
**EPE'25 – Call for Tutorials**

# Design, Validation, and Embedded Deployment of AI-Based Virtual Sensor for Battery State-of-Charge (SOC) Estimation

## Name(s) and Affiliation(s) of the Lecturer(s):

**Moubarak Gado**
**MathWorks**
2 Rue de Paris, 92196
Meudon, France
mgado@mathworks.com
+33 1 41 14 87 57

**Daniele Sportillo**
**MathWorks**
2 Rue de Paris, 92196
Meudon, France
dsportil@mathworks.com
+33 1 41 14 87 17

## Tutorial Objectives:

Virtual sensor (also known as soft sensor) modeling is a powerful technique for mimicking the behavior of a physical sensor when the signal of interest cannot be directly measured, or when a physical sensor adds too much cost and complexity to the design. When developing a Battery Management System (BMS) for electrified systems, having an accurate value of the Battery State-of-Charge (SOC) is a critical design element, and it is challenging to directly measure the SOC. Artificial Intelligence (AI) techniques can be applied as alternatives or supplements to Kalman Filters and other well-known techniques. However, the AI models must be verified and tested with other parts of the overall system to ensure reliability and safety in operation. The AI models should also satisfy compute requirements when being deployed onto resource-constrained embedded devices.

This tutorial will use the example of SOC in BMS to demonstrate the integration of AI models into system-level design, the execution of validation and verification activities, and the trade-off management between different deployment objectives. You will learn the process that involves linking design to requirements and tests for full traceability, simulating the AI model within the larger system, and applying both simulation-based testing and formal neural network verification techniques. You will also learn about compressing the AI model, evaluating performance, memory, and accuracy tradeoffs, and automatically generating code for processor-in-the-loop (PIL) testing and production deployment on embedded systems, such as a microcontroller board.

The presented workflow in this tutorial will offer you a comprehensive view of an end-to-end solution for designing, training, validating and verifying, compressing, and deploying AI-based virtual sensor models to embedded processors with MATLAB and Simulink.

## Target Audience:

Power electronics engineers and battery management system engineers looking to integrate AI algorithms into their designs for system-level testing and deployment onto embedded devices, such as microcontrollers.

## Topical Outline:

**Introduction: (Estimated time: 10 minutes)**

- Tutorial logistics and setup

**Overview: (Estimated time: 10 minutes)**

- AI trends in system design

**Tutorial - Theme 1: AI Modeling and Simulation (Estimated time: 50 minutes)**

- Develop AI model using Deep Learning in MATLAB
- Import AI model created using Python-based frameworks
- Integrate AI model in Simulink for running simulations

**Tutorial - Theme 2: Requirements Verification and Testing (Estimated time: 50 minutes)**

- Link requirements to the AI model
- Create test cases for the AI model
- Run test cases to verify requirements

**Tutorial - Theme 3: Compress AI Models and Generate Code (Estimated time: 50 minutes)**

- Perform compression on the AI model to reduce memory footprint
- Generate C code from the compressed AI model
- Profile performance of the generated code from the AI model
- Compare performance of different AI models and the impact on embedded deployment

**Tutorial - Theme 4: AI Verification (Estimated time: 30 minutes)**

- Train a discriminator for Out-of-Distribution Detection (OODD) based on input data
- Perform simulations to detect potential deviations of inputs

**Conclusions (Estimated time: 10 minutes)**

- Resources and Next Steps

## Provisional Schedule of the Tutorial:

Schedule:

09:30 – 11:00 : Introduction / Overview / Theme 1 / Theme 2
11:00 – 11:30 : Coffee break / Lunch Break
11:30 – 13:00 : Theme 3 / Theme 4 / Conclusions

**About the Lecturers:**

**Moubarak Gado** has been an application engineer at MathWorks since 2019 focusing on AI, simulation, parallel computing, and algorithm deployment for both embedded systems and cloud platforms. He holds a PhD in computational mechanics from University of Bordeaux (France) and University of the Basque country (Spain), a MSc in materials and processes from University of Lorraine & Arts et Métiers ParisTech & ENIM, as well as an Engineering Degree in mechanical engineering from ENIM.

Before joining MathWorks, he worked as a Research and Development Engineer at CT Engineering Group, where he developed AI and simulation tools for a wide range of domains and applications, including aeronautics, predictive maintenance, acoustics, fluid dynamics, mechanical systems, and structural optimization.

**Daniele Sportillo** has been part of the academic team at MathWorks since 2022. He supports professors, researchers, and students in the realization of their teaching and research projects. He holds a PhD in Robotics and Virtual Reality from Mines Paris – PSL (France), an MSc in Embedded Computing Systems from Scuola Superiore Sant'Anna (Italy), and a BSc in Computer Engineering from the University of Parma (Italy).

Prior to joining MathWorks, Daniele worked as an R&D Engineer in the automotive field, where he designed ADAS and Automated Driving algorithms.