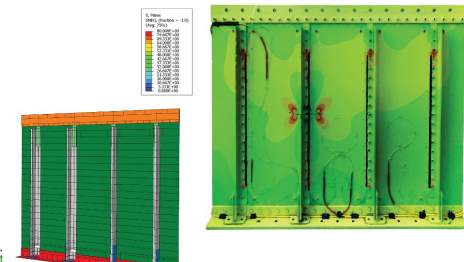
Hybrid Approach



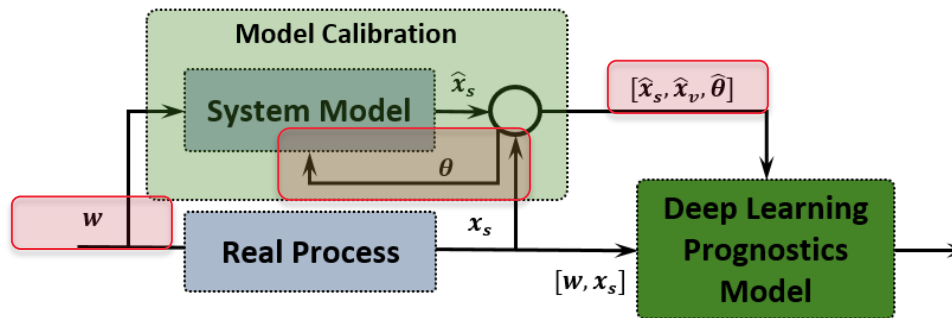
Machine-Learning underlying physics parameters



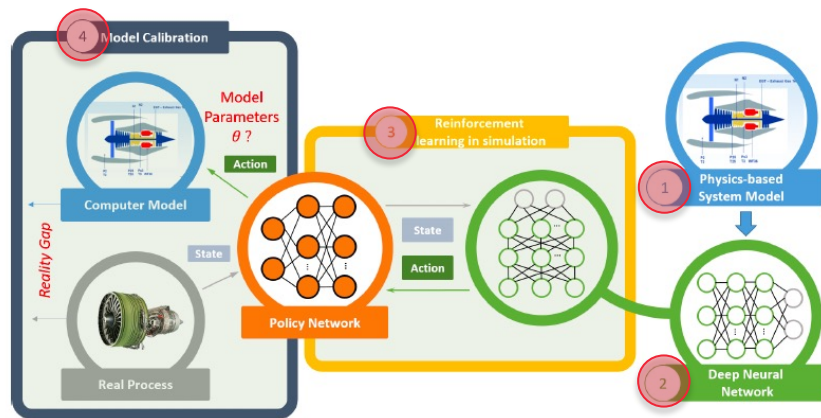
Understanding and Learning underlying Physics for Complex Systems



Approach 1 : Deep Learning + Physics Model Calibration



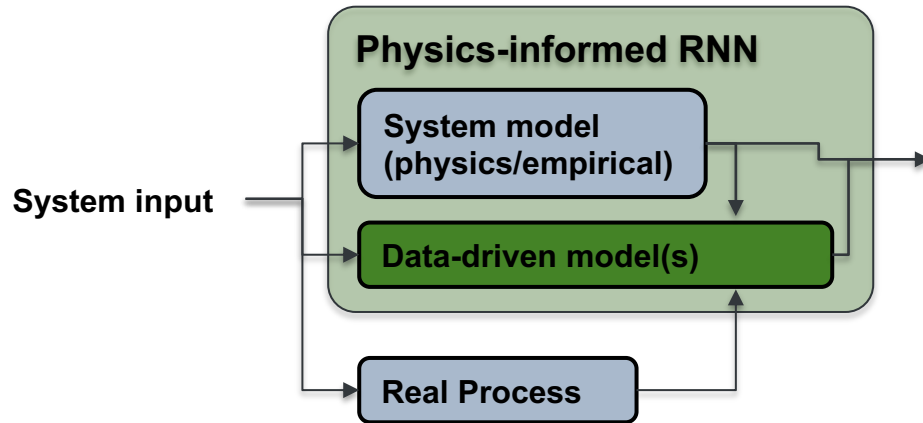
Overall architecture of the hybrid prognostics framework fusing physics-based and deep learning models.



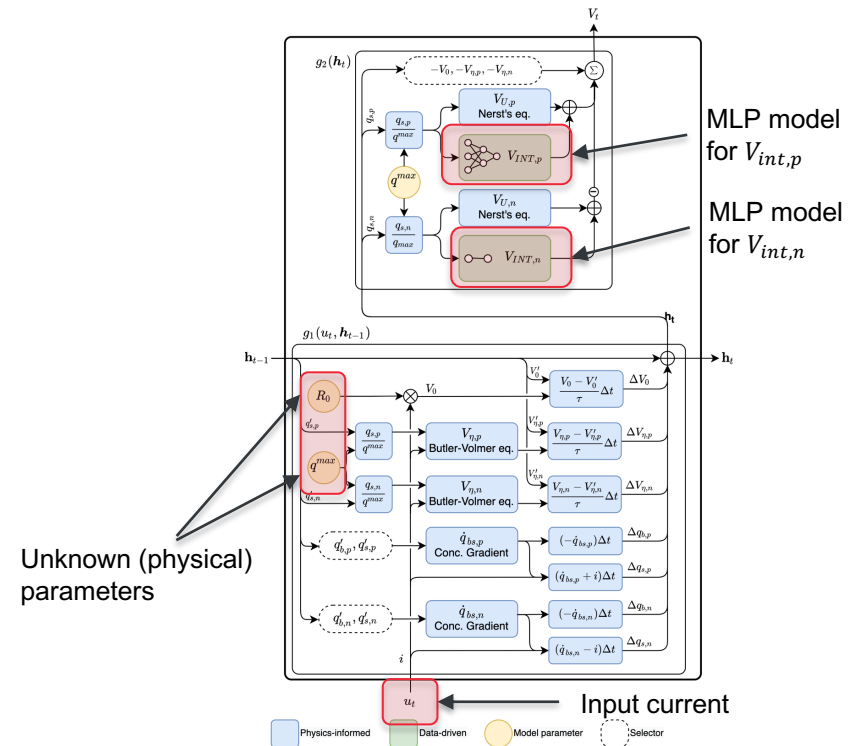
Calibration Policy

Yuan Tian, Manuel Arias Chao, Chetan Kulkarni, Kai Goebel, Olga Fink, "Real-Time Model Calibration with Deep Reinforcement Learning", arXiv:2006.04001

Approach 2 : Physics + RNN



Overall architecture of the physics-informed recurrent neural network

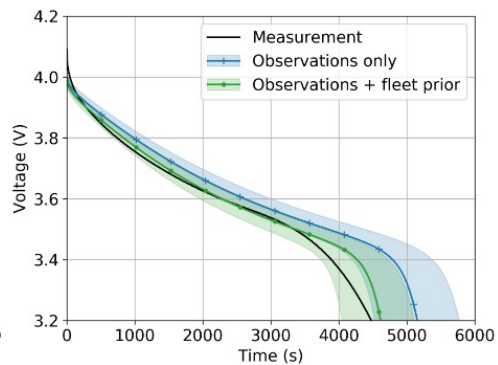
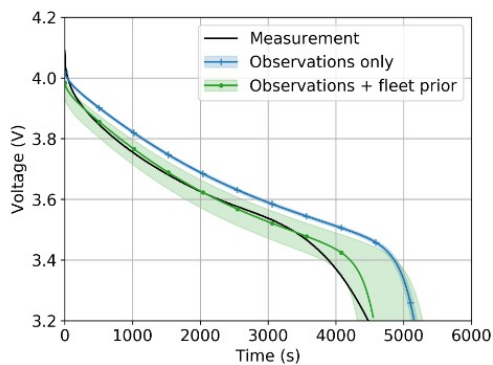
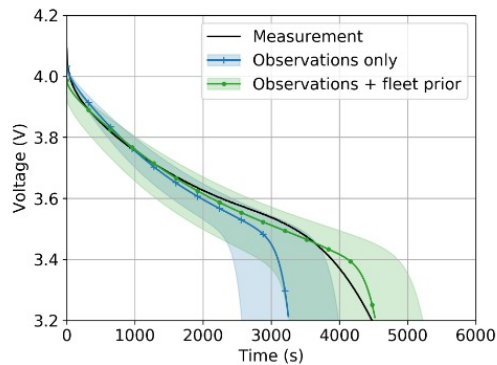


Physics-informed neural network framework for Li-ion Battery SOC estimation

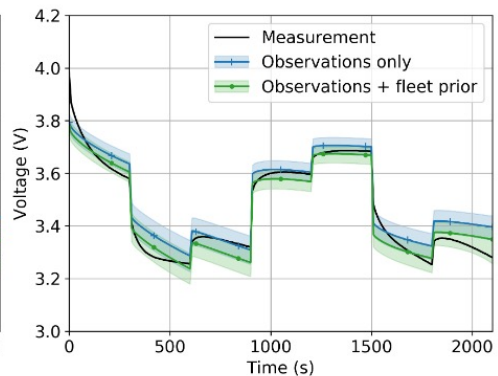
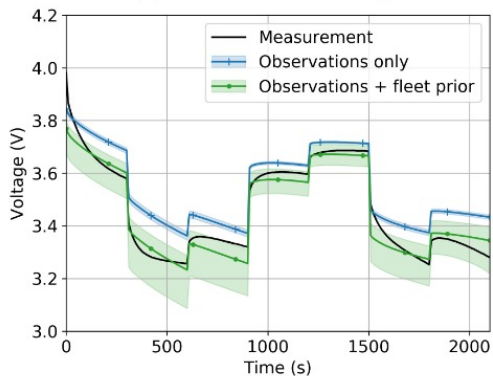
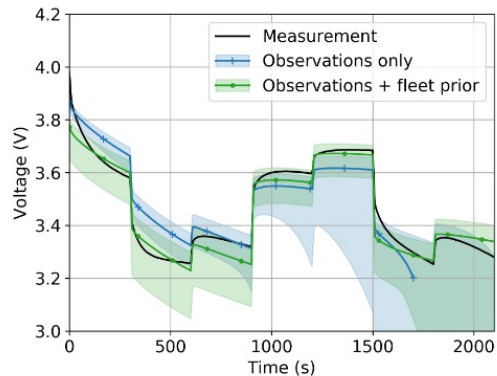
Nascimento, R.G. & Viana, F. A. & Corbetta, M. & Kulkarni, C. S. (2021). "Usage-based Lifting of Lithium-Ion Battery with Hybrid Physics-Informed Neural Networks," AIAA Aviation 2021.

Renato G. Nascimento; Matteo Corbetta; Chetan S. Kulkarni; Felipe A.C. Viana, "Hybrid Physics-Informed Neural Networks for Lithium-Ion Battery Modeling and Prognosis". Journal of Power Sources 2021 (accepted)

Approach 2 : Physics + RNN

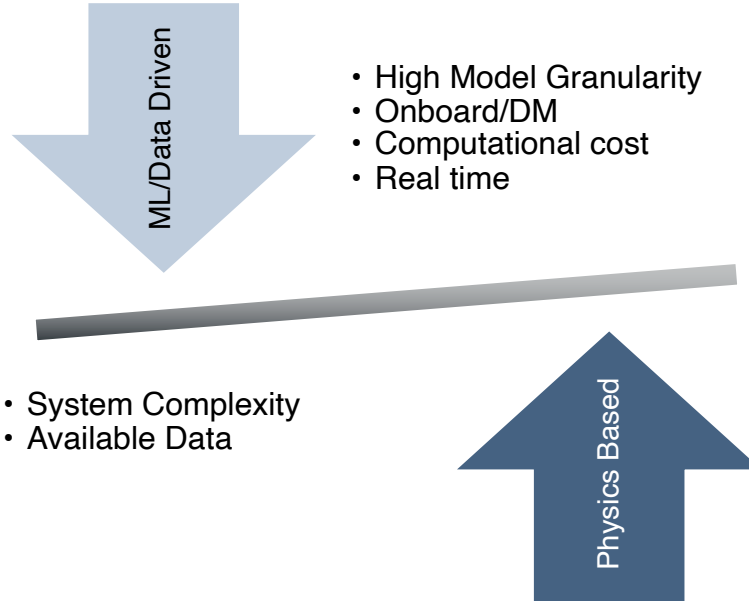


(a) Reference discharge.

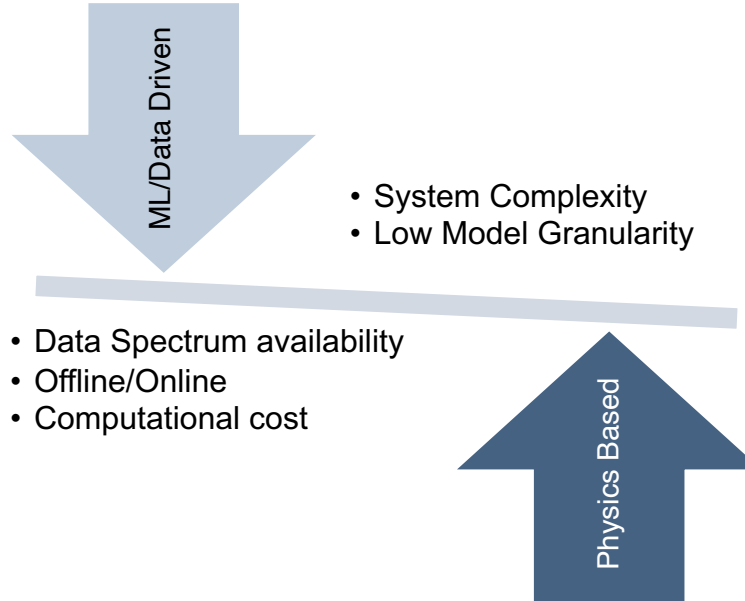


(b) Random-loading discharge.

Next Steps : Looking Ahead



Next Steps : Looking Ahead



Next Steps : Looking Ahead



Concluding Remarks

- Health Management framework helps enable
 - Systems safe and efficient
 - Decision making
- Research approach challenges
 - How to balance lack of knowledge of the system vs own expertise on particular PHM tools
- Validate models and algorithms with data from lab experiments and fielded systems

Concluding Remarks

- Hybrid Approaches
 - Physics based methods can be combined with machine learning to determine and evaluate models for complex physical systems.
 - High Fidelity simulation
 - Field and Tests
 - These models enable in verification and validation for autonomy in minimum period of time than current state of the art.
 - Computational tools are too slow.
 - With availability of test and field data, machine learning able to blend the digital data fabric for model update
 - Uncertainty Quantification
- Requirements for autonomous systems

<https://www.nasa.gov/content/prognostics-center-of-excellence-data-set-repository/>

Publications using this data set

Description	MOSFET Thermal Overstress Aging Data Set Publications using this data set
Format	
Datasets	
Dataset Citation	
Publication Citation	
Management	

Description	Capacitor Electrical Stress Data Set - 2 Publications using this data set
Format	
Datasets	
Dataset Citation	
Publication Citation	
Management	

Description	Turbofan Engine Degradation Simulation Data Set-2 Publications using this data set
Format	The generation of data-driven prognostics models requires the availability of datasets with run-to-failure trajectories. In order to contribute to the development of these methods, the dataset provides a new realistic dataset of run-to-failure trajectories for a small fleet of aircraft engines under realistic flight conditions. The damage propagation modelling used for the generation of this synthetic dataset builds on the modeling strategy from previous work . The dataset was generated with the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) dynamical model. The data set is been provided by the Prognostics CoE at NASA Ames in collaboration with ETH Zurich and PARC. Readme file for the dataset describing the experimetal details and data can be found here .
Format	The dataset is in csv format and has been zipped. A python code to unzip the files can be found here .
Datasets	+ Download Turbofan Engine Degradation Simulation Data Set - 2 (8862 downloads)
Dataset Citation	M. Chao, C.Kulkarni, K. Goebel and O. Fink (2021). "Aircraft Engine Run-to-Failure Dataset under real flight conditions", NASA Ames Prognostics Data Repository (http://ti.arc.nasa.gov/project/prognostic-data-repository), NASA Ames Research Center, Moffett Field, CA



Discussion !!

chetan.s.kulkarni@nasa.gov

<https://www.nasa.gov/content/prognostics-center-of-excellence>