Integrating AI-based Virtual Sensors into Model-Based Design

Emelie Andersson, MathWorks





Javier Gazzarri, MathWorks





Why Virtual Sensors?

When estimating a quantity that is not measurable

Battery State of Charge (SOC)



Not directly measurable

We measure voltage, current, temperature and calculate SOC

- Matlab **Expo**



SOC

Agenda

- Develop AI-based virtual sensor for battery SOC estimation
- Workflow From data acquisition to hardware deployment



Voltage

Current

Temperature

Compare different AI methods



Battery State of Charge (SOC)





Affected by sensor error

Extended Kalman Filter



How About...



Instead of creating a physics-based model – Train a Statistical Model

Comparison

Extended Kalman Filter

- Well established
- Accurate
- Detailed battery model required
 Operating condition range
- Computationally intensive

AI

- Training on real data
- Capture very complex data relationships
- No need for battery model
- Interpretability
- Computationally intensive

Al-driven System Design



Steps involved in creating an AI-based virtual sensor

Back to SOC estimation







Robust xEV Battery State-of-Charge Estimator Design Using a Feedforward Deep Neural Network

Carlos Vidal, Phillip Kollmeyer, and Mina Naguib McMaster Automotive Res. Centre

Pawel Malysz and Oliver Gross FCA US LLC

Ali Emadi McMaster University

Citation: Vidal, C., Kollmeyer, P., Naguib, M., Malysz, P. et al., "Robust xEV Battery State-of-Charge Estimator Design Using a Feedforward Deep Neural Network," SAE Technical Paper 2020-01-1181, 2020, doi:10.4271/2020-01-1181.

Abstract

B attery state-of-charge (SOC) is critical information for the vehicle energy management system and must be accurately estimated to ensure reliable and affordable electrified vehicles (xEV). However, due to the nonlinear temperature, health, and SOC dependent behaviour of Li-ion (FNN) approach. The method includes a description of data acquisition, data preparation, development of an FNN, FNN tuning, and robust validation of the FNN to sensor noise. To develop a robust estimator, the FNN was exposed, during training, to datasets with errors intentionally added to the data, e.g. adding cell voltage variation of ± 4 mV, cell current

Read data

0.3851

0.3852

0.3852

0.3852

0.3852

Current Temperature

0.3031

0.3046

0.3061

0.3076

0.3091

Voltage

0.7510

0.7510

0.7510

0.7510

0.7510

Simulation & Test

Deploymen

÷	i	i	i	ł	ł		
						Ε	
						Ē	
ī	I	I	I	I.	ī		





Configure	Al mode		sequenceinput sequenceInput	Data Prepara	a A ntion	Modeling	Simulation & Test
Connigure				Voltage Current 0.7510 0.3851 0.7510 0.3852	0.3031 0.3046		
Train and	Test AI m	nodel	fc_1 fullyConnected	0.7510 0.3852 0.7510 0.3852 0.7510 0.3852	0.3061 0.3076 0.3091		
Training Options	– 🗆 X						
SOLVER			layer				
Solver	sgdm 🔻		tanhLayer				
InitialLearnRate	0.01 -						
BASIC					A 5	кі — I	Б.:
ValidationFrequency	50 -		fc_2		🦚 Deep	Network	c Designer
MaxEpochs	30 🜩				TDAIN	UNIC .	
MiniBatchSize	128				TRAIN	IIING	
ExecutionEnvironment	auto 💌						
SEQUENCE			leakyrelu leakyReluLayer		(Q):		
SequenceLength	longest •				- · ·	- ·	
SequencePaddingValue	0				iraining	Irain	Export Training
SequencePaddingDirection	(right 💌		fc 3		Options		Plot
ADVANCED			fullyConnected		OPTIONS	TRAIN	EXPORT
L2Regularization	0.0001				Design	er I	Data Training
GradientThresholdMethod	12norm 💌		4	I	_		_
GradientThreshold	Inf 🚖		clippedrelu				
ValidationPatience	Inf 🚊		clippedReluLayer				
Shuffle	every-epoch 💌						
CheckpointPath	Specify checkpoint path						
CheckpointFrequency	1		regressionout regressionLayer				

 \checkmark

Export

 $\overline{\mathbf{w}}$



Feed Forward NN is simple – but it has no memory

Moving average added to the input signals

Data source <u>https://data.mendeley.com/datasets/cp3473x7xv/3</u> ¹³

Results







-10°C

Prediction is good even at low temperatures

prediction ground truth

Import Pre-Trained Model



You can also import an AI model trained outside of the MathWorks ecosystem into MATLAB







Test



Simulink Library Browser ▶ Q 2 + 2 + 0 💠 💠 transmission line 🗸 🗞 🕶 🔂 🖛 🐨 🖝 🕝 SIMULATION DEBUG MODELING FORMAT Deep Learning Toolbox/Deep Neural Networks Stop Time input(end, , 🔽 Oper Mata Logic ×. 4 9 (\mathbf{b}) Normal -🔒 Save NXP Model-Based Design Toolbox for S32K3xx MCUs Signal Table Bird's-Eye Step Run Step Simulation Library Analyzer Scope Manager Simulink Inspector - 🚔 Print Browse Bast Restart Back -Forward ÷ Aerospace Blockset REVIEW RESULTS LIBRARY Audio Toolbox 10000 simulinkImplementation Automated Driving Toolbox Automotive Math and Motor Control Library for NXP S32K3 simulinkImplementation Communications Toolbox Image Classifier Communications Toolbox HDL Support \odot Computer Vision Toolbox K 3 Control System Toolbox \$ Data Acquisition Toolbox ⇒ Deep Learning Toolbox A: Deep Neural Networks Y Shallow Neural Networks 2 Control Systems Predict Net Input Functions Processing Functions Ś Transfer Functions Weight Functions DSP System Toolbox u^{T} input DSP System Toolbox HDL Support Embedded Coder Embedded Coder Support Package for ARM Cortex-M Proc Stateful Classify **Eived Daint D** File Tools View Simulation Help 51 Sample based **EixedStenDiscrete**

Simulink provides blocks with different AI functions We just parameterize them with the AI function name and feed them with signals with the predictors



With Variant Subsystems we can implement several AI functions in the same model and try them one at a time

Processor-in-the-Loop (PIL) Testing on ARM Cortex-M7 Processor



Deployment Current Temperature 0.3851 0.3031 0.7510 0.7510 0.3852 0.3046 0.3061 07510 0 3852 0.3076 07510 0 3852 0.7510 0.3852 0.3091 **Automatic Library-Free** C Codeaugmented Any CPU Texas Inc. ARM Cortex-M **INSTRUMENTS**



Finally, we can configure the model for Processor in the Loop execution 1- Configure hardware and communication ports 2- Select PIL execution 3- Code is generated for the AI function subsystem and downloaded onto the evaluation board 4- The algorithm now runs on target

Tradeoffs and Benchmark

	EKF Extended Kalman Filter	Tree Fine Regression Tree	FFN 1-hidden layer Feedforward Network	LSTM Stacked Long Short-Term Memory Network
Training Speed	N/A	\bigcirc		
Interpretability		ightarrow		
Inference Speed *				
Model Size *				
Accuracy (RMSE)				

Results are specific to this example

Here is a comparison among AI methods and the EKF benchmark There is a trade-off among training effort, predictive

accuracy, and on-target execution time

User Stories



Onboard Battery Pack State of Charge Estimation Using a Neural Network

MathWorks AUTOMOTIVE CONFERENCE 2022

 KPI1
 Math video

 7th April 2022
 Battery SOC and SOH

Estimation using a Hybrid

Machine Learning Approach

MathWorks Automotive Conference 2022 videos available on demand

Summary

- Develop AI-based Virtual Sensor for Battery SOC Estimation
- Workflow From Data Acquisition to Hardware Deployment
- Compare Different AI Methods







Thank you



© 2022 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See *mathworks.com/trademarks* for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.