NOVEL STATE ESTIMATION TECHNIQUES AND THEIR APPLICATIONS

Jiayi Su
Marquette University

NOVEL STATE ESTIMATION TECHNIQUES AND THEIR APPLICATIONS

By

Jiayi Su, B.S., M.S.

A Dissertation Submitted to the Faculty of the
Graduate School, Marquette University,
in Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy

Milwaukee, Wisconsin

August, 2024

ABSTRACT

NOVEL STATE ESTIMATION TECHNIQUES AND THEIR APPLICATIONS

Jiayi Su, B.S., M.S.

Marquette University, 2024

This dissertation contributes to the field by developing and applying novel state estimation techniques in three distinct areas: the detection of cyber-physical systems (CPS) intrusion, the estimation of State-of-Charge (SOC) and State-of-Health (SOH), and multi-object tracking in video sequences.

In the first area, emphasis is placed on the detection of unknown intrusion signals within CPS. A pioneering detection scheme is formulated to discern potential cyber-attacks on industrial control systems. A novel intrusion model is first introduced and subsequently incorporated into the Multi-Model Adaptive Estimation (MMAE) technique for the identification of sensor or actuator attack signals. Simulation results underscore the effectiveness of the proposed technique in detecting unknown intrusions. Furthermore, the integration of a fading memory technique into the MMAE approach expedites the detection process, simulation results verified the proposed approach.

The second area addresses the estimation of SOC and SOH of Lithium-ion batteries. Three innovative estimation techniques are developed to precisely estimate SOC and capacity of Lithium cells in an online fashion. Initially, an adapted MMAE technique is employed for online SOC estimation, and subsequent improvement is achieved by combining MMAE with EKF to enhance accuracy and reduce computation cost. Finally, a simultaneous SOC and cell capacity estimation technique is introduced, demonstrating superior performance compared to existing methods under similar conditions, as validated by simulation results.

The third area includes the multi-object tracking (MOT) problem. Herein, an enhanced motion model and a novel reduced-order Kalman filter are introduced to augment MOT accuracy while concurrently alleviating computational burdens. Experimental results substantiate the heightened tracking performance without incurring additional computational costs. Additionally, a novel steady-state reduced-order Kalman filter is presented, further contributing to the reduction in computation costs.

ACKNOWLEDGMENTS

Jiayi Su, B.S., M.S.

I want to express my deep gratitude to the many individuals who have contributed significantly to the successful completion of my PhD dissertation, shaping my academic and personal growth.

First and foremost, I extend my appreciation to my advisor, Dr. Edwin Yaz, whose unwavering support, mentorship, and expertise have been the bedrock of my graduate journey. His visionary guidance and encouragement have not only inspired me but also instilled the confidence to pursue my academic and professional aspirations.

I also thank Dr. Schneider, whose invaluable guidance, expert insights, and constructive feedback have significantly shaped my research ideas and methodology.

I owe a debt of gratitude to my parents, whose constant love, inspiration, and unwavering support have been constants in my life. Their belief in me and their encouragement have been instrumental in my academic and personal development.

I also want to acknowledge my labmates, Tim, Justin, and Paris, whose camaraderie and support have enriched my graduate experience.

Lastly, I appreciate the faculty, staff, and fellow students whose feedback, criticism, and encouragement have helped refine my research.

TABLE OF CONTENTS

List of Tables

List of Figures

Chapter 1: Introduction

State estimation theory plays a crucial role in various fields, ranging from robotics [4] and autonomous systems [5] to finance [6] and aerospace engineering [7]. It involves the estimation of unobserved states of a system based on available measurements, control inputs, and system dynamics. By combining noisy and incomplete observations with the system dynamics, state estimation techniques provide a means to infer the true state and provide valuable insights for decision-making, control, and prediction.

This dissertation presents an in depth introduction to state estimation theory, discussing its fundamental concepts, methodologies, and applications in many areas. Specifically, different state estimation approaches, such as Multiple Model Adaptive Estimation (MMAE) and reduced-order Kalman filter, were explored. This dissertation highlights the importance of state estimation in real-world scenarios, emphasizing its applications in cyber-physical system (CPS) attack detection [8, 9, 10], Lithium-ion cells' state estimation [2, 3, 11], and multi-object tracking (MOT) problems.

In this chapter, Section 1.1 of the literature review begins by exploring the historical background of state estimation algorithms, encompassing the evolution of the theory and its diverse applications. Subsequently, an evaluation of the strengths and weaknesses of previous state estimation techniques is provided in a concise manner.

Following that, several estimation techniques used in this dissertation for addressing the intrusion detection, the Lithium-ion cell state estimation and the MOT problems are derived. Section 1.2 provides the derivation of the Kalman filter, the Extended Kalman filter and Multi-Model Adaptive Estimation (MMAE) technique, which will be used later in this dissertation. Building upon the existing literature and estimation techniques, Section 1.3 provides a general introduction to CPS intrusion detection, Battery Management System (BMS) and the MOT problem, along with current techniques to solve those problems. The subsequent sections of the chapter are organized as follows: Section 1.4 presents the dissertation's overall structure and organization, Section 1.5 outlines the specific contributions made by this dissertation and provides a summary on previous introductions.

1.1 Literature Review on State Estimation Theory

State estimation is the process of determining the internal state of a system from its input and output [12]. It is a fundamental problem in many fields of science and engineering, such as control theory, signal processing, robotics, navigation, and power systems [12, 13, 14]. State estimation can be performed using deterministic or stochastic approaches, depending on the nature of the system and the available measurements.

Figure 1.1 shows a block diagram of a typical simple discrete-time state estimator-based control system. The state estimator computes the state estimate $\hat{x}_k$ with the given control input u_k and sensor measurement y_k. After that, a state feedback controller gain can be designed based on the given state estimate $\hat{x}_k$, and the closed-

loop control input is represented as $u_k = -K\hat{x}_k + r_k$, where r_k is the reference signal. Figure 1.1, illustrates the basic concept of state estimation-based control.

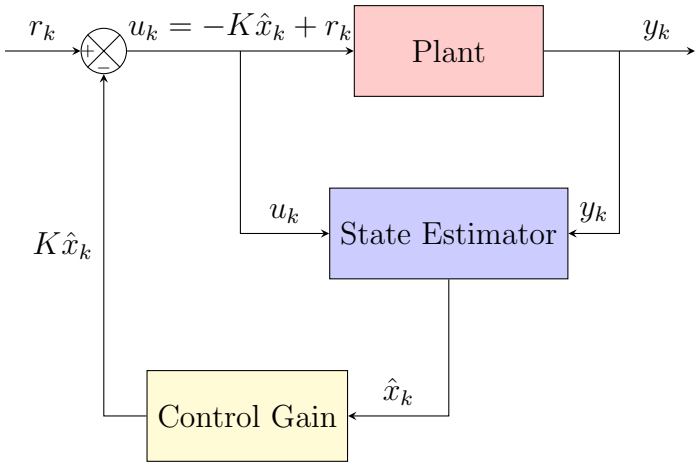

Figure 1.1: Block diagram of a simple discrete-time state estimator-based control system.

One of the earliest and most widely used methods for state estimation of linear systems is the Luenberger observer, proposed by David G. Luenberger in 1964 [15]. The Luenberger observer is a state estimator that runs in parallel with the physical system and uses the input-output measurements to correct its state estimate [16]. The Luenberger observer requires the system to be observable, meaning all state variables can be fully reconstructed from the outputs. The design of the Luenberger observer involves choosing an observer gain matrix that determines how much the error between the model output and the true output affects the state estimate [16].

In terms of the stochastic system with noisy input and measurement, the Kalman filter is usually implemented to minimize the mean square error between the estimated and true states [13]. If the noise is Gaussian distributed, an optimal estimator gain can be computed to find system state estimates. This optimality property makes the Kalman filter ideal for scenarios where the system is linear and noise is additive with known characteristics.

In terms of the nonlinear stochastic system, the Kalman filter can be modified to the Extended Kalman Filter (EKF) or the Sigma Point Kalman Filter (SPKF). Both are used to estimate states for nonlinear systems [12]. The basic idea of the EKF is to linearize the system around the current state estimate using Taylor series expansion, then the Kalman filter estimate is based on the first term of the Taylor series expansion [12]. Note the EKF is usually used when the nonlinearity of the system is mild. If the system dynamics are highly nonlinear, the first order Taylor approximation would not be accurate enough to capture the global system behavior. In this case, a second order EKF can be applied, however, the computation cost is usually increased by adding higher order terms.

Another approach for solving the nonlinear state estimation problem is the SPKF. Compared to the EKF, there is no need to linearize the system before applying SPKF [1]. Nonlinearities of the system can be propagated through the selected Sigma Points, and then the state estimates can be computed recursively [17]. A fine-tuned SPKF state estimation usually yields superior accuracy compared to the EKF. Notably, the SPKF exhibits the remarkable ability to provide robust state estimates even when the initial state estimate is imprecise. In essence, meticulous calibration

through trial and error of SPKF parameters can yield exceptionally accurate and resilient state estimations, all the while incurring a computational cost that is nearly equivalent to that of the EKF.

However, it is important to note that achieving the best SPKF performance necessitates a careful adjustment of weighting constants – a task that can be quite challenging [1, 2]. If these constants are not finely tuned, the SPKF's estimation results may fall short of those obtained with the EKF. Additionally, the crux of the SPKF's success largely hinges on one's experience, as there is no systematic knowledge available to guide the tuning of these constants. Variations of Kalman filter's adaptability to nonlinear systems offers a distinct advantage in scenarios where linear models inadequately represent the complexities of the system dynamics.

The Particle Filter (PF) stands as another powerful and versatile technique for estimating the state of a dynamic system. Particle filtering is particularly adept at handling nonlinear and non-Gaussian state estimation problems, making it invaluable in a wide range of applications, from robotics and autonomous navigation to finance and environmental modeling [18, 19, 20, 21]. The core idea behind the Particle Filter is to represent the probability distribution of the system's state using a set of discrete particles or samples [22]. Each particle encapsulates a hypothesis about the system's state at a given time. By propagating these particles through the system's dynamic model and updating their weights based on the likelihood of observed measurements, the Particle Filter provides an approximation of the posterior distribution of the state [23]. As more measurements are acquired over time, the particles converge to better representations of the true state, allowing for highly accurate and adaptable

estimation.

However, several drawbacks still limit its usage scenarios despite its ability to estimate system states with highly nonlinear dynamics. Since the number of particles assigned to the filter is related to the computation cost, it is hard to have accurate estimation results, especially when it is running on devices with limited computation resources. Also, resampling techniques are required to maintain diversity among particles at each iteration, since some particles may become degenerate, meaning they carry very little weight or significance in the estimation process [12]. In addition, the choice of initial particle distribution can also impact the Particle filter's convergence and accuracy. Finding an appropriate initialization strategy can be challenging, especially when little prior information is available. In this case, while it offers significant advantages in handling complex and dynamic scenarios, its computational complexity, sensitivity to initialization, and potential challenges related to high-dimensional state spaces should be carefully considered in its application.

Some variants of the aforementioned techniques have also been investigated. For example, to improve the computational efficiency and reduce the real-time processing time, the reduced-order Kalman filter and steady-state Kalman filter are introduced. These can be implemented on devices with limited computational resource, or can be used to improve the processing time [24].

The reduced-order Kalman Filter is commonly used when a subset of the state variables can be measured directly [12]. In this case, there is no need to estimate all states of the system. Hence, a reduced-order estimator can be designed to only estimate the remaining state variables. By using the reduced-order Kalman Filter, a

balance between estimation accuracy and computational efficiency can be achieved. It is particularly well-suited for applications in which high-dimensional state space models or real-time processing constraints pose challenges [25]. The strength of this filter lies in its ability to reduce the dimensionality of the state space while still providing accurate and reliable state estimates. This reduction in computational requirements translates into faster estimation updates and real-time capabilities.

The steady-state Kalman filter is another approach used when the computational resource are limited [13, 14]. Different from the reduced-order Kalman filter, the steady-state Kalman filter computes the steady-state Kalman gain and state error covariance matrix offline. In this case, the estimator can be iterated by using the steady-state (constant) Kalman gain, hence eliminating the need for continuous Kalman gain and covariance matrix updates. Compared to the traditional Kalman filter, the Steady-State Kalman Filter calculates these parameters once and assumes that they remain constant over time. By doing this, it significantly reduces the computational burden compared to the traditional Kalman filter. It provides better memory efficiency compared to the regular Kalman filter due to its steady-state Kalman gain [26]. Similar to the reduced-order Kalman filter, this feature is advantageous in resource-constrained environments.

Another advanced estimation technique is Multi Model Adaptive Estimation (MMAE), MMAE is an adaptive estimation technique that can be used for system parameter identification, or system state estimation [2, 3, 11, 12, 13]. It estimates system state variables or parameters by employing a set of distinct models to capture different operating modes or behaviors.

The essence of MMAE lies in its ability to maintain multiple candidate models, each of which represents a hypothesis about the system's behavior. These models can encompass a wide range of dynamics and operating conditions, allowing the estimator to be more adaptive compared to traditional estimation approaches. By simultaneously considering various potential system behaviors, MMAE offers a more comprehensive and accurate estimation of the true system parameter and state.

MMAE has found applications across diverse domains, including aerospace [27], autonomous navigation [28], robotics [29] and Battery Management Systems (BMS) [2, 3, 11]. It has proven especially valuable in scenarios where systems exhibit abrupt changes, nonlinearities, or uncertainties that challenge the assumptions of single-model estimators like the Kalman filter. For example, in the Lithium-ion battery cell BMS estimation problem, MMAE can effectively handle changes in state-of-charge, state-of-health and sensor reliability, ensuring safe and reliable operation [2, 3, 11].

Other estimation techniques also play an important role in different applications. For example, the H-infinity filter is usually used for applications where system modeling is imprecise or where the system dynamics are subject to significant variations. Robust Kalman filtering is often applied to address the challenges posed by noise and parameter uncertainties. The Kalman filter with delayed measurements is designed in the case where sensor measurements may not be available in real-time and can be subject to delays due to various factors, such as communication lag or sensor processing time [12, 13].

1.2 Introduction to the Estimation Techniques Used in This Work

1.2.1 Kalman Filter

Consider the system

$$x_{k+1} = A_k x_k + B_k u_k + F_k v_k \tag{1.1}$$

$$y_k = C_k x_k + D_k u_k + G_k w_k \tag{1.2}$$

where $A_k, B_k, C_k, D_k, F_k, G_k$ are system dynamics, $x_k \in \mathbb{R}^n$ and y_k represent system state variables and measurements, v_k, w_k are system state and measurement noise with statistics defined by

$$v_k \sim N(0, V_k) \tag{1.3}$$

$$w_k \sim N(0, W_k) \tag{1.4}$$

where V_k, W_k represent system state and measurement noise covariance, and v_k, w_k are white noise (uncorrelated in time).

To estimate the system state $\hat{x}_k$ with given control input u_k and output measurement y_k, the state estimator can be written as

$$\hat{x}_{k+1} = A_k \hat{x}_k + B_k u_k + K_k(y_k - \hat{y}_k) \tag{1.5}$$

where K_k is the estimator gain (also known as Kalman gain), and

$$\hat{y}_k = C_k \hat{x}_k + D_k u_k \tag{1.6}$$

Note in (1.5), once the estimator gain K_k is found, the state estimate $\hat{x}_k$ can be found recursively by initializing $\hat{x}_0$. This can be done by minimizing the expected value of the estimation error covariance $P_{k+1} = E\{(e_{k+1})(e_{k+1}^T)\}$, where

$$e_{k+1} = x_{k+1} - \hat{x}_{k+1} \tag{1.7}$$

Substitute (1.1) and (1.5) into (1.7):

$$\begin{aligned} e_{k+1} &= [A_k x_k + B_k u_k + F_k v_k] - [A_k \hat{x}_k + B_k u_k + K_k(y_k - \hat{y}_k)] \\ &= (A_k - K_k C_k) e_k + F_k v_k - K_k G_k w_k \end{aligned} \tag{1.8}$$

Substitute (1.8) into the error covariance equation

$$\begin{aligned} P_{k+1} &= E\{(e_{k+1})(e_{k+1}^T)\} \\ &= A_k P_k A_k^T - A_k P_k C_k^T K_k^T - K_k C_k P_k A_k^T \\ &+ K_k C_k P_k C_k^T K_k^T + F_k V_k F_k^T + K_k G_k W_k G_k^T K_k^T \end{aligned} \tag{1.9}$$

Here, the trace of the error covariance matrix P_{k+1} can be minimized to determine the Kalman gain K_k. Since P_{k+1} is a diagonal and symmetric matrix (v_k, w_k are assumed to be uncorrelated, thus the off-diagonal entries of V_k, W_k are all zero), minimizing P_{k+1} is equivalent to minimize the trace of P_{k+1}. In this case, the Kalman gain

K_k can be found by taking the partial derivative of $Tr\{P_{k+1}\}$ with respect to the Kalman gain:

$$\frac{\delta Tr\{P_{k+1}\}}{\delta K_k} = -2A_k P_k C_k^T + 2K_k(C_k P_k C_k^T + G_k W_k G_k^T) \tag{1.10}$$

The Kalman gain, K_k, can be found by setting (1.10) to be zero, which yields

$$K_k = A_k P_k C_k^T (C_k P_k C_k^T + G_k W_k G_k^T)^{-1} \tag{1.11}$$

By substituting (1.11) into (1.9),P_{k+1} can be simplified as

$$P_{k+1} = A_k P_k A_k^T - A_k P_k C_k^T (C_k P_k C_k^T + G_k W_k G_k^T)^{-1} C_k P_k A_k^T + F_k V_k F_k^T \tag{1.12}$$

In summary, the Kalman filter is shown as:

Algorithm 1 Summary of Kalman filter

1: **Initialization**: Initialize the state estimate $\hat{x}_0$ and error covariance matrix P_0
2: **for** $k = 0, 1, 2, \ldots$ **do**
3: $\quad K_k = A_k P_k C_k^T (C_k P_k C_k^T + G_k W_k G_k^T)^{-1}$
4: $\quad \hat{x}_{k+1} = A_k \hat{x}_k + B_k u_k + K_k(y_k - \hat{y}_k)$
5: $\quad P_{k+1} = A_k P_k A_k^T - A_k P_k C_k^T (C_k P_k C_k^T + G_k W_k G_k^T)^{-1} C_k P_k A_k^T + F_k V_k F_k^T$
6: **end for**

1.2.2 Extended Kalman filter

Consider the nonlinear system

$$x_{k+1} = f(x_k, u_k, v_k) \tag{1.13}$$

$$y_k = h(x_k, u_k, w_k) \tag{1.14}$$

where $f()$ and $h()$ represent the nonlinear system state and output function. Similar to the linear system shown in (1.1) and (1.2), x_k is the system state, u_k is the control signal, $v_k \sim N(0, V_k)$ and $w_k \sim N(0, W_k)$ are system state and measurement noise. To estimate x_k, an extension to the linear Kalman filter is made, so that it can be adapted for the nonlinear system. Specifically, system dynamics (1.13) and (1.14) are linearized using Taylor series expansion

$$A_k = \left.\frac{\partial f}{\partial x}\right|_{x=\hat{x}_k, u=u_k, v_k=\bar{v}} \tag{1.15}$$

$$C_k = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}_k, u=u_k, w_k=\bar{w}} \tag{1.16}$$

$$F_k = \left.\frac{\partial f}{\partial v}\right|_{x=\hat{x}_k, u=u_k, v_k=\bar{v}} \tag{1.17}$$

$$G_k = \left.\frac{\partial h}{\partial w}\right|_{x=\hat{x}_k, u=u_k, w_k=\bar{w}} \tag{1.18}$$

where A_k, C_k, F_k, G_k are linearized system dynamics evaluated at current state estimate $\hat{x}_k$, control signal u_k, and the mean of state or measurement noise, $\bar{v}$ or $\bar{w}$

respectively.

With the given linearized system dynamics, the linear Kalman filter can be extended for nonlinear system state estimation, and the state estimator is modified as

$$\hat{x}_{k+1} = f(\hat{x}_k, u_k, \bar{v}_k) + K_k(y_k - h(\hat{x}_k, u_k, \bar{w}_k)) \tag{1.19}$$

In summary, the extended Kalman filter is shown as:

Algorithm 2 Summary of extended Kalman filter

1: **System Linearization**: Linearize the system, find A_k, C_k, F_k, G_k matrices
2: **Initialization**: Initialize the state estimate $\hat{x}_0$ and error covariance matrix P_0
3: **for** $k = 0, 1, 2, \ldots$ **do**
4: $\quad K_k = A_k P_k C_k^T (C_k P_k C_k^T + G_k W_k G_k^T)^{-1}$
5: $\quad \hat{x}_{k+1} = f(\hat{x}_k, u_k, \bar{v}_k) + K_k(y_k - h(\hat{x}_k, u_k, \bar{w}_k))$
6: $\quad P_{k+1} = A_k P_k A_k^T - A_k P_k C_k^T (C_k P_k C_k^T + G_k W_k G_k^T)^{-1} C_k P_k A_k^T + F_k V_k F_k^T$
7: **end for**

In summary, the EKF is an extension of the linear Kalman filter, developed for estimating state variables for nonlinear systems. Compared to the linear Kalman filter, the nonlinear system is first linearized with Taylor series expansions to approximate the system and measurement models around the equilibrium point. Then the extended Kalman gain and the error covariance matrix can be updated at each iteration. Finally, state estimates can be computed with the given Kalman gain.

Note, while the EKF can provide reasonable estimates for mildly nonlinear systems by locally linearizing the models, its performance may deteriorate as the degree of nonlinearity increases and the linearization errors become more significant [12]. In highly nonlinear scenarios, the EKF may not provide reliable results due to those issues. In this case, another nonlinear state estimator, such as the SPKF or the Particle filter are usually used.

1.2.3 Multi-Model Adaptive Estimator

In the MMAE technique, the unknown parameter of a system, can be estimated adaptively using the control signal and the measurement data. The unknown parameter is quantized as a discrete set of N possible values. With the given set of the quantized parameter candidates, N estimators can be designed, each of which specifically targets a possible parameter candidate from among this set. The estimators are then grouped as a bank with a common input. The corresponding conditional probability for each possible parameter candidate can be adaptively calculated using Bayes' rule. The specific filter with the highest conditional probability is selected as the most likely parameter at the current iteration. [2, 3, 11, 12, 13].

Consider the system with an unknown parameter θ, which is quantized to a finite number of parameter candidates $\{\theta_1, \theta_2, \ldots, \theta_i, \ldots, \theta_N\}$:

$$x_{k+1|\theta_i} = A_{k|\theta_i} x_{k|\theta_i} + B_k u_k + F_k v_k \tag{1.20}$$

$$y_{k|\theta_i} = C_{k|\theta_i} x_{k|\theta_i} + D_k u_k + G_k w_k \tag{1.21}$$

where $i = 1, 2, \ldots, N$. Each θ_i have an assumed or known initial *a priori* probability

$p(\theta_i \mid Y_0)$, where Y_k represents all measurement up through time k.

Given the system model with quantized unknown parameter candidate, N Kalman filters can be assigned, one for each model:

Algorithm 3 Kalman filters designed for quantized unknown parameter candidates

1: **Initialization**: Initialize the state estimate $\hat{x}_{0,\theta_i}$ and error covariance matrix P_{0,θ_i} for each quantized parameter candidate θ_i

2: **for** $k = 0, 1, 2, \ldots$ **do**

3: $\quad \Omega_{k|\theta_i} = C_{k|\theta_i} P_{k|\theta_i} C_{k|\theta_i}^T + G_k W_k G_k^T$

4: $\quad K_{k|\theta_i} = A_{k|\theta_i} P_{k|\theta_i} C_{k|\theta_i}^T \Omega_{k|\theta_i}^{-1}$

5: $\quad \hat{x}_{k+1|\theta_i} = A_{k|\theta_i} \hat{x}_{k|\theta_i} + B_k u_k + K_{k|\theta_i}(y_k - \hat{y}_{k|\theta_i})$

6: $\quad \hat{y}_{k|\theta_i} = C_{k|\theta_i} \hat{x}_{k|\theta_i} + D_k u_k$

7: $\quad P_{k+1|\theta_i} = A_{k|\theta_i} P_{k|\theta_i} A_{k|\theta_i}^T + F_k V_k F_k^T - K_{k|\theta_i} C_{k|\theta_i} P_{k|\theta_i} A_{k|\theta_i}^T$

8: **end for**

Since the unknown parameter is quantized to a finite number of grid points, the conditional probability of the each parameter given the set of all measurements Y_k up through time k can be found using Bayes' rule:

$$p(\theta_i \mid Y_k) = \frac{p(y_k \mid Y_{k-1}, \theta_i) p(\theta_i \mid Y_{k-1})}{\sum_{i=1}^{N} p(y_k \mid Y_{k-1}, \theta_i) p(\theta_i \mid Y_{k-1})} \tag{1.22}$$

In (1.22), $p(\theta_i \mid Y_{k-1})$ is the *a priori* conditional probability of $p(\theta_i \mid Y_k)$, and (1.22)

can be solved recursively once $p(y_k \mid Y_{k-1}, \theta_i)$ and $p(\theta_i \mid Y_{k-1})$ are known. The initialization of the *a priori* probability $p(\theta_i \mid Y_0)$ is simply by design. Note the sum of all *a priori* probabilities tuned to its corresponding parameter is one.

In order to update $p(\theta_i \mid Y_k)$, the probability density function $p(y_k \mid Y_{k-1}, \theta_i)$ needs to be found. It is shown that the convergence of the *a posteriori* probability is independent of the probability density function [13, 30]. Therefore, all state and measurement noise can be assumed to have a Gaussian distribution, which produces Gaussian conditional probabilities. In this case, $p(y_k \mid Y_{k-1}, \theta_i)$ can be represented as

$$\begin{aligned} p(y_k \mid Y_{k-1}, \theta_i) = (2\pi)^{-n/2} \left|\Omega^{-1}_{k|\theta_i}\right|^{1/2} \cdot \\ \exp\left(-\frac{1}{2}\tilde{y}^T_{k|\theta_i}\Omega^{-1}_{k|\theta_i}\tilde{y}_{k|\theta_i}\right) \end{aligned} \tag{1.23}$$

In (1.23), n is the order of the system dynamics, $\tilde{y}_{k|\theta_i}$ is the innovation sequence with its corresponding θ_i, represented as

$$\tilde{y}_{k|\theta_i} = y_k - \hat{y}_{k|k-1,\theta_i} \tag{1.24}$$

where $\hat{y}_{k|k-1,\theta_i}$ is the estimated measurement signal (also called innovation) associated with the Kalman filter with its corresponding parameter candidate.

In (1.23), $\Omega_{k|\theta_i}$ is the innovation covariance and is presented as

$$\begin{aligned} \Omega_{k|\theta_i} &= E\left[(y_k - \hat{y}_{k|k-1,\theta_i})(y_k - \hat{y}_{k|k-1,\theta_i})^T\right] \\ &= C_k P_{k|\theta_i} C_k^T + G_k W_k G_k^T \end{aligned} \tag{1.25}$$

where $P_{k|\theta_i}$ is the error covariance matrix which can be computed recursively using the Riccati equation from its corresponding Kalman filter equations.

Proof of convergence of the MMAE technique can be found in [13], and the pseudocode of the MMAE technique is generalized as below:

Algorithm 4 MMAE algorithm

Initialization 1: Quantize the unknown parameter to a finite set: $\{\theta_1, \theta_2, \ldots, \theta_i, \ldots, \theta_N\}$. (Note: N is the number of filters)

Initialization 2: Initialize the state estimate, $x_{0|\theta_i}$, the *a priori* probabilities, $p(\theta_i \mid Y_0)$ and the error covariance, $P_{0|\theta_i}$ for all filters.

for $k = 0, 1, 2...$ **do**

  for $i = 1...N$ **do**

    $\Omega_{k|\theta_i} = C_{k|\theta_i} P_{k|\theta_i} C_{k|\theta_i}^T + G_k W_k G_k^T$ ▷ Set up Kalman filters for each quantized parameter candidate

    $K_{k|\theta_i} = A_{k|\theta_i} P_{k|\theta_i} C_{k|\theta_i}^T \Omega_{k|\theta_i}^{-1}$

    $\hat{x}_{k+1|\theta_i} = A_{k|\theta_i} \hat{x}_{k|\theta_i} + B_k u_k + K_{k|\theta_i}(y_k - \hat{y}_{k|\theta_i})$ ▷ Conditional state estimate

    $\hat{y}_{k|\theta_i} = C_{k|\theta_i} \hat{x}_{k|\theta_i} + D_k u_k$

    $P_{k+1|\theta_i} = A_{k|\theta_i} P_{k|\theta_i} A_{k|\theta_i}^T + F_k V_k F_k^T - K_{k|\theta_i} C_{k|\theta_i} P_{k|\theta_i} A_{k|\theta_i}^T$

    $\tilde{y}_{k|\theta_i} = y_k - \hat{y}_{k|\theta_i}$

    $p(y_k \mid Y_{k-1}, \theta_i) = (2\pi)^{-n/2} \left| \Omega_{k|\theta_i}^{-1} \right|^{1/2} \exp\left(-\frac{1}{2} \tilde{y}_{k,\theta_i}^T \Omega_{k,\theta_i}^{-1} \tilde{y}_{k,\theta_i}\right)$

    $p(\theta_i \mid Y_k) = \frac{p(y_k|Y_{k-1},\theta_i) p(\theta_i|Y_{k-1})}{\sum_{i=1}^N p(y_k|Y_{k-1},\theta_i) p(\theta_i|Y_{k-1})}$ ▷ Compute the *a posteriori* probabilities for each parameter candidate

  end for

  The most likely unknown parameter can be determined based on $p(\theta_i \mid Y_k)$

  $\hat{x}_{k+1|k} = \sum_{i=1}^N p(\theta_i \mid Y_k) \hat{x}_{k+1|,\theta_i}$ ▷ Blended state estimate

  $P_{k+1} = \sum_{i=1}^N p(\theta_i \mid Y_k)(P_{k+1,\theta_i} + [\hat{x}_{k+1|k,\theta_i}][\hat{x}_{k+1|k,\theta_i}]^T) - [\hat{x}_{k+1|k}][\hat{x}_{k+1|k}]^T$ ▷ Blended covariance update

end for

1.3 Introduction to the Problems Investigated in this Dissertation

This dissertation tackles three distinct problems across different domains by employing various estimation techniques described in this work. Each problem presents unique challenges and calls for tailored solutions.

The first problem revolves around Cyber-Physical intrusion detection, where a proposed estimation technique is designed specifically to detect the sensor and actuator intrusion with different types of attack signals. This problem is driven by the escalating threats to critical systems and infrastructure. Current solutions in the field are often limited in their ability to detect intricate cyber-physical intrusions, prompting the need for advanced estimation techniques [31].

The second problem centers on the estimation of critical states within Lithium-ion batteries, specifically focusing on the State-of-Charge (SOC) and State-of-Health (SOH) of LiFePO4 cells. This investigation is rooted in the growing importance of efficient energy storage and electric vehicle battery technology. The proposed estimation technique seeks to provide precise estimates of SOC and SOH, crucial for optimizing battery performance and lifespan. To assess the efficacy of the proposed approach, estimation results are compared with the EKF across various scenarios. This comparative analysis serves to underscore the effectiveness of the newly introduced method.

The third problem explored in this dissertation pertains to the challenging domain of Multi-Object Tracking in videos. This problem finds applications across diverse fields, from surveillance to autonomous navigation and sports analytics. The

proposed research endeavors to enhance tracking accuracy while mitigating computational demands. To achieve this, an improved motion model and a reduced-order Kalman filter are introduced, aiming at striking a balance between precision and computational efficiency.

In this section, a brief overview of each problem is first introduced, and then each of these problems will be comprehensively described in later chapters, providing a deeper dive into the background, motivations, and existing solutions. The object is to shed light on the intricacies of these challenges and to showcase how the proposed estimation techniques address them effectively.

1.3.1 Introduction to Cyber-Physical Systems Intrusion Detection Problem

In today's interconnected world, where the boundaries between the digital and physical realms continue to blur, the importance of safeguarding our critical infrastructure and systems cannot be overstated. The convergence of cyberspace and the physical world has given rise to what is known as the Cyber-Physical System (CPS), where the integration of computational elements and physical processes opens the door to unprecedented opportunities and vulnerabilities. In this dynamic landscape, the CPS intrusion detection problem looms large as a paramount concern for governments, industries, and individuals alike [32].

CPSs, at their core, are complex, interconnected systems that span a wide array of applications – from smart cities and autonomous vehicles to industrial automation and healthcare [31]. These systems bridge the gap between the virtual world of information technology (IT) and the tangible world of operational technology (OT).

They offer unparalleled benefits in terms of efficiency, automation, and convenience, but they also present novel challenges, particularly in terms of security [32].

The CPS intrusion detection problem stems from the growing vulnerability of CPS to malicious actors who seek to compromise these systems for a variety of reasons, including economic gain, political motives, or simply to sow chaos [33]. As these systems become increasingly integrated into our daily lives, the risks associated with cyber-physical intrusions become more pronounced and the consequences more severe. From disrupting critical infrastructure like power grids and water supplies to compromising the safety of autonomous vehicles and medical devices, the potential impacts of such intrusions are far-reaching and potentially catastrophic [34].

Cyber-physical intrusions can manifest in various forms, including:

Malware and Software Exploitation: Attackers may infiltrate the cyber component of a CPS through malware, viruses, or exploiting software vulnerabilities [35]. Once inside, they can manipulate data, disrupt operations, or even take control of physical assets [36].

Physical Tampering: In some cases, attackers may physically tamper with components of a CPS, such as sensors, actuators, or control systems [37]. This can involve the installation of rogue devices, bypassing security measures, or even sabotaging equipment [38].

Network Intrusions: As CPSs rely on networks for communication and data transfer, infiltrating these networks is a common attack vector [39]. Attackers may intercept data, launch Distributed Denial of Service (DDoS) attacks, or gain unauthorized access to networked devices [40].

In this work, CPS intrusion signals are detected using the proposed estimation technique, aiming to prevent the physical system from being corrupted by the intrusion signal. Specifically, detection on false data injection for both the sensor and the actuator side will be considered. Several types of false signals, such as ramp and parabola will be injected into both the sensor and the actuator signals to test the effectiveness of the proposed detection technique. Details on this work will be described in Chapter 2.

1.3.2 Introduction to the Battery Management System Problem

In our rapidly evolving world, where the demand for portable and sustainable energy sources is ever-increasing, Lithium-ion batteries have emerged as indispensable components of modern life [41]. From powering our smartphones and electric vehicles to storing renewable energy, Lithium-ion batteries play a pivotal role in our quest for efficiency and sustainability [42]. However, the efficient utilization and longevity of batteries are not guaranteed without proper management. This is where the Battery Management System (BMS) comes into play.

The BMS problem encompasses a multifaceted set of challenges, ranging from battery modeling and SOC estimation to SOH assessment, state-of-power (SOP) control, cell balancing, thermal management, and advanced modeling techniques [1]. Specifically, accurate cell SOC and SOH estimation are critical to the effectiveness and safety of the BMS, since other BMS functions are all developed based on accurate SOC and SOH estimation results [2, 3, 11].

SOC is a critical metric that quantifies the amount of charge remaining in a

battery relative to its maximum capacity [1]. Accurate SOC estimation is essential for preventing overcharging, which can damage the battery, and avoiding deep discharge, which can reduce its lifespan. BMS solutions employ a variety of techniques, including voltage and current integration, coulomb counting, and advanced algorithms based on battery models, to estimate SOC with high precision [43].

SOH reflects the overall health and performance degradation of a battery over time [1]. It is essential for predicting the battery's remaining useful life and replacing it when necessary. SOH assessment encompasses tracking factors such as capacity fade, impedance growth, and chemical degradation [3]. Many methods, from simple least-square estimation to machine learning algorithms, are employed to monitor and assess SOH [1, 44].

In this work, an MMAE based SOC and SOH estimation technique is introduced, with the aim of providing accurate and reliable SOC and SOH estimation results. Specifically, lab data acquired from an LiFePO4 battery is used to build the cell model, and the proposed MMAE technique is tested based on the given cell [2, 3, 11]. Results of this estimator are compared with the EKF to demonstrate the effectiveness of the proposed technique. Details on this work will be presented in Chapter 3.

1.3.3 Introduction to the Multi-Object Tracking Problem in Video Sequences

In today's digital age, the ubiquity of video cameras has ushered in an era where vast amounts of visual data are captured continuously from diverse sources, including security cameras, drones, and smartphones. Within this ever-expanding sea of imagery lies a challenging computational problem that holds immense importance

across various domains: the Multi-Object Tracking (MOT) problem in videos. This problem encapsulates the intricate task of autonomously identifying and following multiple objects as they move and interact within video sequences, revealing insights, enhancing safety, and empowering applications ranging from surveillance to autonomous navigation [45].

The MOT problem extends far beyond simple object detection. While object detection algorithms can identify objects in individual frames, tracking involves linking these detections across consecutive frames, effectively following the objects' trajectories over time [46]. This intricate task has gained prominence due to its relevance in numerous real-world scenarios, including:

Surveillance and Security: In security applications, tracking multiple individuals or vehicles across video feeds helps monitor and respond to potential threats, track suspicious activity, and enhance situational awareness [47]. For example, a tracking system can not only be helpful for police department to locate the suspect once violence is detected, it can also be used to count people and traffic flow for the rest of the time [48].

Autonomous Vehicles: For self-driving cars and unmanned aerial vehicles (UAVs), tracking neighboring vehicles, pedestrians, and other objects is essential for safe navigation and collision avoidance [49].

Sports Analysis: In sports broadcasting and analytics, tracking players and the ball provides valuable insights for enhancing viewer experience and strategic analysis [50].

Retail and Marketing: Tracking customer behavior within stores or commercial

spaces aids in understanding shopping patterns and optimizing store layouts [51].

However, objects in videos may change appearance due to illumination variations, occlusions, or changes in pose [46]. Objects can be partially or fully occluded by other objects or environmental factors, making tracking challenging [52]. Additionally, objects may interact with one another, this leads to complex motion patterns that require sophisticated algorithms for accurate tracking [53]. In addition to rigid objects like vehicles, tracking may involve non-rigid and deformable objects, such as pedestrians and animals [54].

Addressing these challenges requires accurate vision detector and motion model, and a reliable state estimator. The most frequently used framework is tracking by detection, where objects are first detected using a computer vision technique, and a state estimator is then assigned to each objects to predict its trajectory [46]. Following this framework, many methods have been proposed to improve tracking accuracy [55, 56, 57, 58].

In this dissertation, the MOT problem in videos delves into the intricacies of object tracking across dynamic visual scenes. Within the framework of tracking by detection, two efficient techniques are introduced in this work to enhance tracking accuracy and reduce computational cost. Specifically, an improved motion model is derived to boost tracking performance without adding computational costs, and a reduced-order Kalman filter is designed to decrease computational complexity while maintaining tracking accuracy equivalent to that of the full-order Kalman filter. Experimental results are shown to demonstrate its effectiveness. Details on this work will be presented in Chapter 4.

1.4 Overall Structure and Organization of this Dissertation

In this dissertation, three different applications using estimation techniques are introduced. Chapter 2 delves into the intricacies of the CPS intrusion detection problem. The introduction to the challenge of safeguarding CPS from intrusion signals will be presented first. Then the proposed estimation technique for detecting these signals is discussed in detail. Chapter 2 also explores the methodology involving false data injection for both sensor and actuator sides, providing a rigorous test for the effectiveness of the proposed detection technique.

In Chapter 3, the focus shifts to the Lithium-ion cell state estimation problem, specifically centering on the estimation of SOC and SOH in LiFePO4 batteries. The chapter begins with a comprehensive introduction to the significance of accurate battery state estimation in various applications. Then the proposed MMAE-based estimation technique is introduced, with a detailed account of the methodology, including the use of lab data to build the cell model. After that, simulations comparing the proposed MMAE technique and the EKF are presented, illustrating the effectiveness of the proposed approach in accurately estimating SOC and SOH.

Chapter 4 presents the MOT problem. It begins by elucidating the intricacies of object tracking in dynamic visual scenes. Then two innovative techniques designed to enhance tracking precision while reducing computational complexity are introduced. Detailed descriptions of the improved motion model and the reduced-order Kalman filter are then provided, highlighting their roles in achieving a harmonious balance between tracking accuracy and computational efficiency. The chapter concludes with

the presentation and analysis of experimental results, demonstrating the efficacy of the proposed techniques in real-world MOT scenarios.

1.5 Contributions Made by this Dissertation

This dissertation addresses three distinct applications using advanced estimation techniques. In Chapter 2, it tackles the challenge of CPS intrusion detection, presenting a proposed estimation technique that enhances security by detecting intrusion signals and rigorously testing its effectiveness against false data injection methods. Chapter 3 shifts to Lithium-ion battery state estimation, introducing an innovative MMAE-based technique to accurately estimate SOC and SOH in LiFePO4 batteries, with simulations showcasing its superiority. Chapter 4 advances Multi-Object Tracking in videos, presenting novel techniques to improve tracking precision while reducing computational complexity, contributing to advancements in surveillance, navigation, and analytics.

1.6 Summary

In this chapter, several important state estimation techniques are reviewed, followed by the derivation of some estimation techniques that will be utilized in the subsequent chapters. Then, an overview is provided for three problems that require the application of state estimation techniques, with each problem to be described in greater detail in later chapters. Finally, the overall structure of this dissertation is presented, and the contribution of this dissertation is stated.

Chapter 2: Sensor and Actuator Intrusion Detection Using Multi-Model Adaptive Estimation Method

2.1 Introduction

Cyber-Physical Systems (CPS), including Distributed Control Systems (DCS) and Supervisory Control and Data Acquisition (SCADA) systems, are integral components of infrastructure and industrial control systems [59]. These systems operate by receiving input signals from physical components through sensors and subsequently sending control signals to actuators, enabling flexible, intelligent, and efficient closed-loop control of various processes [60]. However, their widespread use also exposes them to significant vulnerabilities in the face of internet attacks, such as intrusions targeting electric power systems, autonomous vehicles, encrypted control systems and traffic networks [61, 62, 63, 64]. Such attacks can infiltrate control systems, sensors, or actuators within a CPS, enabling the injection of false signals or even controller reprogramming [65]. Consequently, the need for robust fault detection and safety protection mechanisms within a CPS becomes paramount.

Most of the aforementioned cyber attacks are called False Data Injection (FDI) attacks. FDI attacks are a class of cyber-attacks that aim to compromise the integrity and reliability of a CPS by injecting malicious or counterfeit data into the system's sensors or communication networks [66]. These attacks exploit the interconnected

nature of a CPS, where sensors collect data from the physical world and transmit it to control systems for decision-making and actuation. FDI attacks can have devastating consequences, including system destabilization, equipment damage, and misleading control decisions, making them a significant concern for critical infrastructure and industrial systems [10].

The mechanisms behind FDI attacks typically involve compromising sensors, actuators or both [8, 9]. Attackers manipulate sensor readings or inject false measurements, leading to incorrect state estimations and control actions. They may also exploit vulnerabilities in communication networks to intercept, modify, or inject counterfeit data packets, obscuring genuine information [66]. FDI attacks are stealthy and diverse, often designed to avoid immediate detection by blending malicious data with legitimate readings and gradually degrading system performance [67].

Detecting and mitigating FDI attacks require advanced cybersecurity measures, including statistical analysis, machine learning algorithms, and resilient state estimation techniques [8, 9, 67, 68]. These methods aim to identify anomalies or deviations from expected sensor and actuator behavior, recognize patterns of malicious data injection, and ensure system resilience in the face of attacks. Preventative measures such as network security, secure hardware, and access control are also essential to defend against FDI attacks and safeguard the critical operations of CPSs.

2.1.1 Previous Work

There have been numerous research efforts concentrated on analyzing the vulnerabilities of CPSs and developing detection and protection strategies against the FDI attack, where it aims at compromising the integrity of a CPS by injecting fake sensor measurements or actuator signal without being detected [69].

In [70], the possibility of executing stealthy FDI attacks in DC state estimation with no knowledge of the line parameter is addressed. The authors demonstrate that attackers can manipulate state variables in one-degree buses connected solely by a single cut line and can bias state variables in all buses within a one-degree super-bus similarly connected in the power system.

In [71], the cyber security provided by robust power system state estimation methods is investigated. It identifies attack intervals and positions required for each scenario and highlights conditions that minimize detection with robust state estimation approaches.

In [72], the parity space based fault detection monitoring system is analyzed, where it reveals a cyber attack can evade detection by the fault detection system based on the parity space approach. The conditions for the existence of completely stealthy cyber attacks are also investigated, and recommendations to system developers on preventing such attacks are offered.

2.1.1.1 State Estimation-Based Detection Techniques

In [60], the cyber security of state estimators in power grid SCADA systems is considered. Trade-offs between model accuracy and attack impact for different bad

data detection (BDD) schemes are investigated.

In [73], a unique form of FDI attack known as "replay attacks" is described. In these attacks, a hacker injects external input while simultaneously replaying previously recorded measurements of the system's data.

In [62], a sliding mode observer is introduced to identify and estimate cyber-attacks on wireless communications between vehicles. It demonstrates the observer's stability and detection threshold robustness, especially in the context of event-triggered communication using a real-world Vehicle-to-Vehicle network protocol.

In [74], cyber attacks on the controller are investigated by means of optimization techniques in order to determine the worst-case scenario. A novel attack detector based on limit checking is then introduced, where no specific controller knowledge is necessary.

In [75], a platoon of connected vehicles with Cooperative Adaptive Cruise Control (CACC) and FDI are studied, where the attack introduces ghost vehicles into the network to disrupt system performance. A Partial Differential Equation (PDE) model and a diagnostic scheme based on PDE observers are used to detect and locate the attack's injection point in the platoon.

2.1.1.2 Data Driven Detection Techniques

In [67], a data-driven method is presented to addresses the challenge of stealthy CPS attacks without knowing the CPS model parameters or sensor measurements. A closed-loop recursive identification strategy is designed for the CPS's dynamic characteristics for the FDI problem.

In [76], cyber attack vulnerabilities in Connected and Autonomous Vehicles (CAVs) are investigated, where a detection framework based on a physics-informed neural network (PINN) is proposed. By learning uncertain parameters from the physics model, it infers attack scenarios.

In [77], cyber attacks in smart grids are detected using data-driven anomaly detection methods, particularly in the context of electricity market data and locational marginal prices (LMPs). A novel data-driven probabilistic anomaly detection framework is designed and tested against two types of cyber attacks using a model-based electricity market simulator.

In [78], a measured output data-driven cyberattack detection framework is developed by frequently employing a feedforward neural network during the closed-loop process, identifying the cyberattacks from the interactive network-level dynamics.

2.1.1.3 Other Types of Detection Techniques

In [61], a unified framework and advanced monitoring procedures for future power networks is introduced to detect and identify malfunctioning network components or corrupted measurements caused by cyber-physical attacks. The system is modeled as a linear time-invariant descriptor system with unknown inputs, considering various attack scenarios.

In [63], a detection method is proposed to enhance the security of control systems protected by resilient homomorphic encryption (RHE). Specifically, a warning signal will be triggered when an additive attack is detected within the resilience range and an alarm signal for attacks outside this range, utilizing the inner product's symmetric

property.

In [64], cyber attacks on a strip of freeway traffic network controlled by remote ramp-metering is studied. A design framework for cyber attack detection algorithms is proposed, considering stability, robustness, and attack sensitivity.

In [79], the dynamic nature of power system operating conditions are used to implement an active defense method through dynamic clustering. By forming clusters of measurements based on their dynamic responses to disturbances, the strategy detects stealthy cyber attacks via similarity checks within each cluster.

In [80], a set-theoretic framework for detecting bias injection cyber-attacks in the load frequency control loop of networked power systems is introduced. The detection mechanism relies on convex and compact polyhedral robust invariant sets to alarm a potential security breach.

In [81], a distributed attack mitigation defense framework with a dual-mode control system reconfiguration scheme is proposed to prevent a compromised platoon member from causing collisions via message falsification attacks.

2.1.2 Contribution to the CPS Detection Problem

In this work, the primary contribution's are:

An MMAE technique is designed to detect unknown intrusion signals: recognizing that the type of intrusion signal used to attack CPS components can be unpredictable, this work introduces a method capable of detecting unknown intrusion signals effectively.

Detection of intrusion signals when the target (sensor or actuator) is unknown:

Acknowledging that it is often unclear whether an attack will target sensors or actuators within a CPS, this work employs the MMAE technique for the healthy CPS, sensor intrusion, actuator intrusion, and combined intrusion signals to detect and specify the intrusion signal. Simulations confirm the effectiveness of identifying intrusion signals when sensors, actuators, or both are under attack.

The subsequent sections of this chapter provide a detailed exploration of the proposed contributions, beginning with Section 2.2, where a generalized mathematical model is presented for both the healthy CPS and the CPS under attack. In Section 2.3, the focus shifts to the adaptive intrusion detection technique based on MMAE. Section 2.4 is where simulation results are presented, illustrating the algorithm's robustness in the presence of unknown intrusion signals. These results also demonstrate the algorithm's ability to identify the compromised portion of the CPS, whether sensors, actuators, or both, are under attack. Finally, in Section 2.5, a comprehensive conclusion is provided, and potential directions for future research in this critical field are outlined.

2.2 Problem Formulation

In this section, a healthy CPS as a discrete-time system using a state-space representation with additive Gaussian noise is introduced. In addition, the intrusion signal is modeled as a state-space representation. The model with arbitrary analytic intrusion signals is established using the same method when the CPS is under a sensor, an actuator, or a combined attack.

2.2.1 The Healthy CPS Model

Consider a CPS modeled as a Linear Time Invariant (LTI) discrete-time stochastic system with additive state and measurement noise:

$$x_{k+1} = Ax_k + Bu_k + Fv_k \tag{2.1}$$

$$y_k = Cx_k + Du_k + Gw_k \tag{2.2}$$

where $x_k \in \mathbb{R}^n$ represents the CPS state vector, $y_k \in \mathbb{R}^p$ is the CPS measurement, $u_k \in \mathbb{R}^m$ is the control input, A, B, C, D, F, and G are system parameter matrices, $v_k \sim \mathcal{N}(0, , V_k)$ and $w_k \sim \mathcal{N}(0, , W_k)$ are CPS state and system measurement noise. Equations (2.1) and (2.2) represent the healthy CPS model.

2.2.2 The Intrusion Signal Model

Consider the intrusion signal to have the following state-space form:

$$h_{k+1} = \Phi h_k \tag{2.3}$$

$$z_k = \Gamma h_k \tag{2.4}$$

where $h_k \in \mathbb{R}^{p\times 1}$ is the intrusion state vector, $z_k \in \mathbb{R}^1$ represents the intrusion signal, $\Phi \in \mathbb{R}^p$ and $\Gamma \in \mathbb{R}^{1\times p}$ are system matrices for attack signals. By using this form for the unknown signal, any analytic signal (function) can be generated exactly or approximately as a power series by choosing Φ as

$$\Phi = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 0 & 1 & \cdots & 1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 \end{bmatrix} \tag{2.5}$$

where $\Phi \in \mathbb{R}^{n\times n}$, and the dimension of Φ and the initial value of h_0 determines the degree of the polynomial function that the intrusion model generates. For example, setting up $\Phi = 1$, with an arbitrary initial value h_0, the proposed model:

$$h_{k+1} = h_k \tag{2.6}$$

$$z_k = h_k \tag{2.7}$$

which generates a step or bias intrusion signal.

Similarly, following (2.5), a ramp or linear drift signal can be generated by setting up the dimension of Φ as 2×2:

$$h_{k+1} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} h_k \tag{2.8}$$

$$z_k = \begin{bmatrix} 1 & 0 \end{bmatrix} h_k \tag{2.9}$$

and in this case, using any arbitrary initial value h_0, a ramp or linear drift intrusion signal z_k can be generated.

2.2.3 State-Space Form of CPS When the Sensors are Under Attack

The CPS model, when sensor signals are under attack, can be modeled as follows:

$$\underbrace{\begin{bmatrix} x_{k+1} \\ h_{k+1} \end{bmatrix}}_{\mathscr{X}_{k+1}} = \underbrace{\left[\begin{array}{c:c} A & 0 \\ \hdashline 0 & \Phi \end{array}\right]}_{\mathscr{A}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + \underbrace{\left[\begin{array}{c} B \\ \hdashline 0 \end{array}\right]}_{\mathscr{B}} u_k + \underbrace{\left[\begin{array}{c} F \\ \hdashline 0 \end{array}\right]}_{\mathscr{F}} v_k \tag{2.10}$$

$$y_k = \underbrace{\left[\begin{array}{c:c} \Lambda_1 C & \Lambda_2 \Gamma \end{array}\right]}_{\mathscr{C}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + Du_k + Gw_k \tag{2.11}$$

This augmented model combines (2.1) and (2.2) with (2.3) and (2.4) and introduces two scalar weighting factors $\Lambda_1 \in [0, 1]$ and $\Lambda_2 \in [0, 1]$. In (2.10), Φ is employed as the system matrix from the intrusion signal model in (2.3) to generate the state vector h_k.

In (2.11), Γ serves as the measurement from (2.4), while Λ_1 and Λ_2 represent

weighting factors for the sensor signals in the healthy system and the system with sensor intrusions, as implemented by the attackers.

For example, when $\Lambda_1 = 0$ and $\Lambda_2 = 1$, (2.11) simplifies to:

$$y_k = \underbrace{\left[\begin{array}{c:c} 0 & \Gamma \end{array}\right]}_{\mathscr{C}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + Du_k + Gw_k \tag{2.12}$$

In this case, the healthy sensor signal is entirely replaced by the intrusion signal z_k. If, for example, $\Lambda_1 = 0.5$ and $\Lambda_2 = 0.5$, (2.11) becomes:

$$y_k = \underbrace{\left[\begin{array}{c:c} 0.5C & 0.5\Gamma \end{array}\right]}_{\mathscr{C}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + Du_k + Gw_k \tag{2.13}$$

In this scenario, half of the healthy sensor signal is substituted with the intrusion signal. Importantly, the sum of weighting factors does not necessarily equal one. This is because hackers have the capability to directly inject both intrusion and healthy signals to manipulate the CPS. In such instances, Λ_1 and Λ_2 assume values of one.

In real-world situations, hackers may create arbitrary intrusion signals and subsequently substitute them for healthy sensor or actuator signals to impact the CPS. Consequently, parameters such as Φ, Γ, Λ_1, and Λ_2 remain unknown when constructing the affected CPS model. However, it is noteworthy that none of these parameters require prior knowledge for intrusion detection, as will be demonstrated in the following section.

2.2.4 State-Space Form of CPS When Actuator is Under Attack

If the actuator signal undergoes partial or complete replacement by an intrusion signal, the state-space representation of the CPS under actuator attack can be formulated as follows:

$$\underbrace{\begin{bmatrix} x_{k+1} \\ h_{k+1} \end{bmatrix}}_{\mathscr{X}_{k+1}} = \underbrace{\left[\begin{array}{c|c} A & \Lambda_2 B\Gamma \\ \hline 0 & \Phi \end{array}\right]}_{\mathscr{A}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + \underbrace{\begin{bmatrix} \Lambda_1 B \\ \hline 0 \end{bmatrix}}_{\mathscr{B}} u_k + \underbrace{\begin{bmatrix} F \\ \hline 0 \end{bmatrix}}_{\mathscr{F}} v_k \tag{2.14}$$

$$y_k = \underbrace{\left[\begin{array}{c|c} C & \Lambda_2 D\Gamma \end{array}\right]}_{\mathscr{C}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + \underbrace{\Lambda_1 D}_{\mathscr{D}} u_k + Gw_k \tag{2.15}$$

Similar to the state-space form when the measurement signals are under attack, the matrices Φ and Γ model the intrusion signal with dimensions mentioned before, where $\Lambda_1 \in [0, 1]$ and $\Lambda_2 \in [0, 1]$ act as weighting factors as previously described. For instance, if $\Lambda_1 = 0$ and $\Lambda_2 = 1$, (2.14) and (2.15) simplify to:

$$\underbrace{\begin{bmatrix} x_{k+1} \\ h_{k+1} \end{bmatrix}}_{\mathscr{X}_{k+1}} = \underbrace{\left[\begin{array}{c|c} A & B\Gamma \\ \hline 0 & \Phi \end{array}\right]}_{\mathscr{A}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + \underbrace{\begin{bmatrix} F \\ \hline 0 \end{bmatrix}}_{\mathscr{F}} v_k \tag{2.16}$$

$$\begin{aligned} y_k &= \underbrace{\left[\begin{array}{c|c} C & D\Gamma \end{array}\right]}_{\mathscr{C}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + Gw_k \\ &= Cx_k + Dz_k + Gw_k \end{aligned} \tag{2.17}$$

In this scenario, the original actuator signal u_k is entirely substituted by the intrusion signal $z_k = \Gamma h_k$.

2.2.5 State-Space Form of CPS When Sensor and Actuator are Both Under Attack

In the event that both sensor and actuator signals are compromised, a unified model is employed. This combined attack model is described as:

$$\underbrace{\begin{bmatrix} x_{k+1} \\ h_{k+1} \end{bmatrix}}_{\mathscr{X}_{k+1}} = \underbrace{\left[\begin{array}{c|c} A & \Lambda_2 B\Gamma_2 \\ \hline 0 & \Phi \end{array}\right]}_{\mathscr{A}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + \underbrace{\begin{bmatrix} (1-\Lambda_2)B \\ \hline 0 \end{bmatrix}}_{\mathscr{B}} u_k + \underbrace{\begin{bmatrix} F \\ \hline 0 \end{bmatrix}}_{\mathscr{F}} v_k \tag{2.18}$$

$$y_k = \underbrace{\left[\begin{array}{c:c} (1-\Lambda_1)C & (1-\Lambda_1)\Gamma_1 + \Lambda_2 D\Gamma_2) \end{array}\right]}_{\mathscr{C}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + \underbrace{(1-\Lambda_2)D}_{\mathscr{D}} u_k + Gw_k \tag{2.19}$$

In (2.18) and (2.19), $\Lambda_1 \in [0, 1]$ serves as the weighting factor signifying the healthy system with sensor intrusions, while $\Lambda_2 \in [0, 1]$ acts as the weighting factor indicating the healthy system with actuator intrusions. Given that various signals can be employed to interfere with sensors or actuators, Γ_1 and Γ_2 are employed to represent the nature of intrusion signals affecting sensor and actuator signals, respectively.

For example, if half of the healthy sensor signal is supplanted by a step-type signal, and all of the healthy actuator signal is replaced by a ramp-type signal, then the state-space form of the intrusion signal can be represented (based on (2.3) and

(2.4)) as

$$h_{k+1} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} h_k \tag{2.20}$$

$$z_k^1 = \Gamma_1 h_k = \begin{bmatrix} 0 & 1 \end{bmatrix} h_k \tag{2.21}$$

$$z_k^2 = \Gamma_2 h_k = \begin{bmatrix} 1 & 0 \end{bmatrix} h_k \tag{2.22}$$

where $\Lambda_1 = 0.5$ and $\Lambda_2 = 1$ represent the percentage of the corresponding healthy signal being replaced, respectively. z_k^1 is the step-type signal that affects the healthy sensor signal, and z_k^2 represents the ramp-type signal affecting the healthy actuator signal.

In this case (2.18) and (2.19) are simplified to:

$$\underbrace{\begin{bmatrix} x_{k+1} \\ h_{k+1} \end{bmatrix}}_{\mathscr{X}_{k+1}} = \underbrace{\left[\begin{array}{c|c} A & B\Gamma_2 \\ \hline 0 & \Phi \end{array}\right]}_{\mathscr{A}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + \underbrace{\left[\begin{array}{c} F \\ \hline 0 \end{array}\right]}_{\mathscr{F}} v_k \tag{2.23}$$

$$y_k = \underbrace{\left[\begin{array}{c|c} 0.5C & 0.5\Gamma_1 + D\Gamma_2 \end{array}\right]}_{\mathscr{C}} \underbrace{\begin{bmatrix} x_k \\ h_k \end{bmatrix}}_{\mathscr{X}_k} + Gw_k \tag{2.24}$$

Recall (2.19), it can be noticed since all healthy actuator signal is corrupted and replaced by the intrusion signal ($\Lambda_2 = 1$), the original actuator signal (Du_k) does not exist in the measurement equation anymore, and it is being replaced by Dz_k^2 as shown in (2.24). Also, since half of the healthy sensor signal is replaced by the step-

type intrusion signal, the original healthy sensor signal becomes $0.5Cx_k + 0.5\Gamma_1 z_k^1$, which is a combination of half of the healthy and half of the intrusion signal.

By formulating the healthy CPS model, the CPS model under sensor or actuator intrusion, and the CPS model under both types of intrusions with the state-space representation, an algorithm for detecting sensor, actuator, or both intrusions is presented in the following section.

2.3 Proposed Detection Technique

2.3.1 Sensor or Actuator Intrusion Detection via Adapted MMAE Technique

In this application, the MMAE technique is adapted to detect CPS sensor and actuator intrusions with the CPS models shown previously. Different from previous works [82, 83, 84, 85], where the sensor and actuator intrusion problem is transformed to be an estimation problem with system uncertainties, in this work, the unknown intrusion signal is detected by computing the likelihood of the estimated CPS states using the data fusion technique.

Here, we first consider if there is only unknown type of sensor or actuator attacks, then we will consider the situation where we don't know if there is a sensor, actuator or both sensor and actuator are under attack.

If we known if sensor or actuator is under attack, in this case, only two state estimators are required to formulate the MMAE technique to detect different types (sensor/actuator) of unknown intrusion signals. These estimators serve distinct purposes within the system.

The first estimator is tailored to a healthy CPS scenario, and it is constructed

based on the well-functioning CPS model shown in (2.1) and (2.2). The second estimator is specifically engineered to address situations where the intrusion signal impacts either the sensor or the actuator. For sensor intrusion scenarios, models described in (2.10) and (2.11) is applied for the detection, while for actuator intrusion cases, it leverages models presented in (2.14) and (2.15). Creating these estimators is straightforward, as they can be readily established according to the provided models, whether for a healthy or affected space-state configuration.

Both estimators play a pivotal role in estimating the conditional states of the CPS. They achieve this by sharing the CPS's output. Subsequently, they are used to adaptively update the conditional probabilities associated with each possible intrusion scenario. This iterative process relies on the conditional state estimates derived from both estimators, enhancing the system's ability to identify intrusions effectively.

Ultimately, the estimator that yields the highest probability is deemed to represent the current sensor signal, offering a reliable means of intrusion detection. Figure 2.1 provides a visual overview of this detection process.

When a hacker corrupted the healthy CPS, replacing the healthy CPS output signal by a (either sensor or actuator) intrusion signal, two estimators running in parallel have been meticulously designed, one based on the model representing a healthy CPS, and the other tailored to the CPS model under intrusion conditions.

Following this initial stage, both estimators diligently calculate conditional state estimates for the CPS, considering both the healthy and affected models. These estimates serve as critical indicators in assessing the CPS's state.

Subsequently, a Bayes data fusion technique (as introduced in (3.7)) comes into

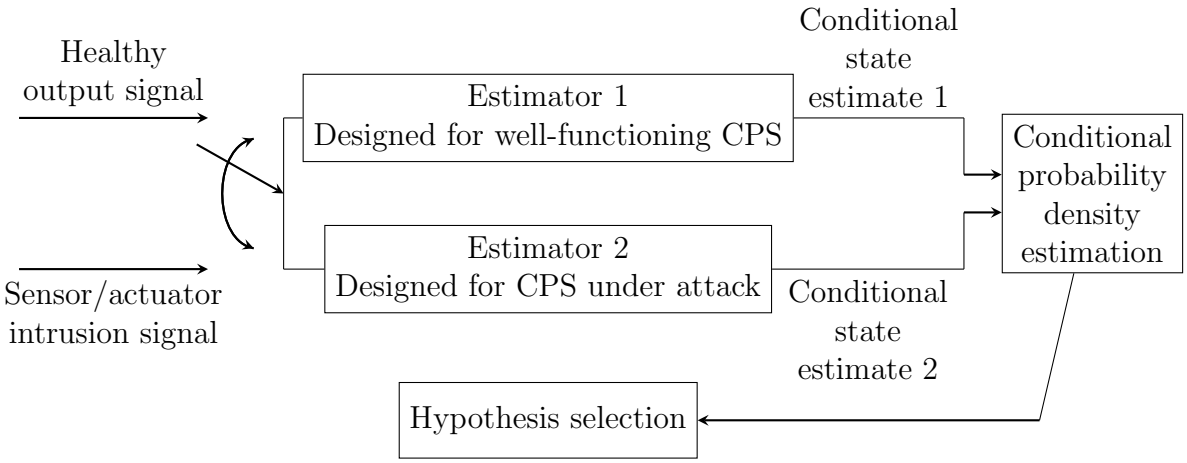

Figure 2.1: Block diagram of MMAE technique adapted for sensor or actuator intrusion detection.

play, working to categorize whether the CPS is in a healthy state or currently under attack.

In summary, by calculating the conditional probabilities of each conditional state estimates under various scenarios, this adapted MMAE approach facilitates the determination of the CPS's status. This process ensures that any deviations from the expected norm, caused by intrusion signals, are promptly detected, enabling swift and effective countermeasures to safeguard the system's integrity and functionality. Simulation results on CPS sensor and actuator intrusion detection is introduced in Section 2.4.

2.3.2 Combined Intrusion Detection via Adapted MMAE Technique

In practice, discerning whether a CPS has been affected by an intrusion signal, and if so, which specific component (sensor, actuator, or both) is compromised, can be a formidable challenge. To tackle this issue comprehensively, a set of four

estimators can be deployed, each addressing distinct CPS states to determine its well-being or the presence of an intrusion signal stemming from a sensor attack, an actuator attack, or a combined attack.

As in the previous approach, the first estimator is designed to suit a healthy CPS scenario, constructed based on the CPS model indicative of proper functionality, encapsulated in (2.1) and (2.2). The second estimator is custom-tailored to situations where the intrusion signal interferes with the CPS's sensors, relying on (2.10) and (2.11). The third estimator is specifically devised for cases where hackers target the actuator of the CPS, drawing from (2.14) and (2.15). Finally, the fourth estimator is designed for scenarios in which both the sensor and actuator are impacted by the intrusion signal, with its foundation in (2.18) and (2.19).

Following the adapted MMAE methodology detailed earlier, these four estimators within a bank are provided with identical sensor and actuator signals. The corresponding state estimators are designed to utilize their respective CPS models to compute conditional state estimates, and then the data fusion technique (as introduced in (3.7)) is applied to calculate conditional probabilities. These probabilities serve as a key determinant in ascertaining the CPS's status, whether it is functioning healthily or subject to a specific form of attack.

Figure 2.2 provides a visual representation of this integrated estimation process employing the adapted MMAE technique. Similar to the previous solution, this combined detection framework features four estimators operating within a bank. Estimator 1 is grounded in the Healthy CPS model as per (2.1) and (2.2). Estimator 2 is crafted for situations involving sensor attacks, utilizing (2.10) and (2.11). Es-

timator 3 is devised for scenarios where the actuator signal faces compromise from malicious actors, drawing from (2.14) and (2.15). Lastly, estimator 4 is engineered for CPS instances simultaneously subjected to sensor and actuator attacks, with its basis in (2.18) and (2.19).

When intrusion signals substitute the healthy sensor or actuator signals, these estimators generate conditional state estimates under various scenarios. The data fusion technique is subsequently applied to compute conditional probabilities, enabling the identification of which CPS components are affected by the intrusion signal.

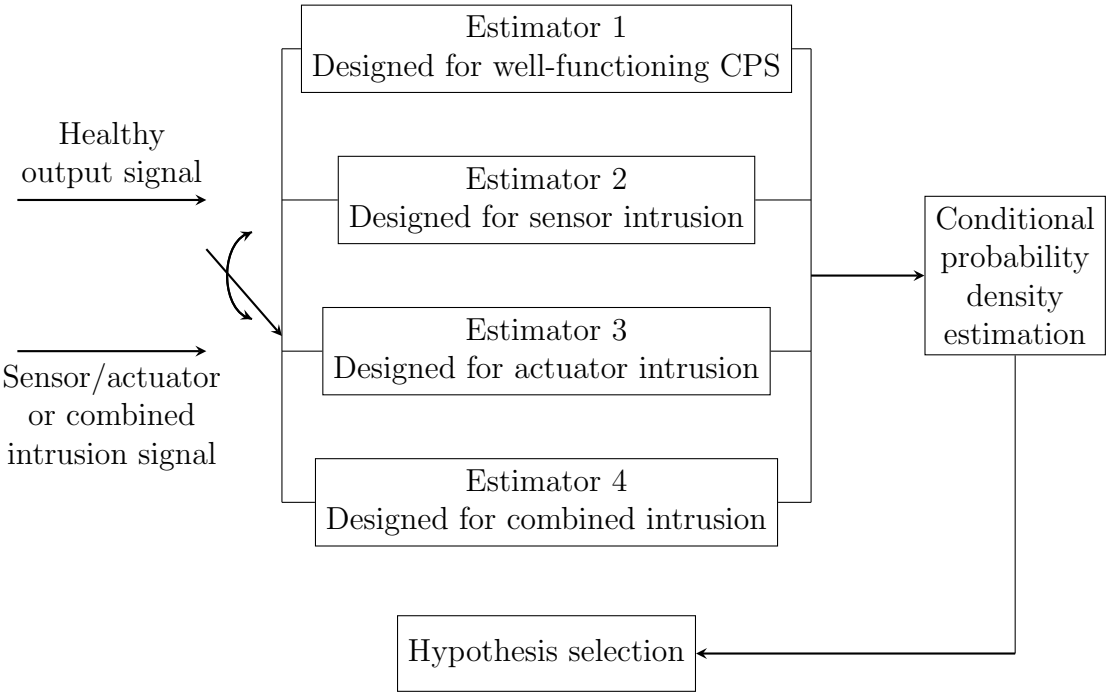

Figure 2.2: Block diagram of MMAE technique adapted for combined intrusion detection.

2.4 Simulation Results

In this section, the method proposed for detecting and identifying intrusion signals in the CPS when sensor, actuator, or both are under attack is simulated with a DC motor model. Unknown sensor and actuator intrusion signals are generated and utilized to replace the well-functioning signals. Then, the detection technique is employed to first identify the intrusion and subsequently determine the type of intrusion signal.

2.4.1 The Healthy CPS Model

The CPS dynamic equations of the proposed DC motor in state-space form are shown as [8]

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -\frac{b}{J} & \frac{K}{J} \\ -\frac{K}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} u \tag{2.25}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{2.26}$$

For this CPS, the voltage source (u) applied to the motor's armature constitutes the input, while the rotational velocity of the shaft (x_1) represents the output, and x_2 denotes the armature current. The physical parameters of this system are as follows:

- J: moment of inertia of the rotor, $0.01 kg.m^2$
- b: motor viscous friction constant, $0.1 N.m.s$

- R: winding resistance, 1Ω
- L: winding inductance, $500mH$
- K_e: electromotive force constant, $0.01V/rad/sec$
- K_t: motor torque constant $0.01N.m/Amp$
- K: In SI units, $K_t = K_e$; hence, K is employed to represent both the motor torque constant and the electromotive force constant.

In this model, it is assumed that the rotor and shaft are rigid, and the friction torque is considered to be proportional to the shaft's angular velocity [86].

After designing the full-state feedback controller and discretizing the system (using zero-order hold) with sampling time $T = 50ms$, the discretized system is

$$\begin{bmatrix} x_{k+1}^1 \\ x_{k+1}^2 \end{bmatrix} = \begin{bmatrix} 0.6565 & 0.03729 \\ -0.0007458 & 0.9048 \end{bmatrix} \begin{bmatrix} x_k^1 \\ x_k^2 \end{bmatrix} + \begin{bmatrix} 0.002059 \\ 0.09516 \end{bmatrix} u_k \tag{2.27}$$

$$y_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_k^1 \\ x_k^2 \end{bmatrix} \tag{2.28}$$

In this case, the generalized healthy DC motor system is given as

$$x_{k+1} = A_d x_k + B_d u_k + F_d v_k \tag{2.29}$$

$$y_k = C_d x_k + G_d w_k \tag{2.30}$$

where $F_d = I_{2\times 2}$, $G_d = 1$. The state and measurement noises, $v_k \sim N(0, \sigma_v)$,

$w_k \sim N(0, \sigma_w)$ and $\sigma_v^2 = 0.001$, $\sigma_w^2 = 0.01$. With the given generalized healthy model, estimator 1 as shown in Figure 2.1 can be designed accordingly.

2.4.2 Detection of the Unknown Sensor and Actuator Intrusion Signal

In practice, it is infeasible to ascertain the precise nature of attack signals that hackers might employ. Consequently, a considerable likelihood exists for a discrepancy to emerge between the assumed and actual CPS models when the system is compromised. For instance, if it is assumed that hackers will utilize a step-type signal to supplant the healthy sensor/actuator signal, and the CPS model is constructed based on this presumption, there may be a scenario in which the hackers employ a different signal type, such as a ramp-type signal, to disrupt the sensor/actuator components of the CPS. In such a case, the assumed CPS model no longer aligns with the actual affected CPS model, primarily because the assumed values of Φ, Γ, Λ_1, and Λ_2 do not correspond to the real-world circumstances.

Therefore, it becomes crucial to determine whether the intrusion signals can still be detected even in the presence of a mismatch between the actual compromised CPS model and the assumed model. To investigate this aspect, two illustrative cases are considered to assess the proposed algorithm's robustness in situations where a disparity exists between the assumed and actual attack models.

In the first case, the filter associated with the affected CPS model (either sensor or actuator) is tailored for a step-type intrusion, while the actual attack signal takes the form of a ramp. In the second case, the assumed attack signal type for which the filters are designed possesses a higher order than the actual attack signal employed

by the hacker.

2.4.2.1 Case 1: The Under Estimation Problem

Suppose that a ramp-type signal is employed by a hacker to manipulate the healthy rotational velocity (measurement) or voltage source (actuator) signal of the DC motor. Meanwhile, the assumed attack model is defined as a step-type signal, forming the basis of the affected CPS model. In this scenario, a discrepancy emerges between the assumed model and the actual attack model. To be more precise, disparities manifest in the actual versus assumed values of Φ, Γ, Λ_1, and Λ_2.

In cases where the actual attack signal exhibits a higher order than the assumed attack signal, the problem at hand is categorized as an "underestimation" issue. Figure 2.3 provides a visual representation of the outcomes associated with underestimation. This situation arises when the healthy rotational velocity (measurement) signal is supplanted by a ramp-type intrusion signal, while the assumed affected CPS model is established based on the assumption of a step-type intrusion signal.

Similarly, in the case of the actuator, the issue of underestimation also arises. Figure 2.4 is employed to simulate a scenario where the motor voltage source signal is substituted with a ramp-type intrusion signal, while the estimator is designed with the assumption that a step-type signal is affecting the motor.

For both cases, the measurement signals assume the ramp shape of the attack signal, as depicted in Figures 2.3 (a) and 2.4 (a). Notably, the conditional probabilities, which transition from the healthy system estimator to the attacked system estimator, promptly after the onset of the intrusion, are shown in Figures 2.3 (b)

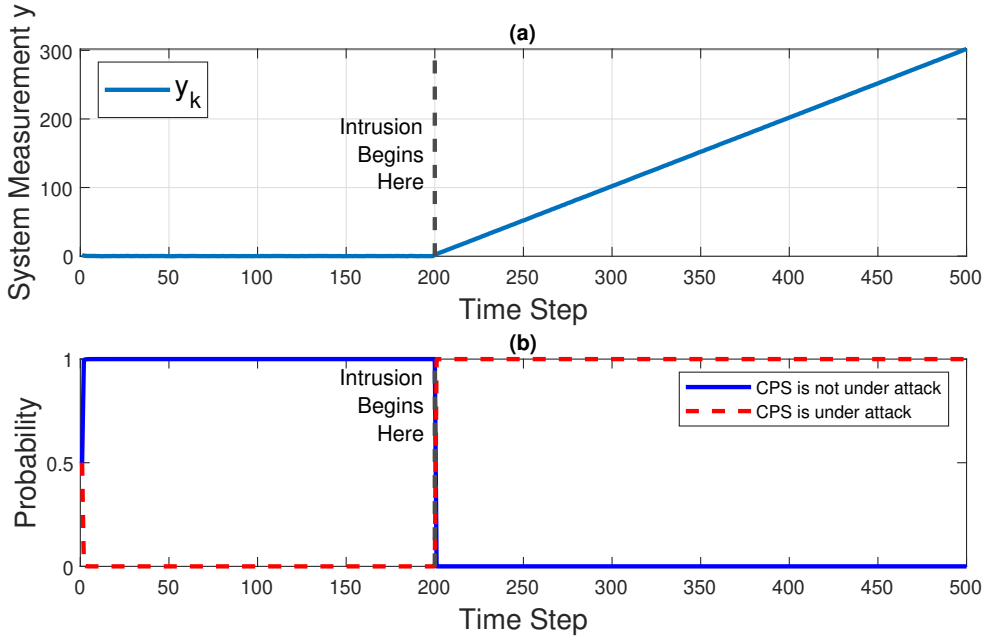

Figure 2.3: Under Estimation Problem: (a) Healthy rotational speed signal is replaced by the ramp-type attack signal at $k = 200$ while the assumed affected model is a step-type based model and (b) conditional probabilities given the healthy rotational speed signal and affected signal at each time step k.

and 2.4 (b).

These findings serve to underscore that the detection of sensor or actuator tampering remains viable even when the attack signal exhibits a higher order than the expected intrusion signal for which the filter was initially designed.

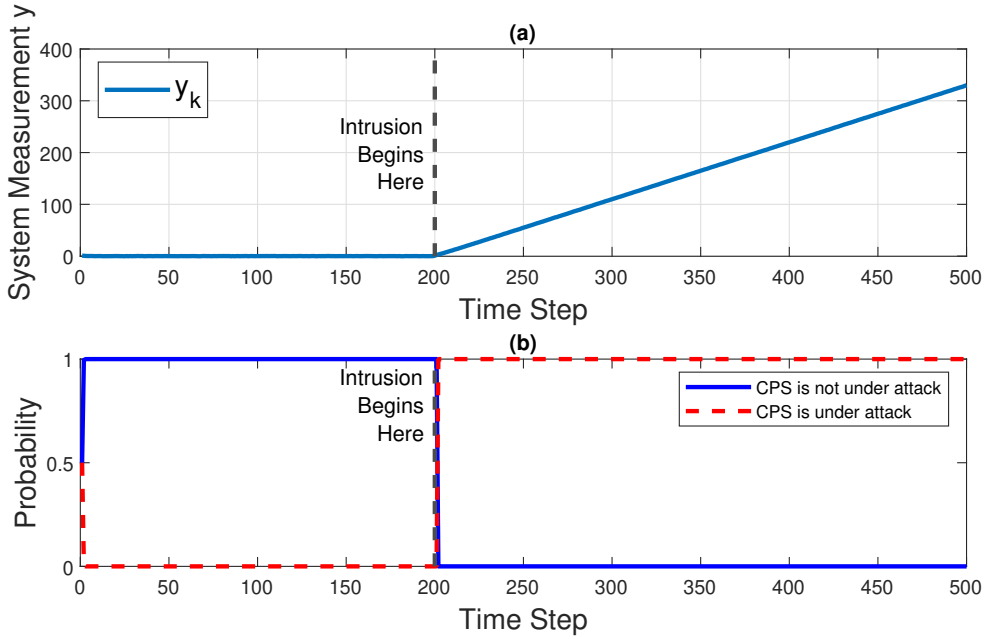

Figure 2.4: Under Estimation Problem: (a) Healthy voltage source (actuator) signal is replaced by the ramp-type attack signal at $k = 200$ while the assumed affected CPS model is a step-type based model and (b) conditional probabilities given the healthy voltage source signal and affected ramp shape signal at each time step k.

2.4.2.2 Case 2: The Over Estimation Problem

If the intrusion signal model used to develop the Kalman filter in the bank is of a higher order than the actual attack signal, detection of this type of intrusion signal is called the over estimation problem. Figure 2.5 shows the over estimation result when the healthy rotational speed signal is replaced by a step-type intrusion signal while the filter is designed assuming a ramp-type signal is affecting the CPS.

For the actuator case, Figure 2.6 shows the over estimation result when the healthy voltage source signal is replaced by a step-type intrusion signal, while the filter is designed based on the ramp-type intrusion signal. The control input signal

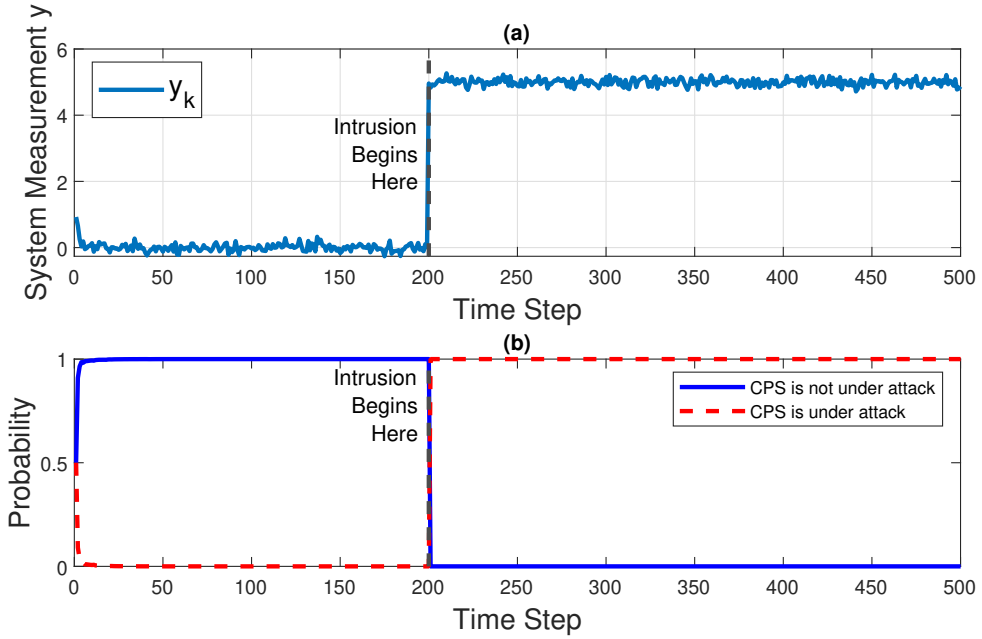

Figure 2.5: Over Estimation Problem: (a) Healthy rotational speed signal is replaced by the step-type attack signal at $k = 200$ while the assumed affected CPS model is a ramp-type based model and (b) conditional probabilities given the healthy rotational speed signal and affected step shape signal at each time step k.

and output signal for these two cases assume the step shape of the attack signal seen in Figure 2.5 (a) and Figure 2.6 (a). Similarly, the conditional probability which takes on the value of 1 switches from the healthy system filter to the attacked system filter after the system is attacked for each case shown in Figure 2.5 (b) and Figure 2.6 (b). These results show that both sensor and actuator hacking is detected when the attack signal is of lower order than that of the expected intrusion signal for which the filter is designed.

These simulation results show that if the intrusion signal is unknown, the affected CPS model can be assumed arbitrary, without caring about what type of intrusion

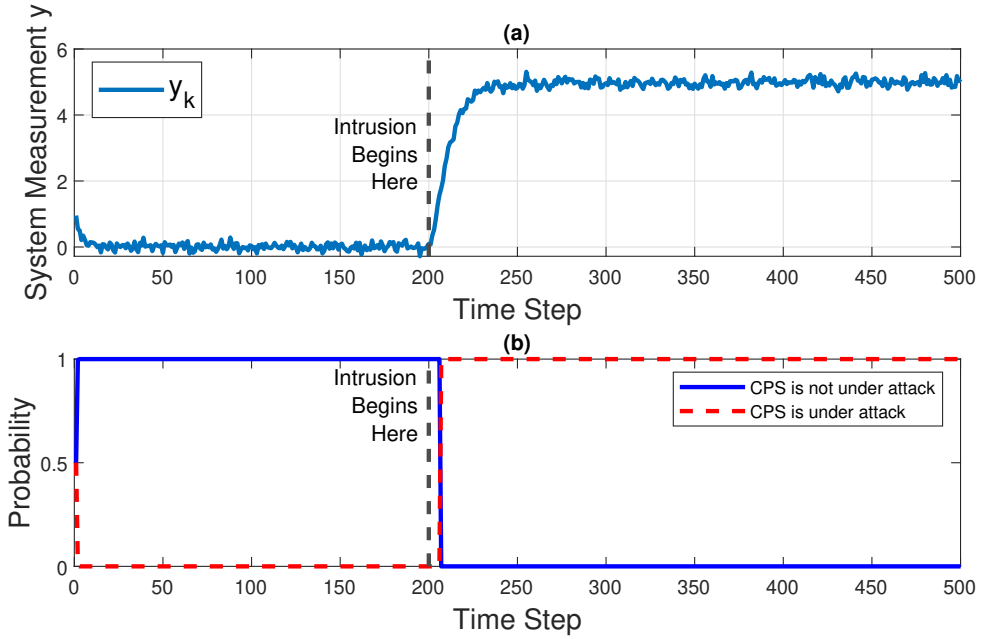

Figure 2.6: Over Estimation Problem: (a) Healthy voltage source (actuator) signal is replaced by the step-type attack signal at $k = 200$ while the assumed affected CPS model is a ramp-type based model and (b) conditional probabilities given the healthy voltage source signal and affected step shape signal at each time step k.

signals hackers will use. When additional simulations of under and over-estimation situations were carried out, the presense of an intrusion was always detected.

2.4.3 Detection of the Intrusion Signal When the Target of the Intrusion is Unknown

In practice, it is not known if the hacker will attack the sensor or the actuator of the CPS, that is to say, the component which hackers want to affect is usually unknown. In this case, as mentioned in Section 2.3.2, by designing four Kalman filters in a bank and with the adaptive estimation algorithm implemented, the intrusion signal can not only be detected but be specified. Suppose the hacker wants to use a step-type signal to to replace the healthy rotational speed signal, then the detection

result is shown in Figure 2.7

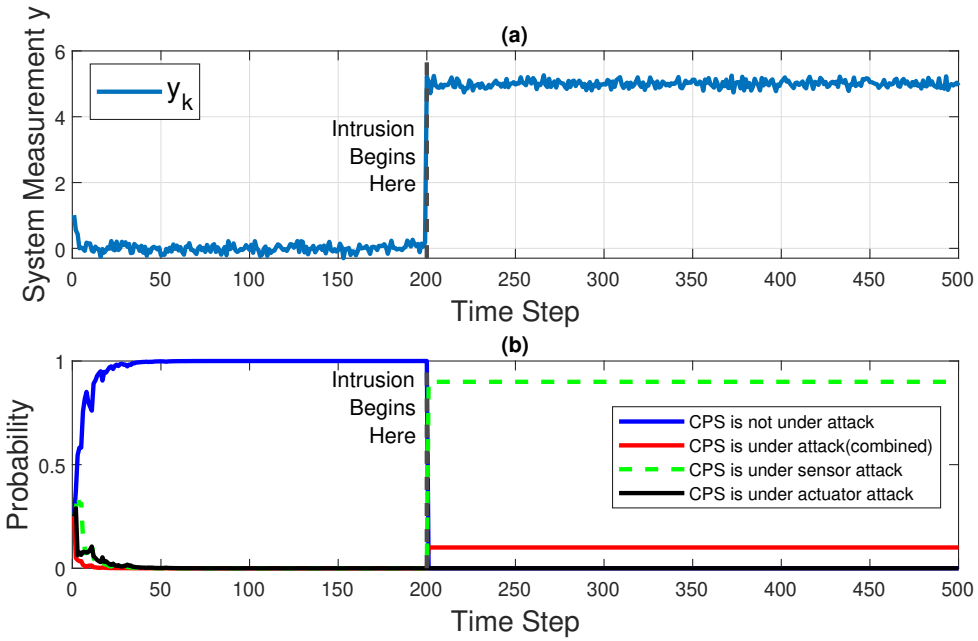

Figure 2.7: (a) Healthy rotational speed (sensor) signal is replaced by a step-type attack signal at $k = 200$ and (b) conditional probabilities given the healthy rotational speed signal, affected rotational speed signal, affected voltage source signal and affected combined signal at each time step k.

As seen in Figure 2.7 when a step-type intrusion signal corrupts the healthy rotational speed signal, detection results show that there is about 95% probability that the sensor is under attack, and there is about 5% of the probability that both sensor and actuator signal are affected. Since the combined CPS model assumes both sensor and actuator are affected, the detection results show that there is a

small probability that both components are under attack.

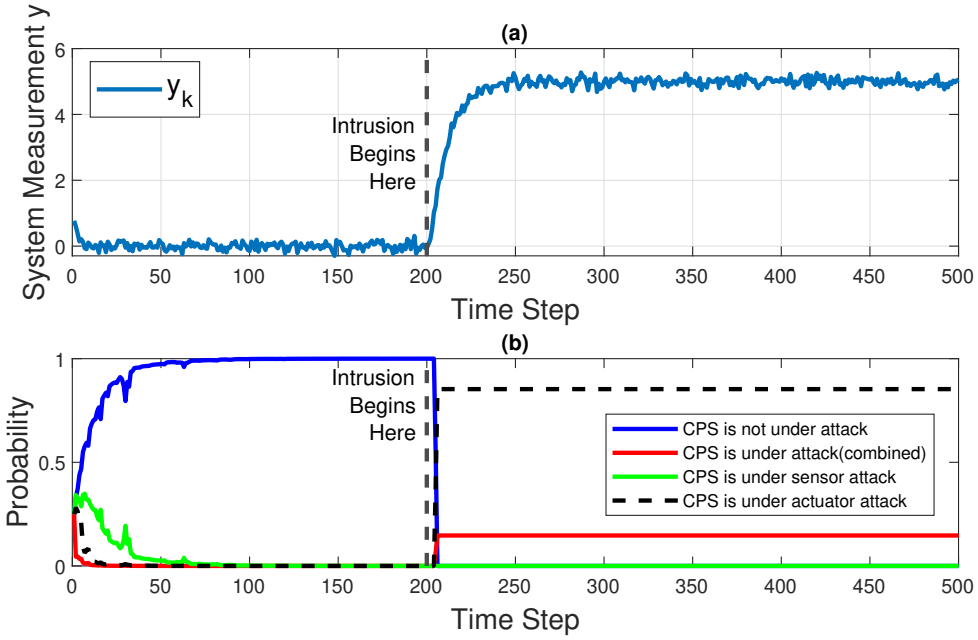

Figure 2.8: (a) Healthy voltage source (actuator) signal is replaced by the step-type attack signal at $k = 200$ and (b) conditional probabilities given the healthy voltage source signal, affected rotational speed signal, affected voltage source signal and affected combined signal at each time step k.

Simulation results are obtained as seen in Figure 2.8 show that if the healthy voltage source signal is replaced by a step-type intrusion signal, there is about 90% probability that the actuator is under attack, and about 10% probability that it is a combined intrusion signal. Similarly, since the combined affected CPS considers the dynamics when the actuator is affected, there is a small percentage of the probability

that shows the intrusion signal might affect both components.

2.4.4 Reducing the Detection Time Delay

When a CPS faces an attack, it necessitates that both estimators within the bank promptly adapt to the shift from the healthy measurement. This adaptation involves computing new optimal estimator gains for both filters. To achieve this, the estimation error covariance undergoes an automatic update through the utilization of the Riccati equation in both filters. This automatic update is the root cause of the detection time delay when transitioning from a healthy measurement to a corrupted one.

To mitigate this detection time delay, the key lies in expediting the discovery of the optimal estimator gain immediately after the sudden change in CPS measurement. This expedited search for the optimal estimator gain can be realized by enhancing the efficiency of the Riccati equation to minimize the error covariance at a quicker pace. One effective approach to achieve this is by introducing a fading memory term, denoted as α, into the Riccati equation [12]. The inclusion of this term accelerates the minimization of the error covariance [12]. Consequently, estimators converge more rapidly to the corrupted measurement.

For the faster convergence of both filters, the technique necessitates a modification of the Riccati equation, which can be expressed as follows:

$$P_{k+1} = \alpha^2 A_k P_k A_k^T - \alpha^2 A_k P_k C_k^T (C_k P_k C_k^T + G_k W_k G_k^T) + F_k V_k F_k^T \tag{2.31}$$

Here, $\alpha \in \mathbb{R}$ represents a tuning parameter which is slightly greater than 1.

Employing this modified Riccati equation for both estimators within the bank significantly reduces the detection time delay.

Figure 2.9 and 2.10 illustrate the effectiveness of this approach in detecting step-type sensor and actuator intrusion signals using the modified Riccati equation with $\alpha = 1.1$.

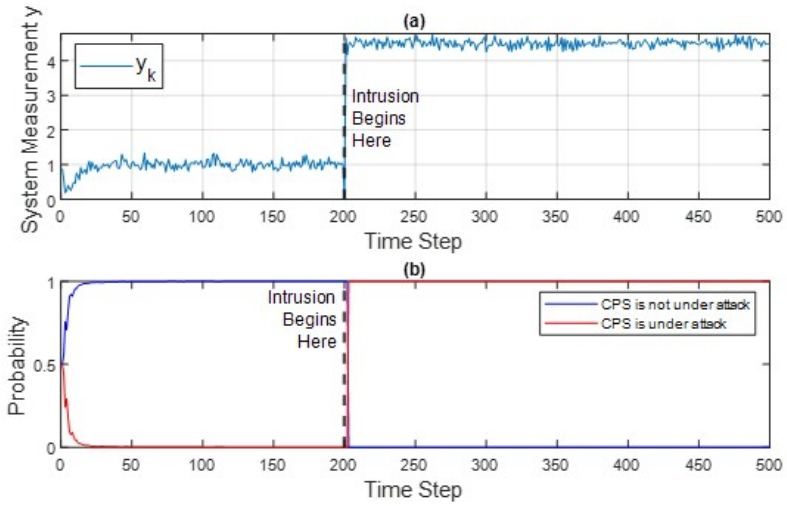

Figure 2.9: Detection result of the step-type sensor intrusion using modified Riccati equation

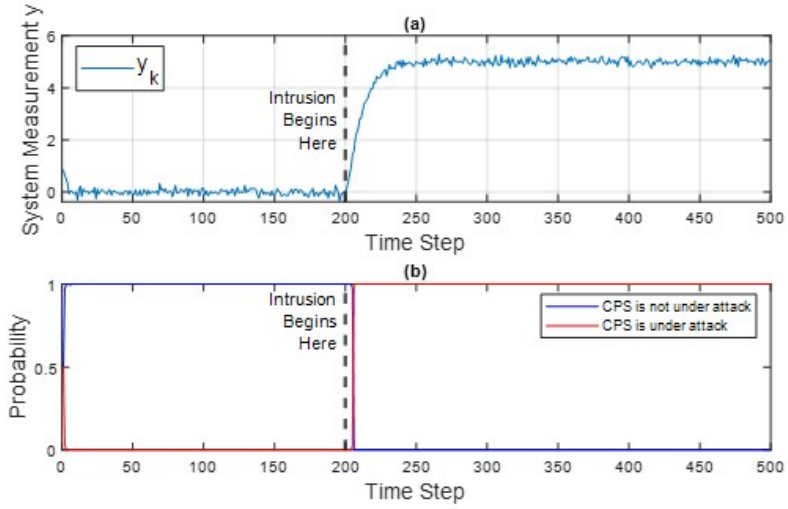

Figure 2.10: Detection result of the step-type actuator intrusion using modified Riccati equation

From Figure 2.9 (a) and Figure 2.10 (a), it is evident that the step-type intrusion signal replaces the original healthy sensor (rotational velocity of the shaft) and actuator (motor voltage source) signals at time step $k = 200$. Moving to Figure 2.9 (b) and Figure 2.10 (b), the MMAE with modified Riccati equation results in the detection of the sensor intrusion signal at time step $k = 201$, while the actuator intrusion signal is detected at time step $k = 204$. Importantly, in this scenario, the detection time delay for sensor intrusion is merely 0.05 seconds, and for actuator intrusion, it is only 0.2 seconds.

2.5 Conclusion

In this chapter, the CPS intrusion detection problem is addressed when subjected to attacks on the sensor or the actuator. Within this framework, the capability to analyze the impact of various intrusion signals on system performance is provided, along with the development of strategies for the detection of these intrusion signals. To tackle the intrusion detection problem, the initial consideration involves the generalized healthy CPS model and the model when the CPS is under different types of attacks. Subsequently, the adaptation of the MMAE technique for detecting the intrusion signals and their effects on the system is executed.

Specifically, the estimation of conditional state estimates in different scenarios is conducted, followed by the Bayesian data fusion technique to determine the most likely intrusion signal. Simulation results, using a DC motor CPS model, demonstrate the effectiveness of the proposed technique in not only detecting unknown intrusion signals but also identifying whether the intrusion signal is related to the sensor intrusion or the actuator intrusion. Additionally, in addressing the detection time delay arising from the estimator's convergence, the exponential data weighting technique is performed to reduce the detection time delay.

As CPS networks grow in complexity and size, future work will focus on the scalability and efficiency of intrusion detection methods to protect large-scale systems effectively. Specifically, intrusions detection on systems with multiple sensors/actuators will be investigated. In addition, developing strategies for automated or semi-automated responses to detected intrusions could also enhance CPS security.

Chapter 3: Improved Battery Management System of Lithium-ion Cells Using Multiple Model Adaptive Estimation Approach

Lithium-ion battery cells are indispensable in numerous applications, necessitating precise online State of Charge (SOC) and State of Health (SOH) estimation for improved safety, performance, and life cycle management. However, the nonlinearity inherent in battery cell models presents significant challenges in achieving accurate SOC and SOH estimates, and this will result in inaccurate SOC and SOH estimation with traditional estimation techniques. To address the problems mentioned above, this chapter introduces novel approaches to estimate SOC and SOH of Lithium-ion cells. Specifically, the proposed technique leverages the Multiple Model Adaptive Estimation (MMAE) method first, employing a bank of Kalman filters to adaptively estimate SOC. After that, an improved approach which combines MMAE with the Extended Kalman filter (EKF) to enhance the accuracy of SOC estimation and mitigate the limitations of each individual method is introduced. In addition, the MMAE technique is also adapted to simultaneously estimate SOC and cell total capacity integrated with an Enhanced Self-Correcting (ESC) model. Within the adapted MMAE technique, conditional probabilities are calculated for all potential quantized capacity values, with the most likely cell capacity estimate being deter-

mined. Subsequently, the SOC estimate is derived based on these given conditional probabilities at each iteration.

3.1 Introduction

Lithium-ion batteries (LIBs) find extensive application in electrified vehicles, portable electronic devices, energy storage systems, and more [2]. The surge in LIB usage can be attributed, in part, to the decline in manufacturing costs and the array of advantages associated with this technology [11]. These advantages include high energy and power density, lightweight and flexible design, minimal self-discharge rates, and an extended cycle life [87]. Nevertheless, the widespread use of LIBs across diverse applications often encounters limitations stemming from inherent disadvantages, including susceptibility to overcharging, over-discharging, and overheating, which, in turn, may lead to performance degradation and a shortened life cycle [88]. Ongoing research endeavors are dedicated to mitigating these drawbacks and ensuring the safe, reliable, and efficient operation of LIBs, with a predominant focus on enhancing battery management system (BMS) technologies.

An essential function of the BMS is the monitoring of the SOC within LIBs. SOC denotes the remaining charge within a battery cell and stands as a critical parameter in averting overcharge and over-discharge, thus enhancing the overall longevity and performance of LIBs [89]. Moreover, various other pivotal BMS tasks, including SOH estimation, power estimation, and cell balancing, hinge upon precise SOC measurements [1]. Consequently, obtaining an accurate SOC measurement not only safeguards LIBs from damage but also elevates the performance of the entire BMS.

Presently, the measurement of SOC cannot be achieved directly through sensors, necessitating the application of state estimation techniques for the accurate estimation of SOC based on measured variables, such as current and voltage. Furthermore, the temporal evolution of SOC is contingent upon intricate nonlinear dynamic processes rooted in thermodynamics, electrode kinetics, and transport phenomena [90]. Consequently, SOC monitoring demands the application of nonlinear estimation techniques, including the Extended Kalman filter (EKF) and the Sigma Point Kalman filter (SPKF) [90, 91].

In [90, 91], an EKF was employed to estimate SOC in LiPB-based battery cells, frequently utilized in Hybrid Electric Vehicles (HEVs). The battery cell was modeled as a nonlinear system, with SOC being one of the system states. Employing load current as input and terminal voltage as output signals, the EKF facilitated online SOC estimation. Nonetheless, the EKF technique exhibits certain limitations, including heavy reliance on the accurate initialization of SOC for optimal convergence and an absence of constraint handling capabilities, which can lead to invalid SOC estimates (potentially exceeding 1 or falling below 0). These limitations result in inaccurate SOC estimates, potentially causing disruptions in BMS control operations.

In a distinct approach [92, 93], an SPKF was adopted for online SOC estimation in LiPB-based cells. In contrast to the EKF, the SPKF eliminates the need for system linearization prior to application. Nonlinearities in the system are propagated through selected Sigma Points, enabling the recursive computation of state estimates [17]. It is shown that a well-tuned SPKF-based SOC estimate outperforms the EKF in terms of accuracy, particularly when the initial SOC estimate is suboptimal.

Generally, a finely tuned SPKF can yield highly accurate and robust SOC estimates, with computational costs roughly on par with those of the EKF. However, the precise tuning of weighting constants in SPKF can be challenging, and suboptimal tuning can yield poorer estimation results than the EKF. Importantly, the effectiveness of the SPKF is significantly reliant on individual experience, as there is no systematic guidance available for tuning these weighting constants.

Other critical tasks performed by the BMS include SOH estimation, power estimation, and cell balancing. In terms of SOH estimation, cell total capacity is another important value that needs to be estimated accurately [3]. Cell capacity is defined as the product of the current drawn from the battery while it remains connected to the load until its cell voltage descends below a predefined threshold for each cell [94]. This metric specifies the quantity of charge, measured in ampere-hours (Ah), that the battery is rated to hold [1] and is typically established by the manufacturer as one of the nominal values. Nevertheless, cell capacity naturally diminishes during usage due to parasitic reactions, such as reactions between lithium metal and battery electrolytes, resulting in a decline in cell performance [95]. Additionally, corrosion, chemical loss through evaporation, crystal formation, and dendritic growth on electrodes contribute to capacity loss [1]. Therefore, the precise measurement of cell capacity is instrumental in enhancing cell performance.

Prior research pertaining to the concurrent estimation of SOC and cell capacity encompasses a variety of estimation techniques. In [87, 89, 96], SOC and cell capacity were simultaneously estimated through the application of dual estimation technique using measured input (load current) and output (terminal voltage) data. This tech-

nique involved the design of two EKFs functioning in parallel to enable the proper estimation of SOC and cell capacity. Specifically, these two estimators shared their state estimates until SOC and capacity estimates converged to their actual values. Nevertheless, the slow convergence of this estimation technique was attributed to the nonlinear cell dynamics arising from cell impedance, electrode kinetics, and hysteresis voltage. Furthermore, inadequate knowledge of the initial SOC and capacity values led to convergence failure.

In [1, 97, 98, 99], SOC and cell capacity were jointly estimated through a joint estimation technique using a single EKF and a simplified cell model. In this approach, the cell model underwent modification to incorporate cell capacity as an additional state variable with associated additive noise. Subsequently, an EKF was applied to estimate all state variables, including cell capacity and SOC. However, this method occasionally encountered limitations due to the numeric conditioning of mathematical operations as a result of the differing time scales between the evolution of states and parameters [1]. For example, SOC can fluctuate from 100% to 0% within a few hours, whereas cell capacity degradation typically unfolds over years. Consequently, the amount of change in cell capacity per iteration is substantially smaller compared to changes in SOC. In such instances, round-off errors gave rise to numerical inaccuracies when using the joint estimation approach, significantly decelerating the convergence of cell capacity compared to SOC, especially in cases where the initial capacity estimate was imprecise.

With the proliferation of machine learning techniques, data-driven approaches, including support vector machines (SVM) and neural network approaches [100, 101,

102], have also been employed for SOC and cell capacity estimation. However, these data-driven methods come with inherent drawbacks, such as substantial computational costs, the need for extensive data, and concerns about the interpretability of results affecting performance.

In this chapter, a novel approach is introduced, utilizing the MMAE technique based on a Bank of Kalman filters (BKF) to estimate the SOC of a LiFePO4 cell. The primary objective is to address the limitations of the EKF and SPKF methods, as discussed earlier. In this new approach, SOC is considered as an unknown parameter, assuming a value within the range of 0 to 1. It is subsequently quantized, where each quantization level represents a possible SOC value at a given time. The BKF is then employed in conjunction with a Bayesian approach to select the most likely SOC value at each iteration. By treating SOC as an unknown parameter, the intricate nonlinear cell model can be transformed into a linear model, significantly simplifying the SOC estimation process.

Following this, an enhanced approach is introduced to further reduce the computation cost by combining the EKF and the MMAE techniques for SOC estimation within a LiFePO4 cell. The primary aim is to reduce the computational overhead associated with the MMAE approach while further enhancing SOC estimation accuracy. In this method, the SOC estimation begins with MMAE to attain the precise initial SOC estimate, thus addressing the sensitivity of the EKF to initial estimation errors. Subsequently, the EKF is employed to refine the SOC estimation using the initial estimate derived from the MMAE technique. This approach not only reduces computational costs through EKF usage but also mitigates estimation errors

stemming from SOC quantization in the MMAE method.

This chapter also explores the concurrent estimation of SOC and cell capacity through the MMAE technique. In contrast to prior scenarios, cell capacity is quantized into a finite set of discrete values. Leveraging the MMAE technique, conditional SOC estimates are derived, along with conditional probabilities for each capacity candidate. These combined estimates of SOC and cell capacity are then computed using data fusion techniques. In comparison to the dual and joint estimation techniques discussed earlier, this approach yields robust and accurate SOC and cell capacity estimates, particularly when the initial estimates for SOC and total capacity are imprecise.

In summary, improvements of using a BKF over the EKF and the SPKF to estimate SOC and cell capacity are summarized as follows:

- In comparison to the EKF, the BKF approach for SOC estimation demonstrates accelerated convergence and the generation of estimations characterized by higher accuracy (lower estimation error) under equivalent noise levels when substantial initialization errors are present.

- Robustness characterizes SOC estimation with the BKF, irrespective of the quality of the initial SOC estimate. Conversely, when the initialization is suboptimal, the EKF occasionally yields estimated SOC values exceeding 1 or falling below 0.

- When compared to the SPKF approach, the BKF exhibits straightforward setup, especially when employing a simplified linear cell model. This obviates

the necessity for trial-and-error methods or parameter tuning to enhance the accuracy of estimation results.

- In the context of the combined SOC estimation approach, it offers the advantage of mitigating quantization errors while simultaneously reducing computational costs.

- Concerning the simultaneous estimation of SOC and cell capacity, the MMAE technique yields robust and accurate SOC and cell capacity estimates, particularly when the initial estimations for SOC and total capacity lack precision, as compared to the dual and joint estimation techniques.

This chapter consists of five sections: section 2 includes the introduction of the BKF for online SOC estimation, Section 3 introduces the improved SOC estimation technique which combines the BKF and the EKF, Section 4 consists of simultaneous SOC and cell capacity estimation using the MMAE technique, and Section 5 is the conclusion and future work.

3.2 Online SOC Estimation Using BKF

3.2.1 Modeling Lithium-ion Battery Cells

To estimate the SOC of a Lithium-ion cell, it is imperative to establish a mathematical model for the battery cell. Various types of cell models have been devised in the past to fulfill this requirement. Among them, two prominent categories are the Equivalent Circuit Model (ECM) and the Physics-Based Model (PBM)[1]. These models serve as critical tools for predicting and understanding the behavior of the

battery cell, aiding in the accurate estimation of SOC and related parameters. In this chapter, an ECM based Enhanced Self-Correcting (ESC) model will be introduced, followed by the derivation of a modified ESC model to be utilized within the context of the BKF approach.

3.2.2 ESC Cell Model

The effect of diffusion voltage, cell impedance, and hysteresis voltage are accounted for by the ESC model. Unlike other simple ECMs, the inclusion of a description of hysteresis voltage is made in the ESC model, resulting in a more accurate portrayal of cell behavior [1]. The circuit diagram for this model is displayed in Figure 3.1.

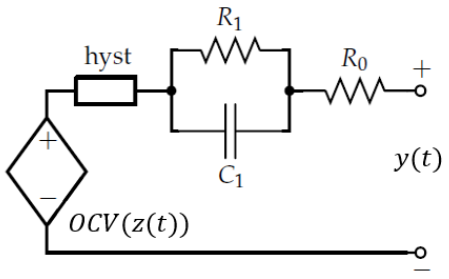

Figure 3.1: The Lithium-ion cell's ESC model [1]

In Figure 3.1, the cell terminal voltage is represented by $y(t)$, cell impedance is modeled by R_0, the diffusion voltage is represented by the parallel combination of

R_1 and C_1, and the hysteresis effect in the circuit model is denoted by the block labeled as "*hyst*." Within this chapter, $z(t)$ is employed to denote the SOC, and OCV signifies the open circuit voltage of the cell, which is typically formulated as a function of SOC [91].

In terms of these variables, the discrete-time state-space representation of the ESC model is presented below, with detailed derivations of this model available in the work by Plett [1].

$$\begin{aligned}\begin{bmatrix} z_{k+1} \\ i_{R_1,k+1} \\ h_{k+1} \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & A_{RC} & 0 \\ 0 & 0 & A_{H,k} \end{bmatrix} \begin{bmatrix} z_k \\ i_{R_1,k} \\ h_k \end{bmatrix} \\ &+ \begin{bmatrix} -\frac{\eta_k T}{Q} & 0 \\ 1 - A_{RC} & 0 \\ 0 & A_{H,k} - 1 \end{bmatrix} \begin{bmatrix} i_k \\ sgn(i_k) \end{bmatrix}\end{aligned} \tag{3.1}$$

$$y_k = OCV(z_k) + M_0 s_k + M h_k - R_1 i_{R_1,k} - R_0 i_k \tag{3.2}$$

Here in (3.1) and (3.2), we have

- z_k: state of charge, $0 \leq z_k \leq 1$
- $i_{R_1,k}$: current through R_1, A
- h_k: hysteresis voltage, V
- y_k: cell terminal voltage, V
- i_k: current through R_0 (load current), A

- $A_{RC} = \exp(-\frac{-T}{R_1 C_1})$
- $A_{H,k} = \exp(-\frac{\eta_k i_k \gamma T}{Q})$
- T: sampling time, s
- η_k: coulombic efficiency of the cell, $0 \leq \eta \leq 1$
- γ: a constant which tunes the rate of decay of the hysteresis voltage
- Q: total capacity of the cell, Ah
- OCV: open circuit voltage, V
- M_0: instantaneous hysteresis constant
- s_k: instantaneous hysteresis, V
- M: non-instantaneous hysteresis constant

Note that

- Usually, η_k in a typical lithium-ion cell is around 99% or higher [1], so in this model, it is considered as a constant and $\eta = 1$
- $s_k = \begin{cases} sgn(i_k), & |i_k| > 0 \\ s_{k-1}, & otherwise \end{cases}$ and $s_0 = 0$
- The OCV-SOC relationship is represented as

$$OCV(z_k) = K_0 - \frac{K_1}{z_k} - K_2 z_k + K_3 \ln(z_k) + K_4 \ln(1 - z_k) \tag{3.3}$$

where K_0 through K_4 are constants which satisfy the OCV-SOC relationship [91].

It can be noticed that (3.3) is nonlinear and therefore the estimation of SOC becomes a nonlinear estimation problem.

3.2.3 Modified ESC Model

In the ESC model described above, SOC is treated as a state variable, thereby allowing the establishment of an estimator, such as the EKF or the SPKF, to derive SOC estimates over time. However, as previously indicated, an inaccurate initialization of the SOC estimate may lead to a significant estimation error. In this study, modifications are applied to the ESC model and are incorporated into the MMAE algorithm to reduce the estimation error compared to EKF, regardless of the initial SOC estimate. In this adaptation, SOC is regarded as an unknown time-varying parameter, transforming the state estimation problem into an unknown parameter estimation challenge resolved through the MMAE algorithm.

Given the known range of SOC ($0 \leq z_k \leq 1$), SOC can be discretized as $z_1, z_2, \ldots, z_i, \ldots, z_N$ within the interval of 0 to 1. As the OCV is dependent on SOC, it is indirectly discretized. Consequently, models corresponding to each discretized SOC value can be readily established:

$$\begin{bmatrix} s_{k+1} \\ OCV_{i,k+1} \\ i_{R_1,k+1} \\ h_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & A_{RC} & 0 \\ 0 & 0 & 0 & A_{H,k} \end{bmatrix} \begin{bmatrix} s_k \\ OCV_{i,k} \\ i_{R_1,k} \\ h_k \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 - A_{RC} & 0 \\ 0 & A_{H,k} - 1 \end{bmatrix} \begin{bmatrix} i_k \\ sgn(i_k) \end{bmatrix} \tag{3.4}$$

$$y_k = \begin{bmatrix} M_0 & 1 & -R_1 & M \end{bmatrix} \begin{bmatrix} s_k \\ OCV_{i,k} \\ i_{R_1,k} \\ h_k \end{bmatrix} - R_0 i_k \tag{3.5}$$

where

$$OCV_i = K_0 - \frac{K_1}{z_i} - K_2 z_i + K_3 \ln(z_i) + K_4 \ln(1 - z_i) \tag{3.6}$$

at each time step k, and the method to compute s_k remains the same.

In comparison to the ESC model, s_k and OCV_i are now regarded as states. Given that s_k is associated with the load current i_k and OCV_i is a function of the quantized value of SOC, these quantities can be pre-computed at each time step. By designating them as state variables and conducting pre-processing, (3.4) and (3.5) are rendered linear. Consequently, the original nonlinear state estimation problem is transformed into a linear estimation problem solved in a parallel fashion.

3.2.4 Using MMAE Algorithm to Estimate SOC

In the MMAE technique, the unknown parameter of a system (in this work, the SOC of Lithium-ion cells) can be adaptively estimated by employing the control signal (in this case, the load current i_k) and measurement data (specifically, the terminal voltage y_k). SOC is discretized into a discrete set of N SOC values. With this set of quantized SOC values, N Kalman filters are configured, each targeting a distinct possible SOC from within this set. These filters are then assembled into a bank, with their common input being the load current i_k, and their outputs are the terminal voltage. The corresponding conditional probability for each potential SOC can be adaptively calculated using Bayes' rule from (3.7). The particular filter with the highest conditional probability is designated as the most likely SOC estimate at the current time step [2]. The block diagram of the MMAE algorithm for unknown SOC estimation is depicted in Fig. 3.11 [2].

Given that SOC is quantized into a finite discrete set, the conditional probability of each quantized SOC, denoted as z_i, given the set of all measurements Y_k up to time k, can be computed utilizing Bayes' rule:

$$p(z_i \mid Y_k) = \frac{p(y_k \mid Y_{k-1}, z_i)p(z_i \mid Y_{k-1})}{\sum_{i=1}^{N} p(y_k \mid Y_{k-1}, z_i)p(z_i \mid Y_{k-1})} \tag{3.7}$$

Within equation (3.7), $p(z_i \mid Y_{k-1})$ signifies the *a priori* conditional probability of $p(z_i \mid Y_k)$, and this equation can be recursively solved once $p(y_k \mid Y_{k-1}, z_i)$ and $p(z_i \mid Y_{k-1})$ are known. The initialization of the *a priori* probability $p(z_i \mid Y_0)$ is determined by design. It should be noted that, given SOC is a time-varying

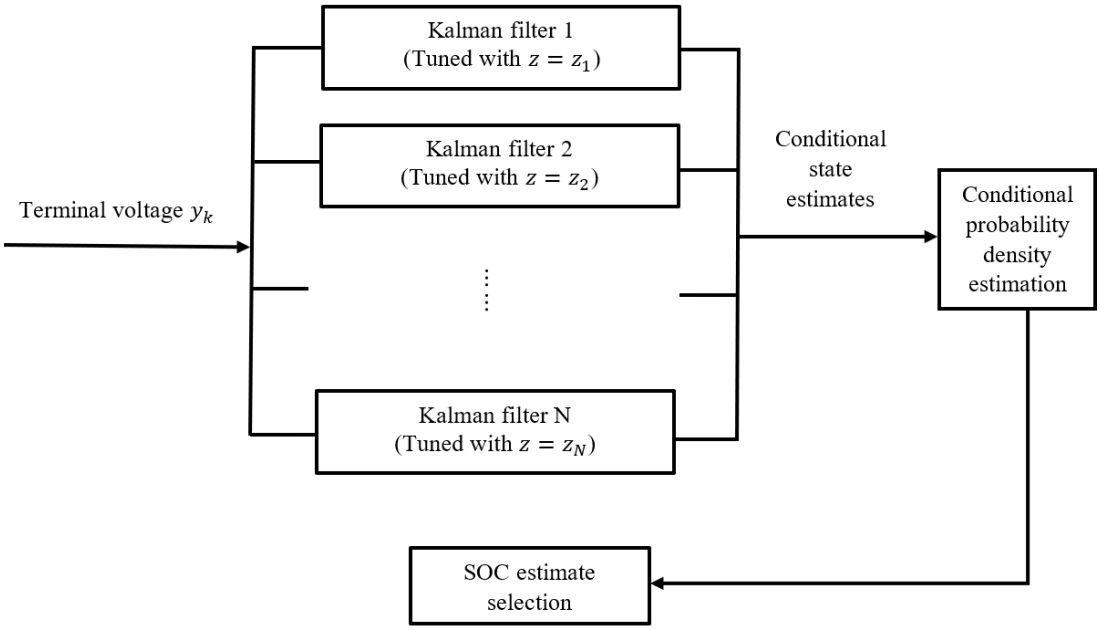

Figure 3.2: Block diagram of MMAE technique for SOC estimation [2].

parameter, it necessitates reinitialization at the outset of each iteration, and the sum of all *a priori* probabilities tailored to its respective SOC amounts to one.

In order to update $p(z_i \mid Y_k)$, the probability density function $p(y_k \mid Y_{k-1}, z_i)$ must be determined as well. It has been demonstrated that the convergence of the *a posteriori* probability is independent of the type of probability density function [2, 13]. As a result, all state and measurement noise in this study is assumed to exhibit a Gaussian distribution, leading to Gaussian conditional probabilities. In this context, $p(y_k \mid Y_{k-1}, z_i)$ can be expressed as follows:

$$p(y_k \mid Y_{k-1}, z_i) = (2\pi)^{-n/2} \left|\Omega_{k|z_i}^{-1}\right|^{1/2} \cdot \exp\left(-\frac{1}{2}\tilde{y}_{k|z_i}^T \Omega_{k|z_i}^{-1} \tilde{y}_{k|z_i}\right) \tag{3.8}$$

In (3.12), n denotes the order of the modified ESC model, $\tilde{y}_{k|z_i}$ represents the innovation sequence associated with z_i, and it is defined as:

$$\tilde{y}_{k|z_i} = y_k - \hat{y}_{k|k-1,z_i} \tag{3.9}$$

Here, $\hat{y}_{k|k-1,z_i}$ represents the estimated measurement signal linked to the Kalman filter with its corresponding SOC.

$\Omega_{k|z_i}$ stands as the innovation covariance and can be defined as follows:

$$\begin{aligned} \Omega_{k|z_i} &= E\left[(y_k - \hat{y}_{k|k-1,z_i})(y_k - \hat{y}_{k|k-1,z_i})^T\right] \\ &= C_k P_{k|z_i} C_k^T + G_k W_k G_k^T \end{aligned} \tag{3.10}$$

Here, $P_{k|z_i}$ denotes the error covariance matrix, which can be computed recursively

using the Riccati equation, based on the corresponding Kalman filter equations.

3.2.5 Pseudocode of MMAE Algorithm for Estimating SOC

In summary, the pseudocode of the MMAE technique to estimate SOC is generalized below [2]:

Algorithm 5 Estimating SOC using MMAE algorithm

Initialization 1: Quantize the unknown SOC to a finite set: $\{z_1, z_2, \ldots, z_i, \ldots, z_N\}$. (Note: N is the number of filters in the bank)

Initialization 2: Initialize the state estimate, $x_{0|z_i}$, the *a priori* probabilities, $p(z_i \mid Y_0)$ and the error covariance, $P_{0|z_i}$ for all filters in the bank.

for $k = 0, 1, 2...$ **do**

  for $i = 1, 2, ...N$ **do**

    $\Omega_{k,i} = C_{k,i} P_{k,i} C_{k,i}^T + G_k W_k G_k^T$ ▷ Set up Kalman filters

    $K_{k,i} = A_{k,i} P_{k,i} C_{k,i}^T \Omega_{k,i}^{-1}$

    $\hat{x}_{k+1,i} = A_{k,i} \hat{x}_{k,i} + B_k u_k + K_{k,i}(y_k - \hat{y}_{k,i})$

    $\hat{y}_{k,i} = C_{k,i} \hat{x}_{k,i} + D_k u_k$

    $P_{k+1,i} = A_{k,i} P_{k,i} A_{k,i}^T + F_k V_k F_k^T - K_{k,i} C_{k,i} P_{k,i} A_{k,i}^T$

    $\tilde{y}_{k,i} = y_k - \hat{y}_{k,i}$

    $p(y_k \mid Y_{k-1}, z_i) = (2\pi)^{-n/2} \left| \Omega_{k|z_i}^{-1} \right|^{1/2} \cdot \exp\left(-\frac{1}{2} \tilde{y}_{k,i}^T \Omega_{k,i}^{-1} \tilde{y}_{k,i}\right)$

    $p(z_i \mid Y_k) = \frac{p(y_k|Y_{k-1},z_i) p(z_i|Y_{k-1})}{\sum_{i=1}^{N} p(y_k|Y_{k-1},z_i) p(z_i|Y_{k-1})}$ ▷ Compute the *a posteriori* probabilities

  end for

  Find z_i with the largest $p(z_i \mid Y_k)$

  Reinitialize $p(z_i \mid Y_k)$

end for

3.2.6 The Case Study LiFePO4 Cell

The nominal values of the LiFePO4 cell utilized in this study are presented in Table 3.1. All nominal values originate from the manufacturer. To estimate SOC and cell capacity of the modified ESC cell model, parameters in (3.1), (3.2), and coefficients in (3.3) must be estimated in advance. Following the data acquisition and parameter estimation method outlined in [1], cell OCV-SOC data is measured at 22°C, after which the least-squares method is employed to estimate coefficients K_0 through K_4 in (3.3). The OCV-SOC relationship, as shown in Fig. 3.3, illustrates the measured data (in blue) and the estimated data (in red), which relies on the estimated parameters detailed in Table 3.2. Other parameters within the ESC model are likewise determined using the non-negative least-squares method as presented in [1], and the results are provided in Table 3.3.

Table 3.1: Nominal values of the LiFePO4 cell

Cell Type	LiFePO4 26650 Rechargeable Cell
Total Capacity	3.3Ah
Terminal Voltage	3.2V
Maximum Output Energy	10Wh

Table 3.2: Estimated OCV-SOC coefficients using least-squares method

K_0	3.3274
K_1	-6.1332×10^{-5}
K_2	0.0044
K_3	0.0498
K_4	-0.0107

Table 3.3: Estimated parameters of ESC model using non-negative least-squares method

Parameter	Value
R_0	$0.0117\,\Omega$
R_1	$0.0388\,\Omega$
γ	150
Hysteresis constant M	0.0013
Instantaneous Hysteresis Constant M_0	0
Time Constant R_1C_1	4.548

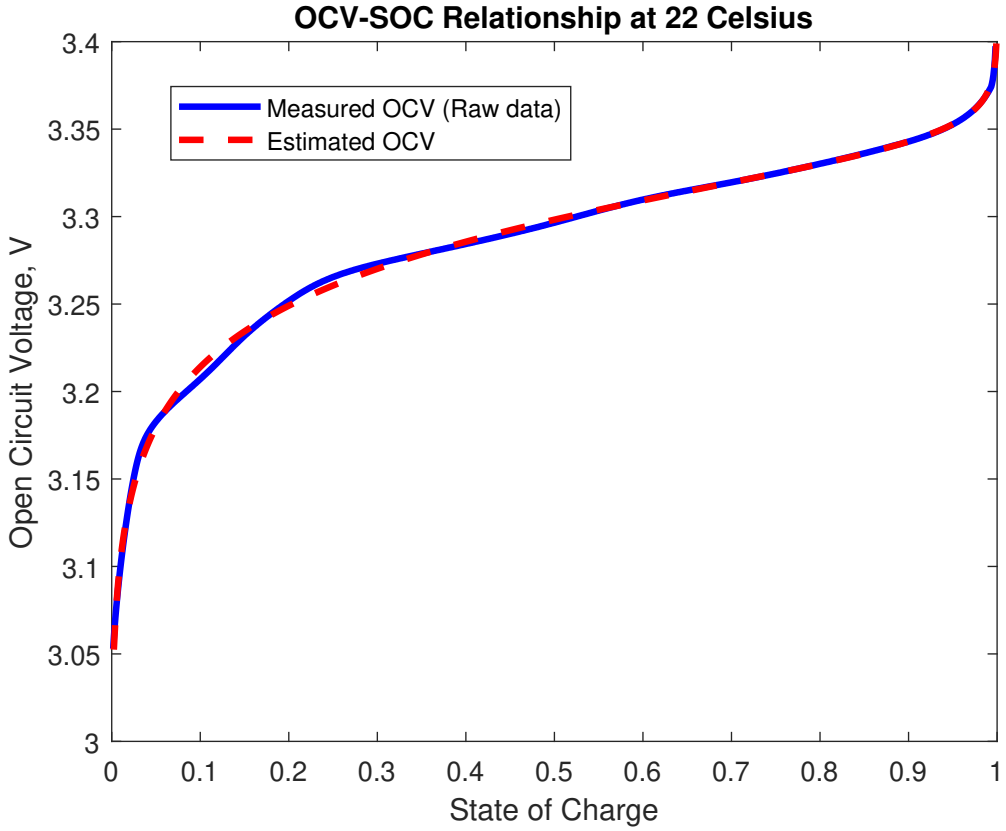

Figure 3.3: OCV-SOC relationship at 22°C

In terms of the initialization of the MMAE technique, each *a priori* probability linked to its respective Kalman filter is uniformly initialized to a probability of 0.01, signifying that the initial SOC has an equal likelihood of representing any of the quantized SOCs within the set. In comparison, the EKF results utilizing the ESC model are also subjected to simulation under identical system state and measurement noise levels. The initial error covariance matrix for both methods is configured

to be identical. Variations in SOC estimate initialization are introduced for both the MMAE and EKF techniques to evaluate which of the methods yields a smaller estimation error.

3.2.7 Results at Different Initial SOC Estimates

Figure 3.4 presents the actual SOC, the estimated SOC using the EKF and the MMAE approaches, and the corresponding absolute error $|z_k - \hat{z}_k|$ in percentage. In Figure 3.4 (a), it can be observed that the cell is fully charged, with the actual SOC denoted as $z_0 = 1$. Subsequently, it undergoes discharge at a rate of 1C. Notably, the C rate serves as a relative measure of the cell's current during charging or discharging [2]. For instance, if a fully charged cell with a total capacity of 10Ah is discharged entirely using a constant current of 2A for 5 hours, the C rate is expressed as $\frac{C}{5}$. Both the EKF and the MMAE approaches assume an initial SOC estimate of 0.5. This demonstration highlights that despite the less-than-ideal initial SOC estimate, the SOC estimation using the MMAE approach rapidly converges to the actual SOC, while the SOC estimates obtained with the EKF approach exhibit relatively slower convergence.

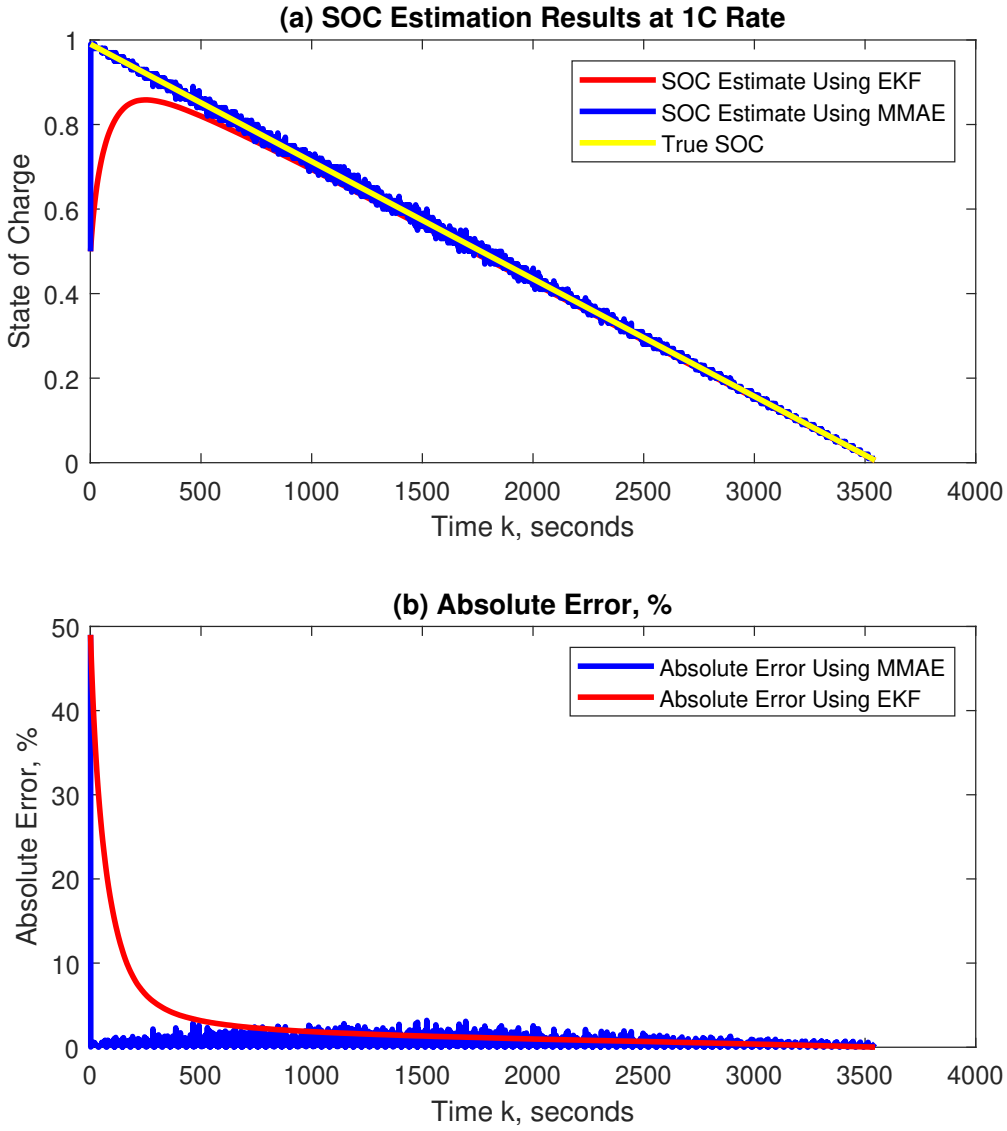

Figure 3.4: SOC estimation result at $\hat{z}_0 = 0.5$

In Figure 3.4 (b), the absolute error for end of the EKF and the MMAE results is computed over time. It is evident that the MMAE results yield relatively smaller absolute errors at the beginning of the simulation, whereas the EKF results exhibit a similar level of absolute error after approximately $k = 500s$. It is worth noting that the MMAE approach introduces a slightly larger absolute error after about $k = 1000s$ compared to the EKF results. This behavior can be attributed to the MMAE algorithm's mechanism and the quantization level of SOC. In this study, SOC is quantized as a set, denoted as $z_i \in \{0.01, 0.02, \ldots, 0.99, 1\}$ which has one hundred values. The MMAE technique computes the *a posteriori* probabilities only in relation to each quantized SOC value within the set. If the actual value falls outside the set, the algorithm converges to the nearest SOC value within the set, potentially introducing quantization errors to the estimation results. While increasing the number of quantized SOC values within the set can enhance estimation accuracy, it would also lead to a substantial increase in computational cost. In addition, it may also fail in convergence criteria.

In Figure 3.5 (a), a simulation is repeated with an initial SOC estimate of $\hat{z}_0 = 0.2$ for both the MMAE and the EKF techniques. In comparison to the previous results, the initial estimates are even less accurate for both approaches. Nevertheless, the MMAE technique still exhibits rapid convergence in comparison to the EKF results. The absolute errors for both approaches are displayed in Figure 3.5 (b), further confirming that the MMAE approach converges more quickly and results in an overall smaller error.

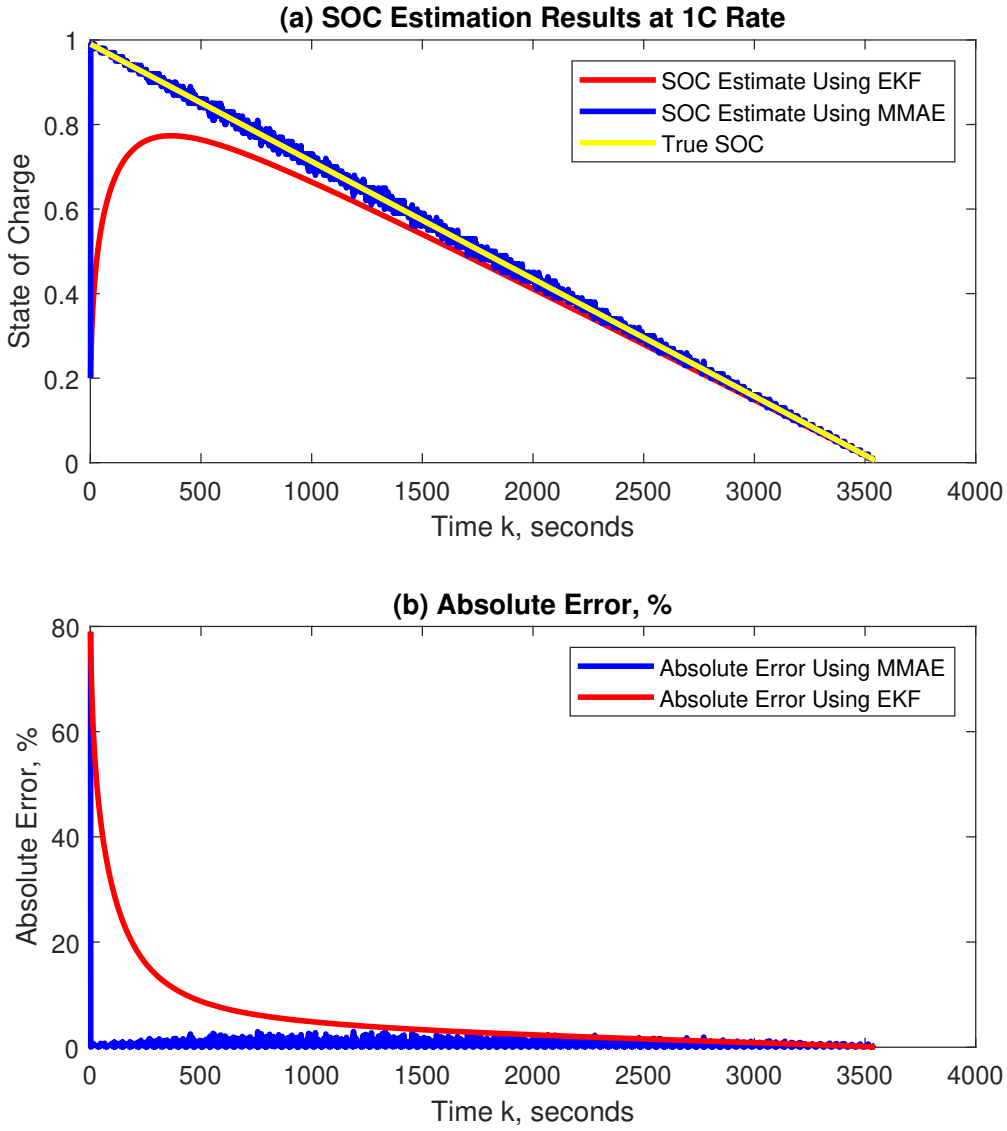

Figure 3.5: SOC estimation result at $\hat{z}_0 = 0.2$

In Figure 3.6, it is postulated that the cell initiates its discharge while it is not at full charge. In this simulation, the actual SOC is set at $z_0 = 0.2$, signifying that the cell possesses only 20% of its total energy capacity. The cell continues to discharge at a rate of 1C. However, in this instance, the initial SOC estimates for both the MMAE and the EKF are established at $\hat{z}_0 = 1$, indicating that both cell is assumed to be fully charged.

The estimation results in Figure 3.6 (a) reveal that even with $\hat{z}_0 = 1$, the MMAE technique swiftly converges to the actual SOC, whereas the EKF method converges to the true SOC after approximately $k = 300s$. The estimation error displayed in Figure 3.6 (b) corroborates the rapid convergence.

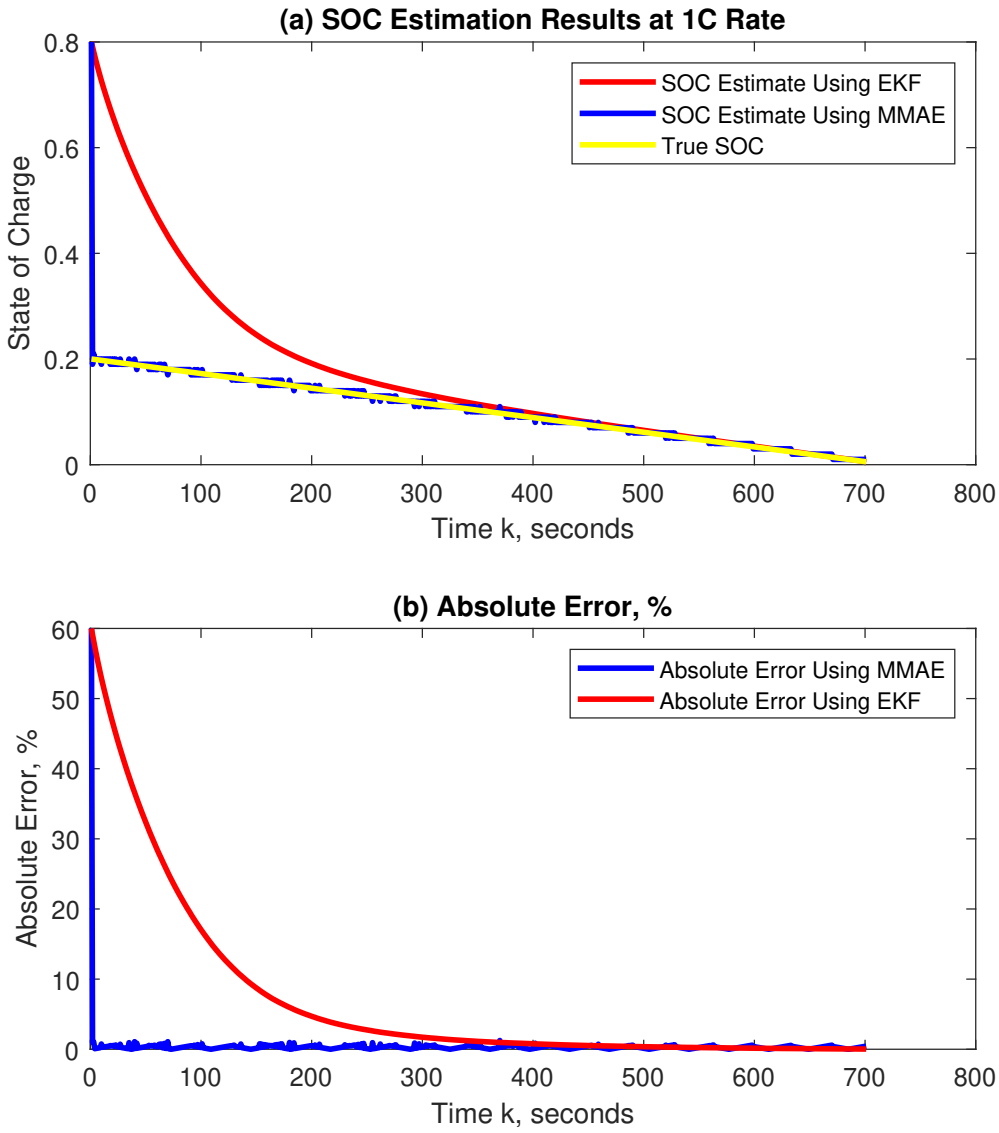

Figure 3.6: SOC estimation result at $\hat{z}_0 = 0.8$

At times, a favorable estimation outcome can be observed when the initial SOC estimate closely approximates the actual initial SOC. In Figure 3.7, simulation results for a duration of 500 seconds are presented, with an initial assumption of $\hat{z}_0 = 0.9$ for both the EKF and MMAE techniques. In this scenario, both methods converge swiftly to the actual SOC. However, the EKF result exhibits an overshoot at the commencement of the simulation, resulting in an impractical SOC estimate, considering the valid SOC range of $0 \leq z_k \leq 1$. Such an overshoot could pose practical issues. In contrast, the MMAE approach remains effective, and the fluctuation in the SOC estimate is attributed to the previously mentioned quantization effect.

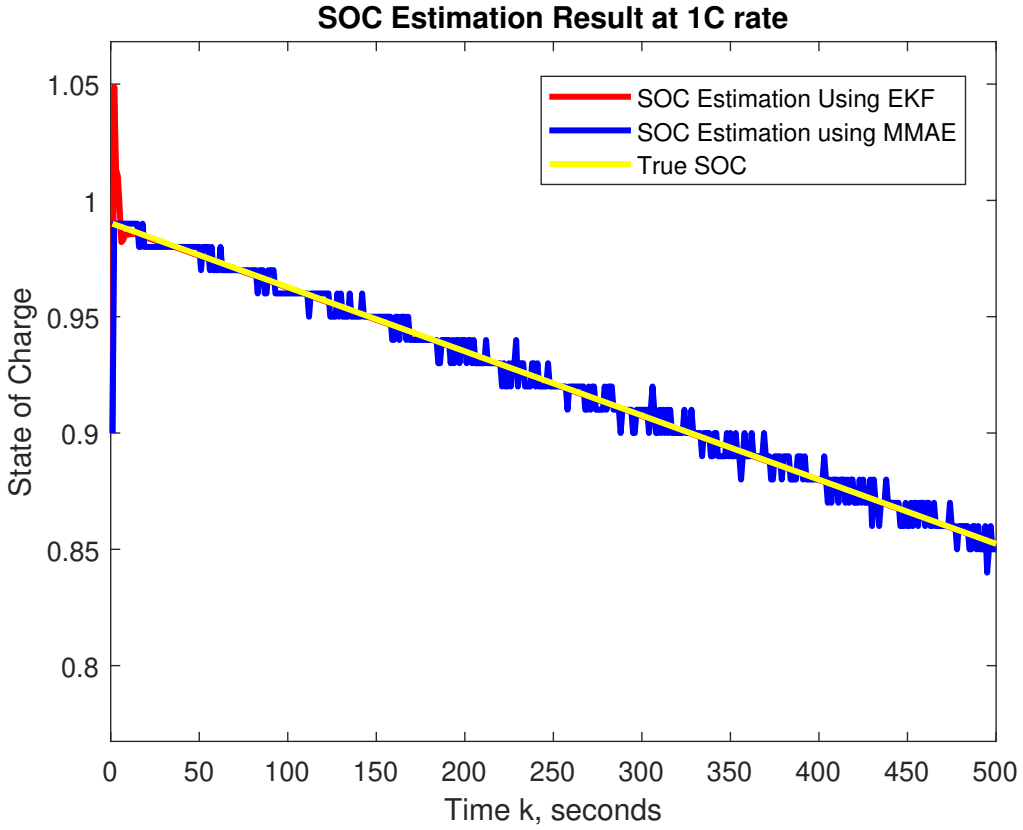

Figure 3.7: SOC estimation result at $\hat{z}_0 = 0.9$

3.3 Improved SOC Estimation Using Combined MMAE and EKF Approach

Previous simulation results have effectively showcased the utility of the MMAE approach for SOC estimation. However, it is essential to recognize that the MMAE technique exhibits a slightly larger absolute error after swiftly converging to the

actual SOC value, as shown in Figures 3.4, 3.5, 3.6, and 3.7, whereas the EKF method yields a smaller estimation error after a slower convergence. This behavior can be attributed to the mechanics of the MMAE algorithm and the quantization level employed for SOC. As shown previously, SOC is quantized into a discrete set, represented as $z_i \in \{0.01, 0.02, \ldots, 0.99, 1\}$, comprising one hundred distinct values. The MMAE technique exclusively computes *a posteriori* probabilities in relation to each quantized SOC value within this predefined set. Should the actual value deviate from the set, the algorithm converges to the nearest SOC value within the set, potentially introducing quantization errors into the estimation results. While augmenting the number of quantized SOC values within the set has the potential to enhance estimation accuracy, it also incurs a significant increase in computational cost.

To address this issue, this section introduces an improved approach that combines the EKF and MMAE techniques for SOC estimation in LiFePO4 cells. The primary aim is to mitigate the computational overhead associated with the MMAE approach and further enhance SOC estimation accuracy. The concept behind this improved approach is straightforward. In this refined methodology, SOC is initially estimated using the MMAE algorithm to obtain a precise initial SOC estimate, effectively countering the EKF's high sensitivity to initial estimation errors. Subsequently, the EKF is employed to estimate SOC using the initial estimate obtained from the MMAE technique. The transition from MMAE to EKF is motivated by the fact that while the MMAE excels in delivering superior estimation accuracy when the initial SOC estimate is suboptimal, it also incurs a substantial computational cost since multiple

models run in parallel at each iteration. To ameliorate the computational burden of the MMAE technique, the EKF is introduced after obtaining an accurate initial SOC estimate via MMAE. Given that the EKF can rapidly converge to the actual SOC with a precise initial SOC estimate furnished by the MMAE, a substantial reduction in computational cost is achieved. Additionally, since the MMAE necessitates the quantization of SOC, switching to the EKF eliminates quantization errors. Consequently, this approach serves to enhance estimation accuracy significantly. The amalgamation of the MMAE and the EKF successfully mitigates their respective shortcomings, resulting in a noteworthy improvement in estimation accuracy compared to either of these methods in isolation.

3.3.1 Pseudocode of SOC Estimation Using MMAE and EKF Approach

In this enhanced approach, the utilization of the MMAE technique is confined solely to the initialization of the SOC estimate. Specifically, MMAE comes into play exclusively at the initial time step, i.e., when $k = 0$. Following the initial iteration at $k = 1$, an inherently more accurate SOC estimate is derived. Subsequently, this refined SOC estimate assumes the role of the initial SOC estimate for the EKF, which subsequently takes charge of SOC estimation in time. To provide a comprehensive overview, the generalized pseudocode of this integrated technique is shown below:

Algorithm 6 Combined SOC estimation technique

Initialization 1: Quantize the unknown SOC to a finite set: $\{z_1, z_2, \ldots, z_i, \ldots, z_N\}$. (N is the number of filters in the bank)

Initialization 2: Initialize the state estimate, $\hat{x}_{0,i} = \bar{x}_i$, the *a priori* probabilities, $p(z_i \mid Y_0) = \frac{1}{N}$ and the error covariance, $P_{0|z_i} = X_{0|z_i}$ for all filters in the bank.

if $k = 0$ **then**

    % Use MMAE technique to find initial SOC estimate:

    for $i = 1, 2, ..., N$ **do**

        % Set up Kalman filters:

        $\Omega_{k,i} = C_{k,i} P_{k,i} C_{k,i}^T + G_k W_k G_k^T$

        $K_{k,i} = A_{k,i} P_{k,i} C_{k,i}^T \Omega_{k,i}^{-1}$

        $\hat{x}_{k+1,i} = A_{k,i} \hat{x}_{k,i} + B_k u_k + K_{k,i}(y_k - \hat{y}_{k,i})$

        $\hat{y}_{k,i} = C_{k,i} \hat{x}_{k,i} + D_k u_k$

        $P_{k+1,i} = A_{k,i} P_{k,i} A_{k,i}^T + F_k V_k F_k^T - K_{k,i} C_{k,i} P_{k,i} A_{k,i}^T$

        % Compute the *a posteriori* probabilities:

        $\tilde{y}_{k,i} = y_k - \hat{y}_{k,i}$

        $p(y_k \mid Y_{k-1}, z_i) = (2\pi)^{-n/2} \left| \Omega_{k|z_i}^{-1} \right|^{1/2} \cdot \exp\left(-\frac{1}{2} \tilde{y}_{k,i}^T \Omega_{k,i}^{-1} \tilde{y}_{k,i}\right)$

        $p(z_i \mid Y_k) = \frac{p(y_k|Y_{k-1},z_i) p(z_i|Y_{k-1})}{\sum_{i=1}^{N} p(y_k|Y_{k-1},z_i) p(z_i|Y_{k-1})}$

    end for

    Find $\hat{z}_i$ with the largest $p(z_i \mid Y_k)$

else

    Set up EKF to keep estimating SOC using the knowledge of $\hat{z}_i$ from MMAE technique

end if

3.3.2 Simulation Results Using Combined Estimation Approach

To showcase the advantages of this integrated estimation technique, simulations are also conducted separately for the EKF and MMAE techniques, allowing for a comparative analysis of SOC estimation accuracy and convergence speed. In the context of the combined estimation technique, each *a priori* probability associated with its corresponding Kalman filter is uniformly initialized to a probability of 0.01. This initialization corresponds to a quantization level of $N = 100$ for SOC, mirroring the setup employed in the previous MMAE simulations. Additionally, EKF results are simulated using the ESC model, and consistent system state and measurement noise levels are maintained across all three techniques.

In order to assess the performance of each technique thoroughly, multiple simulations are conducted. Each simulation commences with varying initial SOC estimates. Absolute errors are computed for all three techniques, enabling a comprehensive comparison of their respective performance. The technique demonstrating the lowest estimation error $z_k - \hat{z}_k$ ultimately signifies the most accurate SOC estimation approach.

3.3.2.1 Results at Different Initial SOC Estimates

The actual SOC, as well as the estimated SOC using the EKF, the MMAE, and the combined estimation approach, along with the corresponding absolute percentage errors, are depicted in Fig. 3.8. In Fig. 3.8 (a), the cell commences the simulation in a fully charged state, signifying an actual initial SOC of $z_0 = 1$. Subsequently, it is discharged at a rate of 1C, implying a cell current of $3.3A$.

In this scenario, the initial SOC estimates for the EKF, the MMAE, and the combined estimation approach are all set at 0.5, underscoring the relatively poor quality of the initial SOC estimation. Remarkably, in Fig. 3.4 (a), it becomes evident that even though the initial SOC estimates are suboptimal, both the SOC estimates produced by the combined estimation approach (green line) and the MMAE approach (blue line) converge rapidly to the actual SOC. Conversely, the SOC estimate generated by the EKF (red line) demonstrates a comparably sluggish convergence. Notably, while the SOC estimate from the MMAE technique converges quickly, it exhibits some degree of noise in the estimation result. This noise can be attributed to the quantization level applied to the SOC as discussed before.

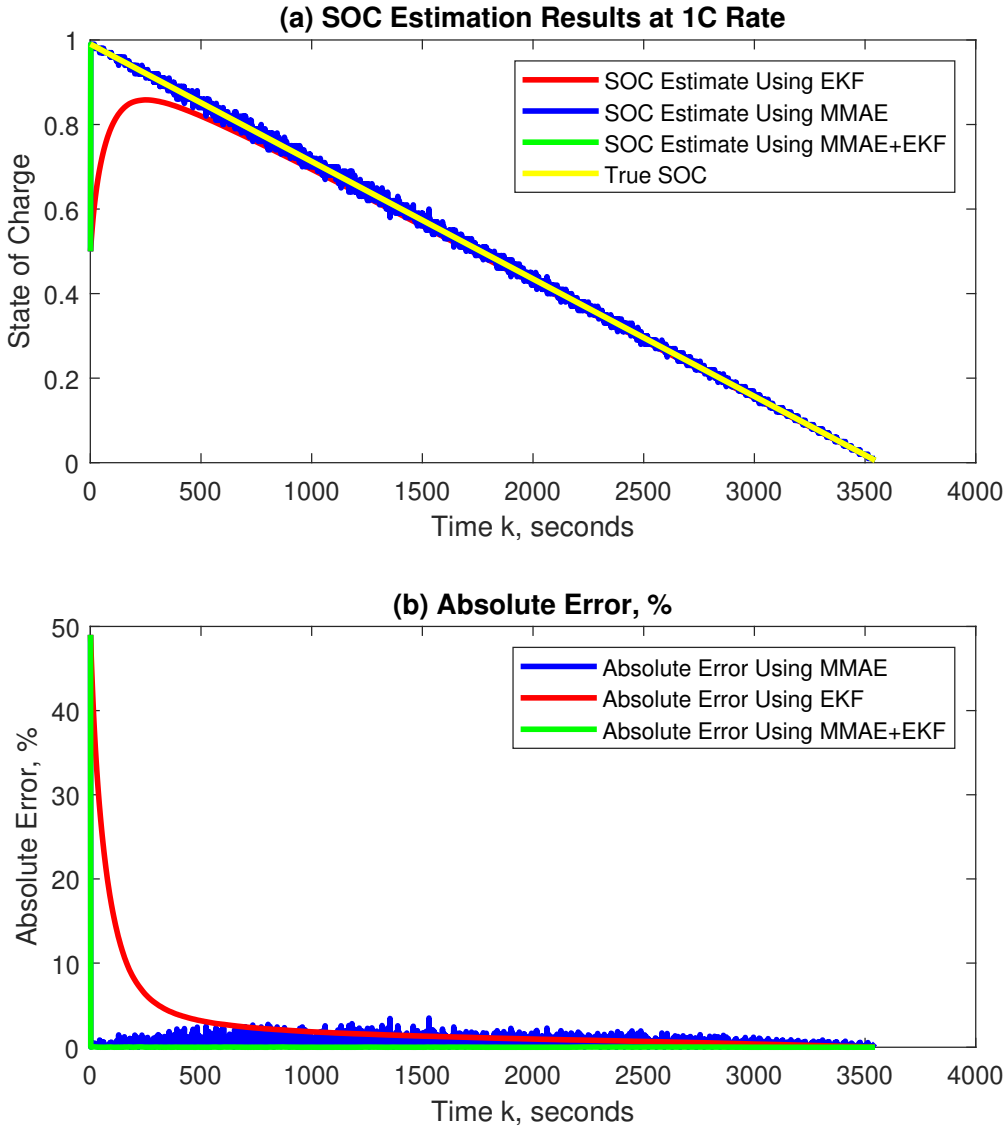

Figure 3.8: SOC estimation result at $\hat{z}_0 = 0.5$

In Fig. 3.8 (b), the absolute errors for each of the three approaches are calculated over time. It becomes readily apparent that the combined estimation technique (depicted by the green line) yields the smallest absolute error, highlighting its superior performance. Conversely, the MMAE approach (represented by the blue line) exhibits an error profile that appears noisy, consistent with the earlier explanation relating to quantization-induced errors. The EKF results (indicated by the red line) feature the largest absolute errors among the three techniques. This outcome is attributed to the EKF's relatively sluggish convergence when dealing with initially poor SOC estimates.

In Fig. 3.9 (a), the simulation is repeated, with the initial SOC estimate ($\hat{z}_0$) set at 0.2 for all three techniques. In comparison to the previous scenario, this initialization is even less favorable. Nevertheless, the combined estimation technique continues to demonstrate a rapid convergence rate and maintains the smallest absolute error, outperforming both the MMAE and EKF results. A visual representation of the absolute errors for all approaches is provided in Fig. 3.9 (b), reaffirming that the combined estimation technique consistently yields the most accurate results.

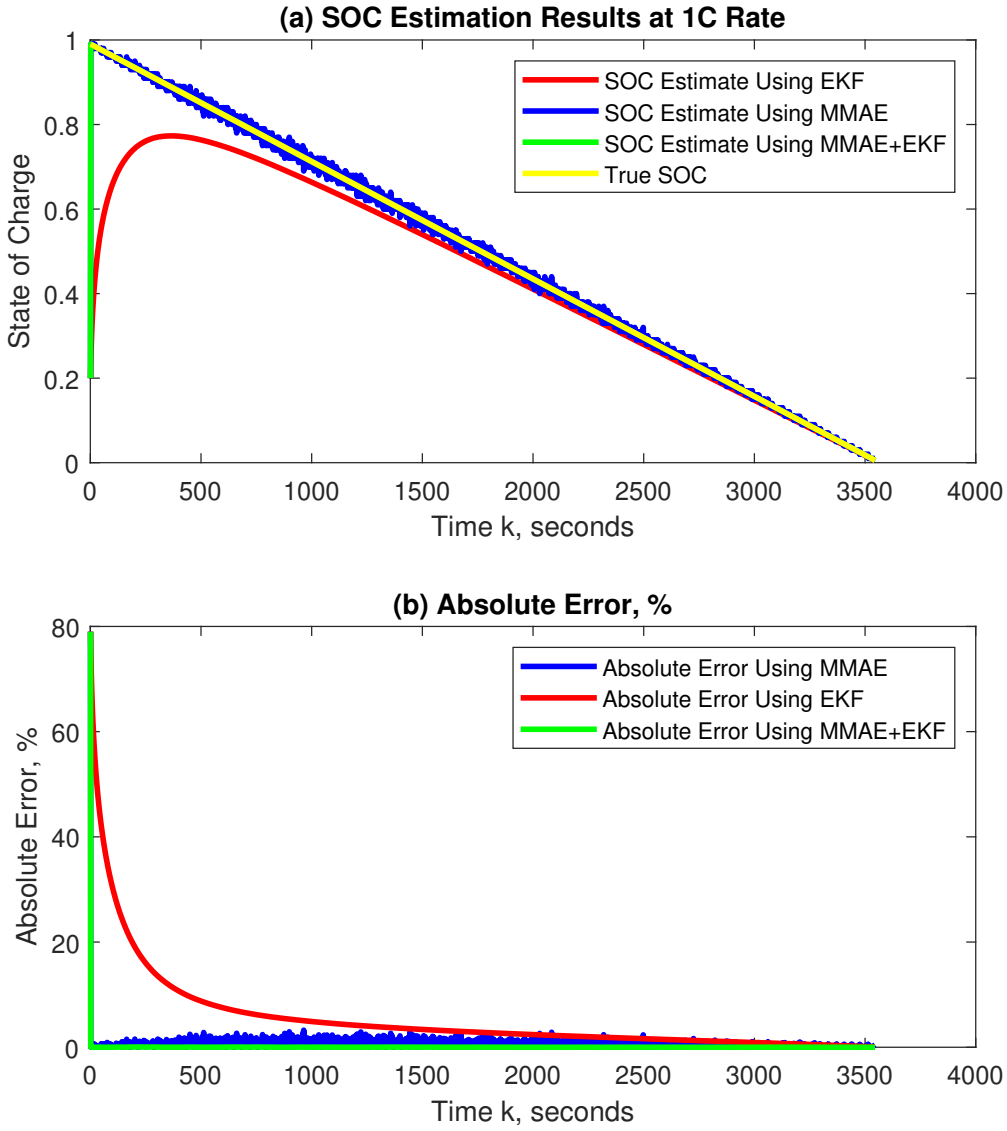

Figure 3.9: SOC estimation result at $\hat{z}_0 = 0.2$

In the simulation results presented in Fig. 3.10, the actual state of charge (SOC) is established at $z_0 = 0.5$. However, in this scenario, a notable distinction is made by setting the initial SOC estimates for all three techniques at $\hat{z}_0 = 0.9$. This implies that all the estimators are beginning with an overestimate of the SOC, assuming that the cell is nearly fully charged, even though the actual SOC is significantly lower. The cell continues to discharge at a constant 1C rate.

The results, as depicted in Fig. 3.10 (a), illustrate that despite the initial overestimation of z_0, both the combined estimation technique (depicted by the green line) and the MMAE algorithm (indicated by the blue line) are capable of swiftly converging to the actual SOC. In stark contrast, the EKF approach (illustrated by the red line) exhibits a markedly slower rate of convergence. The estimation error, as exhibited in Fig. 3.10 (b), serves to substantiate that the combined estimation technique (green line) continues to deliver the highest level of estimation accuracy.

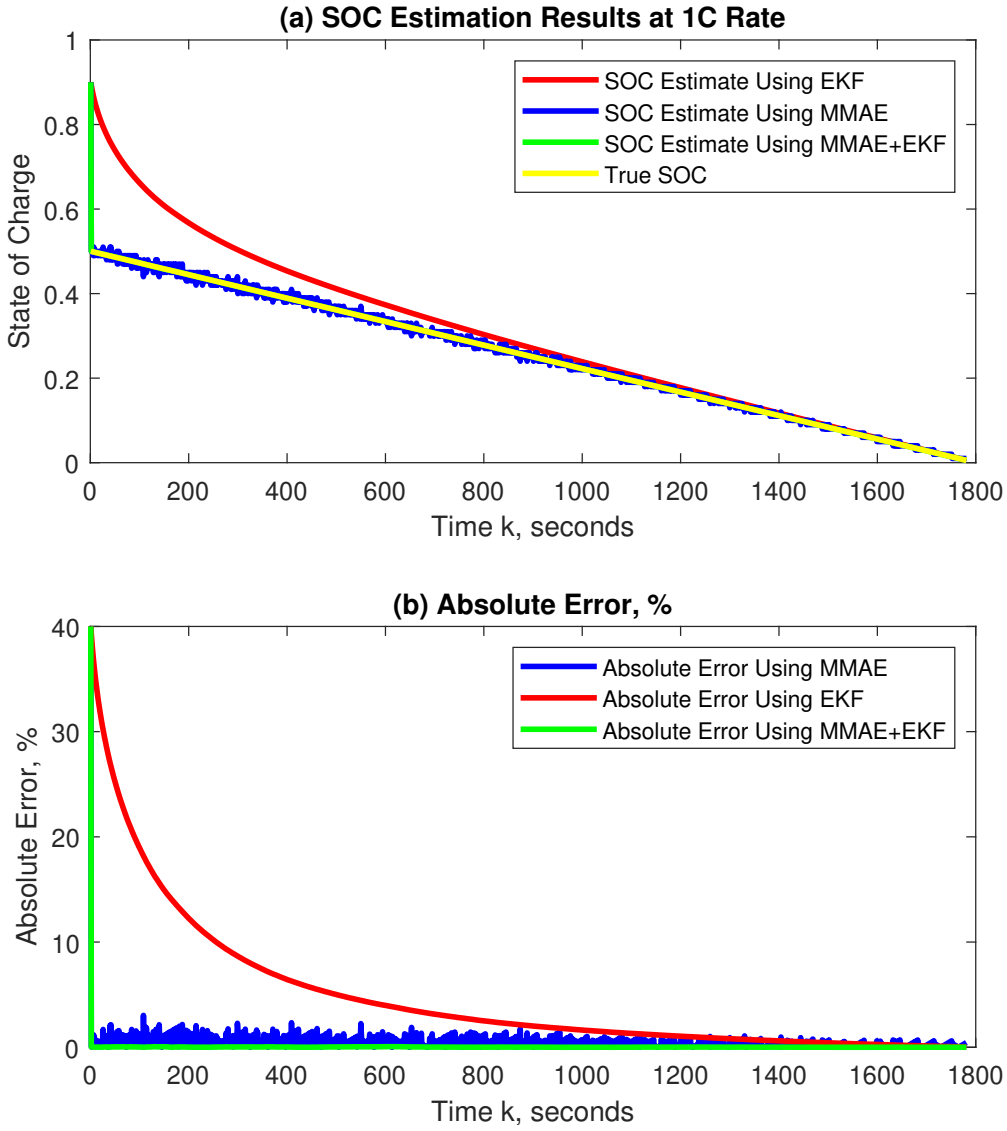

Figure 3.10: SOC estimation result at $\hat{z}_0 = 0.9$

3.4 Simultaneous SOC and Cell Capacity Estimation Using MMAE Approach

In previous sections, the estimation of SOC was introduced using the MMAE approach, along with an improved estimation method. However, in this section, the focus shifts towards the simultaneous estimation of both SOC and total capacity for a LiFePO4 cell, employing the MMAE approach. Unlike previous sections, the primary objective here is to estimate both SOC and cell capacity concurrently, when the initial values of either SOC or total capacity, or even both, are unknown. To accomplish this objective, the total capacity of the cell will be discretized into a finite set of possibilities.

Through the implementation of the MMAE technique, the conditional SOC estimate is initially determined, along with the conditional probability associated with each potential capacity value. Subsequently, the estimates for cell SOC and capacity are computed using a data fusion technique. To showcase the effectiveness of this proposed method, comparisons will be made with the dual and joint estimation techniques, as detailed in [1, 3]. This approach offers robust and accurate SOC and cell capacity estimates, particularly when the initial estimations of SOC and total capacity are less than optimal.

3.4.1 ESC Model Quantization

Concerning the modified ESC, as illustrated in (3.4) and (3.5), the original ESC model, as depicted in (3.1) and (3.2), is employed directly to concurrently estimate

both SOC and cell total capacity Q. In this context, the cell's total capacity Q is discretized into a finite set, denoted as $Q_1, Q_2, \ldots, Q_i, \ldots, Q_N$, where each Q_i represents a feasible value for the actual cell total capacity. This implies that a total of N cell models are established, each containing a distinct Q value from the quantized set.

It is important to note that, as the actual total capacity of an aged cell typically tends to be smaller than the nominal total capacity, the upper bound of the set is often designated as the nominal Q value, while the lower bound is set significantly higher than zero [1]. This aligns with common practice, where aged cells are usually replaced if their actual capacity falls below approximately 80% of the nominal value [1, 3]. Further details regarding the quantization levels will be elaborated upon in subsequent sections.

Having established the N cell models, a collection of N EKFs can be assembled to simultaneously compute the conditional state estimates from each model. Through the application of Bayesian data fusion technique, the most likely capacity value Q is chosen from within the set at each iteration. This choice of capacity value allows for the computation of the SOC. A visual representation of the MMAE algorithm is depicted in Figure 3.11, illustrating the concurrent estimation of SOC and total capacity [3].

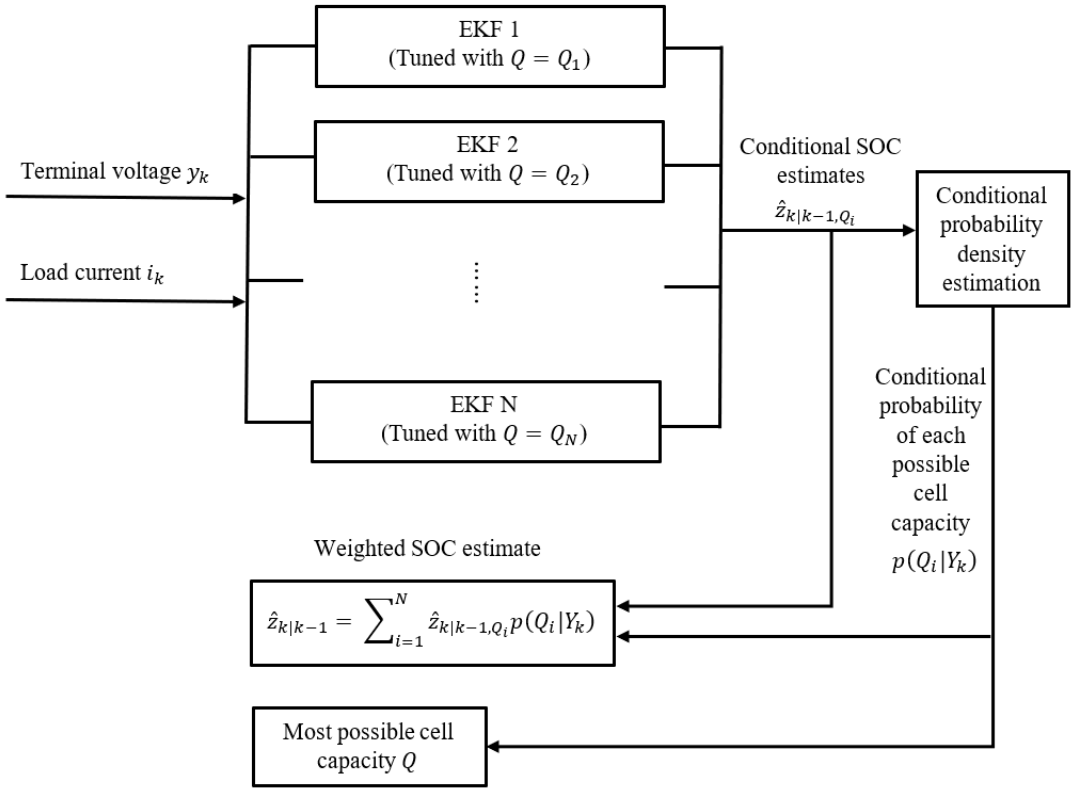

Figure 3.11: Block diagram of MMAE technique for SOC and total capacity estimation [3].

Figure 3.11 illustrates the arrangement of filters within a bank, wherein the shared inputs are the cell terminal voltage y_k and the load current i_k. These inputs serve as the foundation for adaptively calculating the corresponding conditional probabilities for each possible total capacity Q. The utilization of Bayes' rule in this context enables the computation of these conditional probabilities. Specifically, the conditional probability $p(Q_i \mid Y_k)$ is expressed as follows:

$$
\begin{aligned}
p(Q_i \mid Y_k) &= \frac{p(Y_k, Q_i)}{p(Y_k)} \\
&= \frac{p(Y_k, Q_i)}{\sum_{i=1}^{N} p(Y_k \mid Q_i) p(Q_i)} \\
&= \frac{p(y_k \mid Y_{k-1}, Q_i) p(Q_i \mid Y_{k-1})}{\sum_{i=1}^{N} p(y_k \mid Y_{k-1}, Q_i) p(Q_i \mid Y_{k-1})}
\end{aligned}
\tag{3.11}
$$

In (3.11), the variable y_k denotes the terminal voltage of a cell at time k, while Q_i represents one of the quantized values of cell capacity tuned for its corresponding EKF. The recursive solution of (3.11) hinges on the availability of two pieces of information: the conditional probability $p(y_k \mid Y_{k-1}, Q_i)$ and the *a priori* conditional probability $p(Q_i \mid Y_{k-1})$. Notably, the initial *a priori* probability $p(Q_i \mid Y_0)$ is determined by the designer's choice and is often uniformly distributed unless additional knowledge is available to refine the *a priori* probability. It is crucial to emphasize that the sum of all *a priori* probabilities for each Q_i at time k equals one.

In order to update $p(Q_i \mid Y_k)$, it is essential to determine the probability density function $p(y_k \mid Y_{k-1}, Q_i)$. It is worth noting that the nature of the probability density function does not impact the convergence of the *a posteriori* probability, as established in previous studies [2, 3, 11]. Therefore, in this work, it is assumed that both the state and measurement noise follow a Gaussian distribution, resulting in Gaussian conditional probabilities. Consequently, the expression for $p(y_k \mid Y_{k-1}, Q_i)$ takes the form:

$$
p(y_k \mid Y_{k-1}, Q_i) = (2\pi)^{-n/2} \left| \Omega_{k|Q_i}^{-1} \right|^{1/2} \exp\left(-\frac{1}{2} \tilde{y}_{k|Q_i}^{T} \Omega_{k|Q_i}^{-1} \tilde{y}_{k|Q_i} \right) \tag{3.12}
$$

Within (3.12), $n = 3$ corresponds to the order of the ESC model, and $\tilde{y}_{k|Qi}$ represents the innovation sequence pertaining to the specific Q_i:

$$\tilde{y}_{k|Q_i} = y_k - \hat{y}_{k|k-1,Q_i} \tag{3.13}$$

Here, $\hat{y}_{k|k-1,Qi}$ signifies the estimated measurement signal associated with the EKF corresponding to the particular cell capacity Q_i.

The conditional covariance matrix $\Omega_{k|Q_i}$ is defined as:

$$\begin{aligned} \Omega_{k|Q_i} &= E\left[(y_k - \hat{y}_{k|k-1,Q_i})(y_k - \hat{y}_{k|k-1,Q_i})^T\right] \\ &= C_k P_{k|Q_i} C_k^T + G_k W_k G_k^T \end{aligned} \tag{3.14}$$

Here, $P_{k|Q_i}$ denotes the error covariance matrix, which is recursively computed using the corresponding Riccati equation from each EKF.

Once the conditional probability $p(Q_i \mid Y_k)$ is computed for each possible Q_i value, the SOC estimate that accounts for the uncertainty in cell capacity can be calculated with the given conditional probability. This estimate combines each SOC estimate weighted by the corresponding conditional probability:

$$\hat{z}_{k|k-1} = \sum_{i=1}^{N} \hat{z}_{k|k-1,Q_i} p(Q_i \mid Y_k) \tag{3.15}$$

Here, $\hat{z}_{k|k-1,Qi}$ represents the conditional SOC estimate, calculated by the corresponding EKF adjusted for Q_i.

3.4.2 Simulation Results

To evaluate and compare the estimation accuracy and validate the robustness of the proposed MMAE technique, the joint and the dual estimation techniques are also employed to estimate SOC and cell capacity over time [3]. All three techniques undergo simulations using consistent initial SOC and cell capacity estimates, as well as the same error covariance matrix. Furthermore, a load current pattern based on the urban dynamometer drive schedule (UDDS) [103] serves as the input signal for all simulations. Performance evaluation is carried out by computing the absolute error $|z_k - \hat{z}_k|$ and $|Q - \hat{Q}_k|$ for each of the techniques.

3.4.3 Results at $z_0 = 1$ and $Q = 2.8Ah$

Simulation results are presented in Figures 3.12 and 3.13 to compare the performance of the MMAE technique against the dual and joint estimation methods.

In Figures 3.12 (a) and 3.13 (a), the LiFePO4 cell is initially fully charged with an unknown SOC, which is represented by the assumed initial SOC estimate of $\hat{z}_0 = 0.5$ for all the estimation approaches. This value reflects a poor initial SOC setup typically encountered in practical scenarios. The cell is then discharged at a rate of $\frac{C}{3}$, leading to a SOC of $z = 0$ after 3 hours.

Figure 3.12 (a) illustrates that the SOC estimate using the MMAE technique (blue line) rapidly converges to the actual SOC (red line), despite the unfavorable initialization of the SOC estimate. In contrast, the results obtained with the Joint (blue line) and Dual (green line) estimation techniques, as shown in Figure 3.13 (a), exhibit significant noise in the SOC estimates. Notably, the SOC estimate

produced by both the joint and dual methods initially exceeds the upper bound of SOC ($0 \leq z \leq 1$). This failure to adhere to the constraint creates substantial estimation errors.

Figures 3.12 (b) and 3.13 (b) provide a clearer perspective by presenting the absolute percentage error in SOC estimates, which aligns with the observations made above.

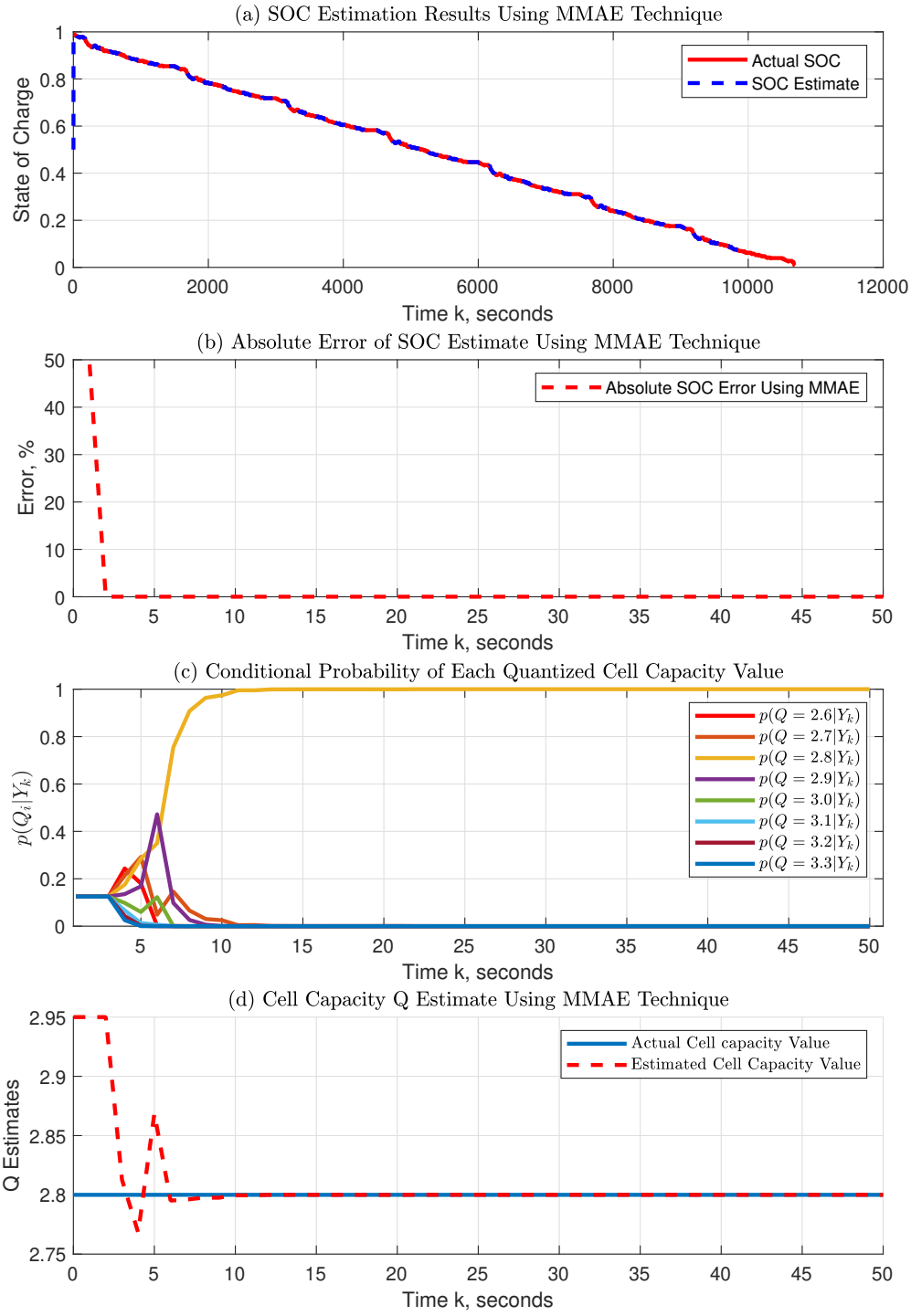

Figure 3.12: Estimation results at $\hat{z}_0 = 0.5$, Q quantized from 2.6 to 3.3 using MMAE technique

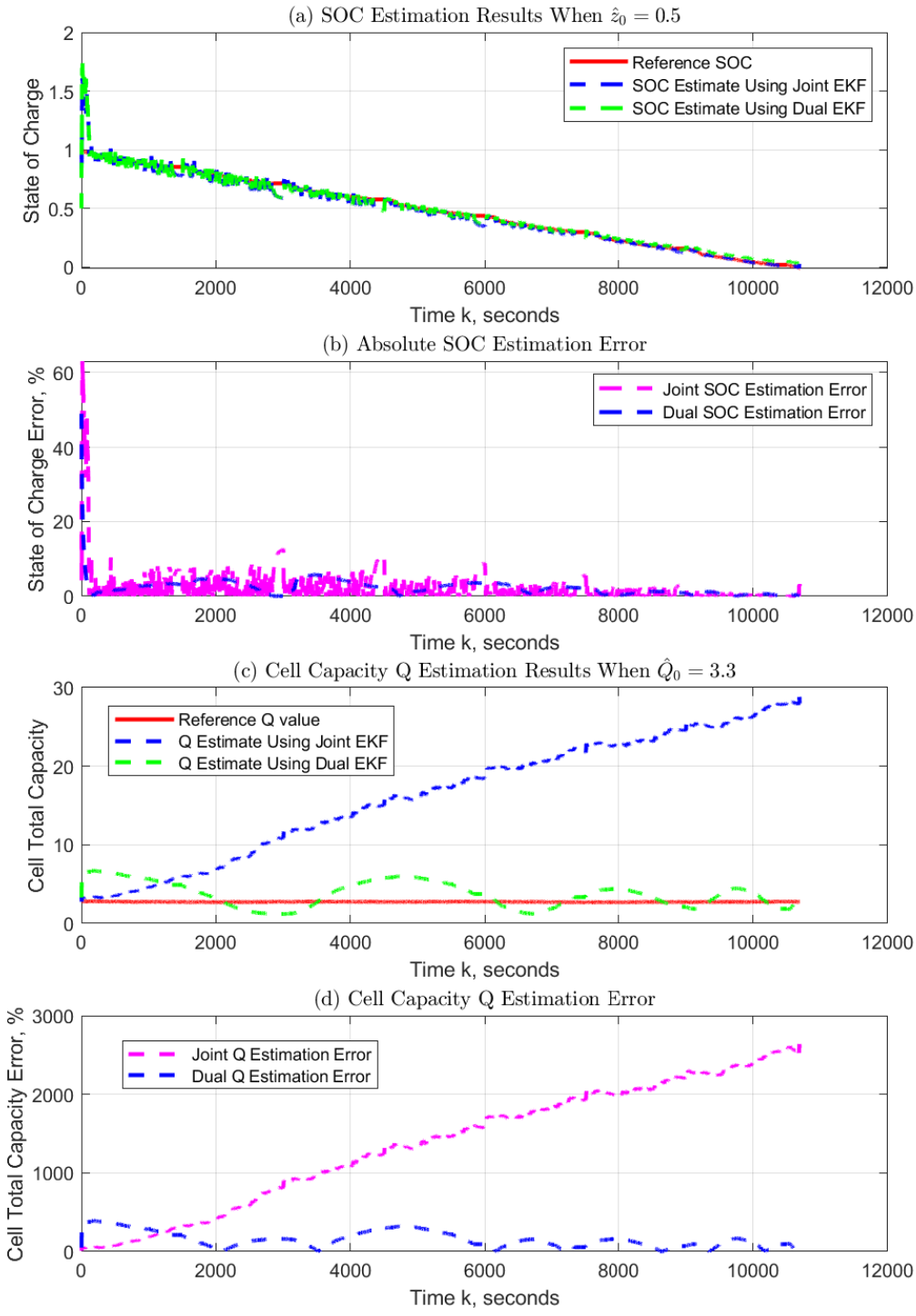

Figure 3.13: Estimation results at $\hat{z}_0 = 0.5$, $\hat{Q}_0 = 3.3$ (same initial set up compared to MMAE approach) using Joint and Dual estimation approach

Cell capacity estimation results are illustrated in Figures 3.12 (c) and 3.13 (c). In this simulation, the actual cell capacity is $2.8Ah$, while the nominal value remains at $3.3Ah$, indicating a cell capacity loss of $0.5Ah$ due to the aging effect.

To estimate the actual cell capacity using the MMAE technique, cell total capacity is quantized into a finite set: $Q_1 = 3.3Ah, Q_2 = 3.2Ah, \ldots, Q_i, \ldots, Q_N = 2.6Ah$ with decrements of $0.1Ah$. The set's upper and lower bounds are $3.3Ah$ and $2.6Ah$, respectively. As mentioned previously, typically, an aged cell is replaced when its actual capacity falls below 80% of its nominal value, which, in this case, corresponds to $2.64Ah$ [1]. Thus, the lower bound is set at $2.6Ah$.

Using (3.7), the conditional probability of each quantized cell capacity given all measurement data, denoted as $p(Q_i \mid Y_k)$, is computed. The value with the highest probability represents the most likely cell capacity estimate, $\hat{Q}$. In Figure 3.12 (c), the conditional probability $p(Q = 2.8 \mid Y_k)$ (yellow line) rapidly converges to one at time $k = 12s$, while the other conditional probabilities converge to zero.

Figure 3.13 (d) presents the weighted cell capacity estimate over time. This estimate is calculated as part of the data fusion technique.

$$\hat{Q}_k = \sum_{i=1}^{N} Q_i p(Q_i \mid Y_k) \tag{3.16}$$

Similar to (3.15), this shows a blended capacity estimation result that considers all quantized hypotheses in the set.

In contrast, results obtained using the Joint and Dual estimation techniques are presented in Figure 3.13 (c), where the initial cell capacity estimates for both techniques are set to $\hat{Q}_0 = 3.3$. Notably, the Dual estimation result (green line)

exhibits a slow convergence, while the Joint estimation technique (blue line) fails to converge altogether due to the model complexity. It is worth mentioning that cell models employed in works such as [89, 96, 97] are considerably simpler, as they do not consider factors like hysteresis voltage. Moreover, the initialization of both SOC and capacity estimates in these studies is much closer to the true initial values.

The absolute errors for both techniques are also calculated, as depicted in Figure 3.13 (d), providing further verification of the trends observed in Figure 3.13 (c).

3.4.4 Results of Some Special Cases

In this section, two special cases are numerically simulated to assess the robustness of the MMAE technique. The first case assumes that the actual cell capacity has a higher resolution compared to the quantization level of the cell capacity. In the second case, it is assumed that the actual cell capacity falls outside the defined quantization range. These scenarios provide valuable insights into the performance of the MMAE technique in challenging situations.

3.4.4.1 Special Case One

In the previous results employing the MMAE technique, it was assumed that the total capacity of an aged cell is $2.8Ah$, a value that falls within the predefined quantized set, as illustrated earlier. In Fig. 3.14, a simulation is conducted again, where the cell capacity is quantized as $Q_1 = 3.3Ah, Q_2 = 3.2Ah, \ldots, Q_i, \ldots, Q_N = 2.6Ah$ with decrements of $0.1Ah$, mirroring the previous quantization scheme. However, in this special case, the actual cell capacity value is set to $2.79Ah$. This means

that although the actual capacity still falls within the range of the set, it does not correspond to any of the quantized values in the predefined set.

Furthermore, to investigate the SOC estimation accuracy under different initial conditions, the actual initial SOC value is set to $z_0 = 0.5$, indicating that the cell is at 50% charge. Meanwhile, the initial SOC estimate is set at $\hat{z}_0 = 1$, indicating that the initial SOC is overestimated in this particular case.

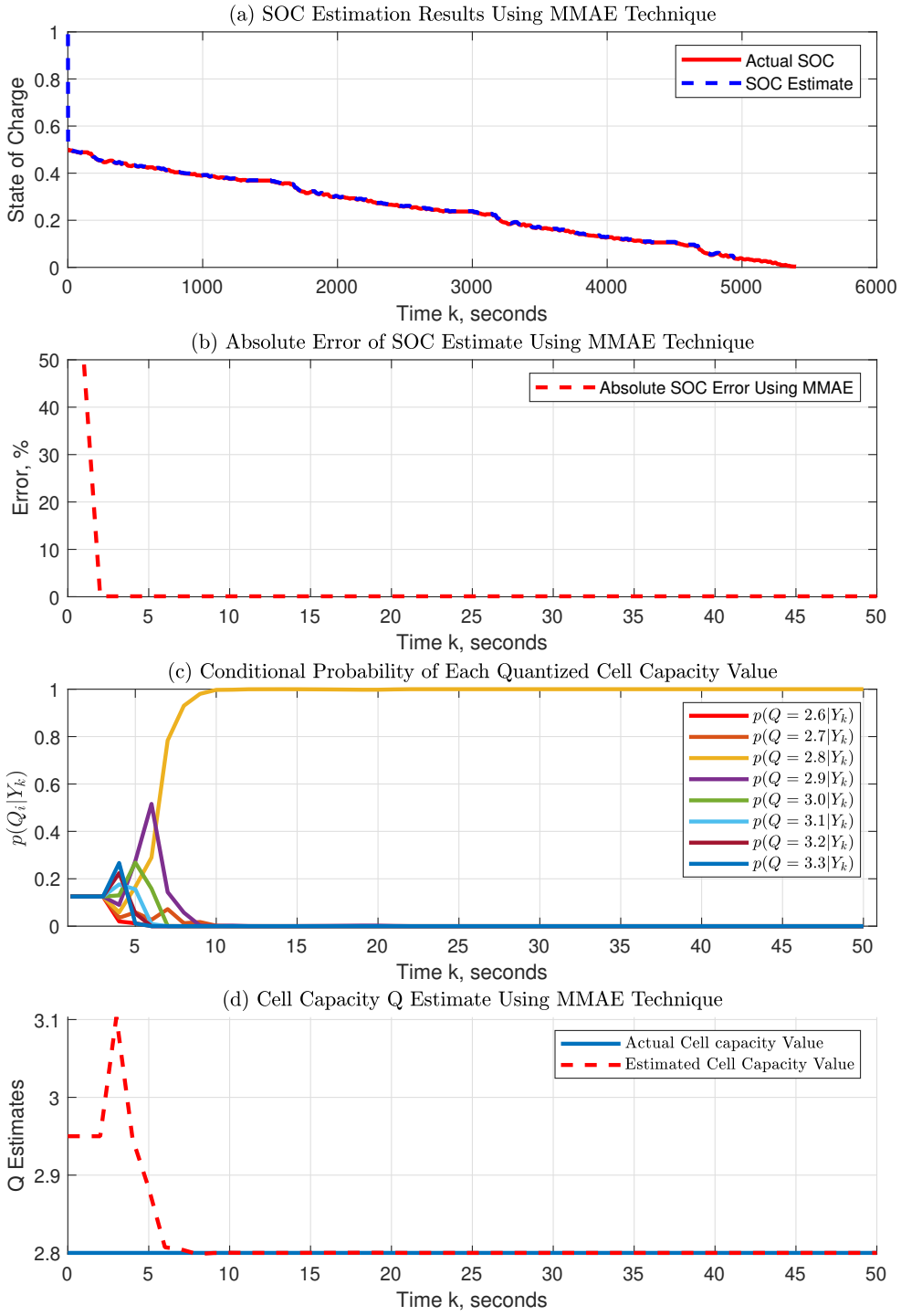

Figure 3.14: Estimation results at $\hat{z}_0 = 1$, Q quantized from 2.6 to 3.3 using MMAE technique

In Figure 3.14 (a), the SOC estimate $\hat{z}_k$ (blue line) continues to exhibit rapid convergence to the actual SOC (red line), and Figure 3.14 (b) illustrates the absolute error within the time span $0 \leq k \leq 50s$, with the error converging to zero at $k = 2s$.

With respect to the capacity estimation results depicted in Figure 3.14 (c), even though the actual capacity value of $2.79Ah$ is not among the quantized values listed in the set, the MMAE technique identifies the closest capacity value, yielding $\hat{Q} = 2.8Ah$ (yellow line) as the most likely capacity estimate. Figure 3.14 (d) shows the weighted cell capacity estimate $\hat{Q}_k$ (red line) over time, and it converges toward the closest quantized cell capacity value, which is $Q = 2.8Ah$.

To further validate this result, the simulation is replicated with different actual capacity values, ranging from $2.71Ah$ to $2.79Ah$ in increments of $0.01Ah$, as outlined in Table 3.4.

Table 3.4: Estimation results and convergence time for cell total capacities

Actual Capacity	Estimated Capacity	Convergence Time
$2.71Ah$	$2.7Ah$	$13s$
$2.72Ah$	$2.7Ah$	$11s$
$2.73Ah$	$2.7Ah$	$17s$
$2.74Ah$	$2.7Ah$	$28s$
$2.75Ah$	$2.8Ah$	$32s$
$2.76Ah$	$2.8Ah$	$25s$
$2.77Ah$	$2.8Ah$	$16s$
$2.78Ah$	$2.8Ah$	$11s$
$2.79Ah$	$2.8Ah$	$10s$

In Table 3.4, it can be observed that the estimated cell capacity consistently converges to the quantized capacity value that are most close to the actual capacity. Furthermore, it becomes apparent that the time required for convergence increases as the actual cell capacity deviates further from the quantized values included in the set. This elongation in the convergence time, with greater disparities between the actual capacity and the available quantized values, may serve as a valuable indicator of discrepancies between the set of quantized capacity values and the true capacity.

3.4.4.2 Special Case Two

In this special scenario, the set comprising quantized capacity values remains unchanged, but the actual capacity value is presumed to be smaller than the lower boundary ($2.6Ah$) of the set. Multiple simulations are conducted with various actual cell capacity values, and the outcomes are documented in Table 3.5. It is important to highlight that the initial conditions for SOC and its estimate are maintained at the same settings as in previous experiments.

Table 3.5: Estimation results when actual cell capacity value is out of the range

Actual Q	Q Estimate	Convergence Time	MAE of SOC
$2.0Ah$	$2.6Ah$	$13s$	2.19%
$2.1Ah$	$2.6Ah$	$11s$	2.76%
$2.2Ah$	$2.6Ah$	$10s$	1.97%
$2.3Ah$	$2.6Ah$	$13s$	1.04%
$2.4Ah$	$2.6Ah$	$11s$	0.79%
$2.5Ah$	$2.6Ah$	$12s$	0.11%

In Table 3.5, the actual cell capacity spans the range from $2.0Ah$ to $2.5Ah$, yet all the estimated cell capacity values uniformly converge to $2.6Ah$. Remarkably,

the MMAE technique consistently selects the value that is closest to the actual cell capacity value, maintaining a consistent convergence time. It is worth noting that the Absolute Error of SOC exhibits a decreasing trend as the actual capacity value approaches the lower boundary of the set.

This special scenario validates the earlier conclusion made in the previous case, wherein the MMAE technique consistently favors the quantized value that is nearest to the actual capacity as the most probable cell capacity estimate. This assertion holds significant relevance in the context of designing the range and quantization level for the cell capacity. For instance, if the estimated cell capacity converges to the lower boundary of the suggested set, then this lower boundary can be adjusted to encompass the actual cell capacity. Following this, the quantization level of the set can be systematically fine-tuned to progress from a coarse to a more precise estimate of cell capacity.

3.5 Conclusions and Future Work

In this chapter, challenges associated with online SOC estimation and the simultaneous estimation of SOC and total capacity for Lithium-ion cells are addressed. The MMAE technique is introduced and adapted as an effective solution. Specifically, the focus is on the estimation of the SOC of a LiFePO4 cell, and simulation results for a case study involving a LiFePO4 cell operating at 22°C with a 1C rate are presented.

The results demonstrate the capability of the MMAE technique to adaptively and accurately estimate SOC, even when the initial SOC estimate is poor. Moreover, the

SOC estimation converges rapidly to the actual SOC, outperforming the EKF results under the same initial set up.

Building upon this, an improved method that combines the strengths of both the EKF and MMAE techniques is proposed. The MMAE's rapid convergence property is leveraged to obtain a relatively accurate initial SOC estimate, which then initializes the EKF for continuous SOC estimation. This combined approach not only reduces computational costs compared to standalone MMAE, but also overcomes individual drawbacks. Simulation results on a LiFePO4 cell validate the effectiveness of this hybrid approach against EKF or MMAE alone.

Furthermore, the MMAE technique is extended to address the simultaneous estimation of SOC and cell total capacity for the LiFePO4 cell. The cell total capacity is discretized based on its nominal value, and a bank of EKFs is employed for conditional SOC estimation. The Bayes data fusion technique is then applied to obtain the most likely SOC and cell total capacity at each iteration. Simulation results showcase superior accuracy compared to dual and joint estimation techniques.

Looking ahead, future work will encompass an exploration of temperature-dependent SOC and simultaneous SOC/cell capacity estimation using the proposed techniques. Additionally, a more detailed investigation into Lithium-ion cell State-of-Power (SOP) estimation, cell pack SOC and SOH estimation, and cell balancing problems will be undertaken.

Chapter 4: Multi-Object Tracking Using Video Sequences with Improved Performance and Reduced Computation Cost

The third application discussed in this dissertation is the Multi-Object Tracking (MOT) problem in surveillance applications, defined as the problem of tracking multiple objects over time in video sequences. Estimation theory plays an important role for getting accurate MOT results. For example, sometimes the sensor measurement of the surveillance signal is intermittent. In this case, a robust estimation technique could still handle the missing data problem and bring a good tracking result. In this work, two efficient techniques are introduced in this study to enhance tracking accuracy and decrease computation complexity. Specifically, the tracking performance is improved with no additional computation cost by employing a more realistic motion model. In addition, a novel reduced-order Kalman filter is introduced, resulting in a reduction in computation cost while maintaining equivalent tracking accuracy compared to the full-order Kalman filter.

Experimental results obtained from standard MOT datasets [104] demonstrate that the proposed method yields competitive outcomes concerning both tracking accuracy and inference speed. Furthermore, the introduced techniques can be effortlessly integrated into other trackers to enhance their tracking performance.

4.1 Introduction

The significance of multi-object tracking in video sequences lies in its indispensable role in comprehending object interactions, estimating object trajectories, and predicting future object states. This capability unlocks diverse applications, including object counting [48], traffic monitoring [105], crowd analysis [106], and activity recognition [47]. Nevertheless, the MOT problem introduces considerable challenges stemming from inherent model inaccuracies and the unpredictable nature of measurement noise levels. These challenges often lead to imprecise motion predictions before tracks can be accurately associated with detections [13]. In addition, the computational cost of the tracker plays a pivotal role in shaping the inference speed, particularly within the tracking by detection framework. The impact of this computational cost on inference speed is noteworthy, underscoring the need for strategies to address and mitigate this challenge in the pursuit of more efficient and accurate MOT systems.

Nevertheless, the prevailing tracking algorithms inadequately address these challenges, particularly the enhancement of model accuracy and the reduction of the tracker's computation cost. Conversely, a multitude of existing methods place emphasis on the utilization of appearance features derived from deep neural networks and straightforward motion model predictions for track-detection association [58, 57, 107, 108]. While these appearance features hold the potential to enhance the accuracy of track-detection associations, they frequently encounter a formidable hurdle in the form of elevated computational costs induced by deep network-based feature

extractors. This, in turn, impedes their practical deployment in environments constrained by resources or systems operating under stringent time constraints [58, 109].

Moreover, the incorporation of appearance features into the tracker is not without its trade-offs. Notably, feature extractors necessitate additional rounds of training to acquire the object's appearance, a process that proves both time and computationally intensive. Therefore, the introduction of appearance features into the tracker is not akin to a "free lunch" and may result in diminished inference speed, ultimately constraining their efficacy in meeting real-time processing requirements, particularly in crowded scenarios.

In the realm of current tracking algorithms, the conventional practice involves the application of the full-order Kalman filter for the prediction and updating of object trajectories by modeling the moving objects using a simple Nearly Constant Velocity (NCV) model. The procedure entails allocating a Kalman filter to each new detection, facilitating the computation of the predicted trajectory during the prediction step. Subsequently, following the track-detection association step, the tracking ID can be appropriately assigned. It is noteworthy that the full-order state estimator can be streamlined, considering that part of the state (specifically, object location) is directly measured by the vision-based detector at each frame. In such instances, the implementation of a reduced-order state estimator becomes useful, leading to a reduction in computation costs. This streamlined approach holds the potential to enhance the inference speed analytically, particularly in scenarios characterized by dense crowds.

In addition, the majority of tracking by detection algorithms has consistently em-

ployed the simple NCV model for the prediction and updating of objects' trajectories [55, 56, 57, 107, 110]. However, the inherent limitations of the simple NCV model in capturing the complexity of motion dynamics become apparent. This model posits that an object's velocity is noise-driven, implying that the mean square error of the velocity is unbounded [13]. As a consequence, when sensor measurements for a given track are intermittent due to occlusion, noise, or detector performance, the assigned estimator can only update its trajectory estimate in the absence of supervisory input from sensor measurements. In such scenarios, the sole recourse for updating state estimates lies in relying on the motion model.

Nevertheless, the simple NCV model, coupled with noise as its input, fails to yield a bounded state estimate in the absence of sensor measurements. This deficiency results in inaccurate predictions and tracking errors when the object's sensor measurements are intermittent, highlighting a critical drawback of the model in handling scenarios with sporadic sensor input.

In this chapter, two improvements over the tracking-by-detection framework are made: firstly, it introduces an enhanced motion model to replace the simple NCV model, addressing the previously mentioned issues, and secondly, it presents a novel reduced-order Kalman filter. The improved motion model serves to rectify the problem of model inaccuracy, thereby enhancing overall tracking accuracy. Simultaneously, the introduction of the reduced-order Kalman filter aims to reduce computation costs.

Crucially, the enhancement in the motion model does not incur additional computation costs, making it a cost-effective solution. In addition, the proposed reduced-

order Kalman filter yields tracking results that are equivalent to those of the full-order Kalman filter but with a reduced computation cost. These methods stand out for their simplicity, effectiveness, and interpretability, making them easily implemented in any tracker without incurring additional computational expenses.

This chapter consists of five sections. Section 4.2 is the related work. Section 4.3 is the introduction of the proposed method. Section 4.4 consists experimental results, and Section 4.5 is the conclusion and future work.

4.2 Related Work

The MOT problem in video sequences has engendered numerous approaches, with two distinct types of solutions emerging as the most extensively researched. The first approach hinges on a motion-based methodology, wherein the detection outcomes from the object detector are utilized as sensor measurements. After that, a state estimator is assigned to each detection, followed by an association step to allocate tracking IDs. In this context, object tracking is conducted on a frame-by-frame basis.

The second approach is grounded in a purely vision-based paradigm, where deep neural networks are trained on given datasets to extract appearance features for the purpose of track-detection association. The output of the network furnishes the detection results with tracking IDs assigned directly, dispensing with the need for intermediate steps.

4.2.1 Motion Based Approach

The tracking-by-detection framework has found widespread application in MOT problems, particularly within the context of the motion-based approach. Among

those tracking-by-detection based MOT methods, SORT [107] is the most basic and efficient MOT technique. In SORT, the object's motion is characterized by a simple Nearly Constant Velocity (NCV) model to facilitate trajectory prediction. Following that, detections with a high confidence score originating from the vision-based detector are linked with the predicted trajectory as part of the track-detection association process. Finally, a state estimator is allocated to update the anticipated trajectory in accordance with the associated detection.

Expanding upon this foundational concept, several trackers have been devised to further enhance tracking performance. ByteTrack [55], for instance, takes all detections into consideration for association with tracks, thus augmenting tracking accuracy. This involves giving precedence to detections with higher confidence scores, followed by the association of detections with lower scores with any remaining unmatched tracks. In the case of OC-SORT [56], an Observation-centric Re-Update (ORU) technique is introduced to mitigate the cumulative prediction error that arises when measurement data is intermittent. SparseTrack [111] presents a Depth Cascading Matching (DCM) algorithm, which leverages depth information to enhance the accuracy of association.

4.2.2 Appearance Based Approach

The approach of track-detection association based on appearance holds a higher degree of intuitiveness when juxtaposed with the motion-based approach. In this technique, various features can be extracted from bounding boxes and subsequently applied for the purpose of association. Notably, deep features extracted from a pre-

trained network can significantly enhance association accuracy.

In DeepSORT [107], a fusion of visual-based features and motion features is utilized to yield improved association results. BoT-SORT [58] adopts a deep appearance extractor for its initial round of association. Deep OC-SORT [57] introduces an adaptive approach to update an object's appearance features, facilitating the integration of track-detection associations. In StrongSORT [109], the feature extractor is upgraded to obtain superior appearance features.

In certain other approaches such as those presented in [112, 113, 114, 115], entirely vision-based methodologies are proposed. These methods focus on learning deep representations from both appearance features and object trajectories. Nevertheless, their performances still exhibit limitations when compared to state-of-the-art tracking-by-detection methods, primarily owing to the higher computation costs involved and the paucity of available training datasets.

4.3 Proposed Method

4.3.1 Limitations on Previous Methods

Despite the claimed advancements in previous trackers, several enduring limitations persist within existing approaches, particularly in the following domains:

Lack of Analytical Interpretation: Numerous prior methods, particularly those rooted in vision-based systems, lack analytical interpretability. While deep learning based methods have achieved impressive performance, their opaque nature restricts the interpretability of outcomes. Understanding the rationale behind specific tracking decisions or diagnosing and rectifying tracking errors becomes challenging.

This absence of analytical interpretation impedes the capacity to glean insights into the tracking process, curbing the transparency and reliability of the proposed tracker.

Increased Computation Cost: An ongoing trend in enhancing tracking accuracy involves elevating the computational load of the algorithm, particularly in vision-based approaches, where deep neural networks are commonly trained to acquire appearance features for track-detection association. However, deep learning models typically demand substantial computational resources for both training and inference, culminating in augmented processing time and memory requisites. For instance, sophisticated learning models and data association techniques, such as TrackFormer [113] or EMA [116], can impose considerably higher computational costs, ultimately resulting in reduced tracking speed. While these approaches might yield improved accuracy, the increase in computational expense is not negligible. This poses challenges for real-time tracking applications, confining their practicality in resource-constrained environments or scenarios with stringent latency requirements.

Insufficient Focus on Motion Model and Kalman Filter Improvement: Despite the widespread use of Kalman filter based methods in the MOT problem, only a few prior approaches have emphasized enhancing the underlying models and the Kalman filter itself to bolster tracking accuracy. In BoT-SORT [58], the introduction of the camera motion compensation (CMC) method addresses motion model compensation when the camera is in motion. Similarly, OC-SORT [56] presents an improved Kalman filter to interpolate sensor measurements when detection results are intermittent. However, as mentioned earlier, these methods increase computational costs. Further research is warranted to explore the potential of advanced

models and improved variants of the Kalman filter that mitigate computational expenses.

4.3.2 Improved Motion Model

In response to the previously mentioned issue of model inaccuracy, a revised motion model is introduced with the aim of ameliorating tracking accuracy. This improved model, characterized by its simplicity and analytical interpretability, has been devised to mitigate the shortcomings associated with the existing motion model.

Consider a simple discrete-time NCV model for a moving object

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \end{bmatrix} = \begin{bmatrix} T & 0 & 1 & 0 \\ 0 & T & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} + \begin{bmatrix} \frac{T}{2} \\ \frac{T}{2} \\ 1 \\ 1 \end{bmatrix} v_k \tag{4.1}$$

$$\begin{bmatrix} x_k \\ y_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \end{bmatrix} \tag{4.2}$$

where T denotes the sampling time. Here, x_k and y_k represent the object's position in the x and y directions at time k, while $\dot{x}_k$ and $\dot{y}_k$ denote the object's velocity. The system input, denoted as v_k, is assumed to follow a normal distribution with $v_k \sim N(0, V)$.

The NCV model, in the context of bounding boxes, typically employs a selection

of system states, such as $x = [x_c, y_c, a, r, \dot{x}_c, \dot{y}_c, \dot{s}]^T$. In this representation, x_c and y_c signify the center coordinates of the bounding box, while $a = hw$ and $r = h/w$, where h and w correspond to the bounding box's height and width, respectively. Variations of the NCV model adopt different state representations for bounding boxes, including the "xywh" type, where $x = [x_c, y_c, h, w, \dot{x}_c, \dot{y}_c, \dot{h}, \dot{w}]^T$, and the "xyxy" type, where $x = [x_1, y_1, x_2, y_2, \dot{x}_1, \dot{y}_1, \dot{x}_2, \dot{y}_2]^T$, among others. These diverse state representations accommodate varying bounding box representations and have found applications across tracking methodologies.

As shown in (4.1), velocity of the object is modeled as

$$\dot{x}_{k+1} = \dot{x}_k + v_k \tag{4.3}$$

$$\dot{y}_{k+1} = \dot{y}_k + v_k \tag{4.4}$$

in x and y direction. The assumption made regarding the object's maneuvering implies that the speed (s) of the object adheres to an equation formulated as shown:

$$s_{k+1} = s_k + v_k \tag{4.5}$$

with $\mathbb{E}[v_k] = 0$, $\mathbb{E}[v_k^2] = \sigma_s^2$. If $v_k \sim N(0, V)$ and $s_0 = 0$, it can be easily derived that

$$\mathbb{E}[s_k^2] = k\sigma_s^2 \tag{4.6}$$

This implies that the mean square speed remains unbounded over time, as mentioned in [13]. Evidently, such an assumption is far from realistic, particularly when applied

to an NCV model, where a constant speed is posited. It would be more reasonable to assert that $\mathbb{E}[s_k^2] = S$, where S is a constant that remains independent of the variable k and is contingent solely upon the speed capabilities of the moving object.

Grounded in this premise, (4.5) can be improved, as indicated in [13]:

$$s_{k+1} = \sqrt{\frac{S^2 - \sigma_s^2}{S^2}} s_k + v_k \tag{4.7}$$

by doing this, the mean square speed can be computed as

$$\mathbb{E}[s_{k+1}^2] = \frac{S^2 - \sigma_s^2}{S^2} \mathbb{E}[s_k^2] + \sigma_s^2 \tag{4.8}$$

where

$$\mathbb{E}[s_k^2] = \mathbb{E}[s_{k+1}^2] = S^2 \tag{4.9}$$

and this implies

$$\dot{x}_{k+1} = \sqrt{\frac{S^2 - \sigma_s^2}{S^2}} \dot{x}_k + v_k \tag{4.10}$$

$$\dot{y}_{k+1} = \sqrt{\frac{S^2 - \sigma_s^2}{S^2}} \dot{y}_k + v_k \tag{4.11}$$

and this improvement result in a more realistic motion model

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \dot{x}_{k+1} \\ \dot{y}_{k+1} \end{bmatrix} = \begin{bmatrix} T & 0 & 1 & 0 \\ 0 & T & 0 & 1 \\ 0 & 0 & \sqrt{\frac{S^2-\sigma_s^2}{S^2}} & 0 \\ 0 & 0 & 0 & \sqrt{\frac{S^2-\sigma_s^2}{S^2}} \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix} + \begin{bmatrix} \frac{T}{2} \\ \frac{T}{2} \\ 1 \\ 1 \end{bmatrix} v_k \tag{4.12}$$

In practice, S serves as a tunable parameter, with its value contingent upon the speed capabilities inherent in the moving objects. Furthermore, the variance σ_s^2 associated with the input noise typically remains undisclosed or unknown. In this case, $\sqrt{\frac{S^2-\sigma_s^2}{S^2}}$ can be aptly regarded as a unified tuning parameter denoted by ϕ, where $0 \ll \phi < 1$.

Figure 4.1 illustrates a simulated trajectory of a moving object, featuring a complete trajectory represented by the blue line, while the red line corresponds to the detected trajectory. The presence of the red line denotes that the detected trajectory is intermittent, reflecting a practical scenario. This intermittence can arise due to multiple factors, including occlusions or potential mis-detections by the object detector, resulting in irregular measurements. In this case, to keep track of the object with intermittent measurements, the Kalman filter can only predict the state estimates using the given motion model when the sensor measurement is missing, and the state update step can only be executed when the state measurement is given.

Figure 4.2 further elaborates upon this by presenting two distinct estimated trajectories based on the intermittent measurements shown in Figure 4.1. The green line illustrates the estimated trajectory derived using the NCV model from (4.1),

while the red line portrays the estimated trajectory formulated with the improved motion model outlined in (4.12), characterized by the inclusion of the tuning parameter $\phi = 0.95$. A clear distinction emerges, with the enhanced motion model delivering notably more accurate estimated trajectories, particularly in situations where measurements are absent.

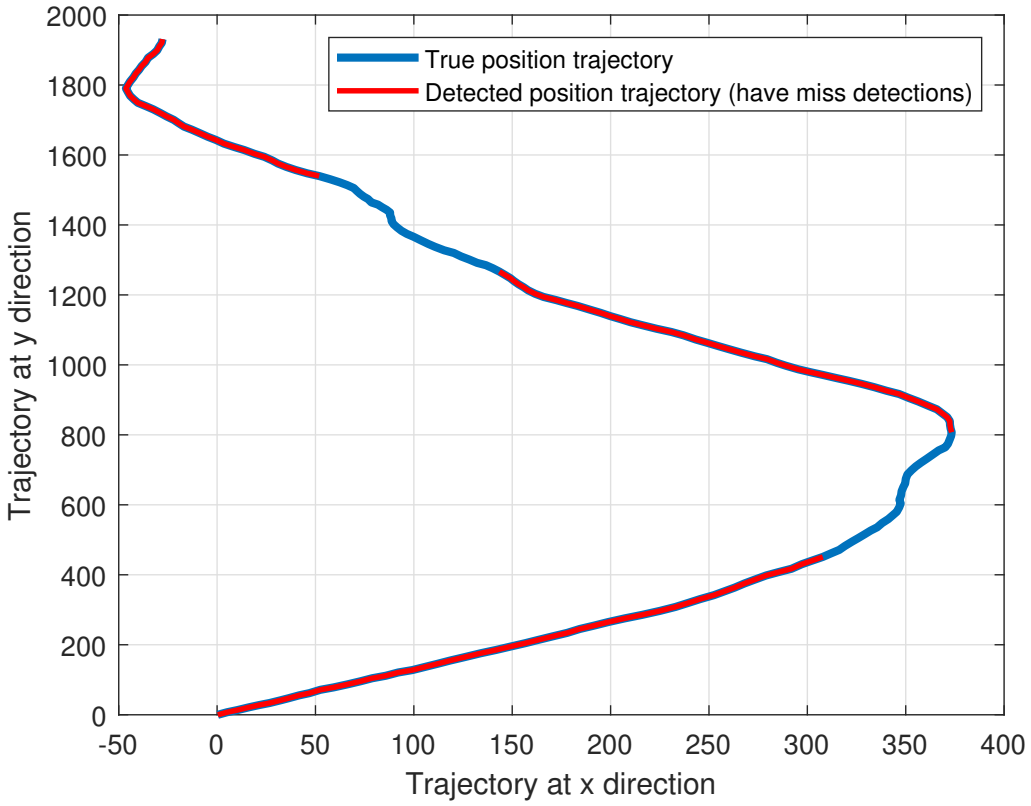

Figure 4.1: Simulated trajectory of a single moving object

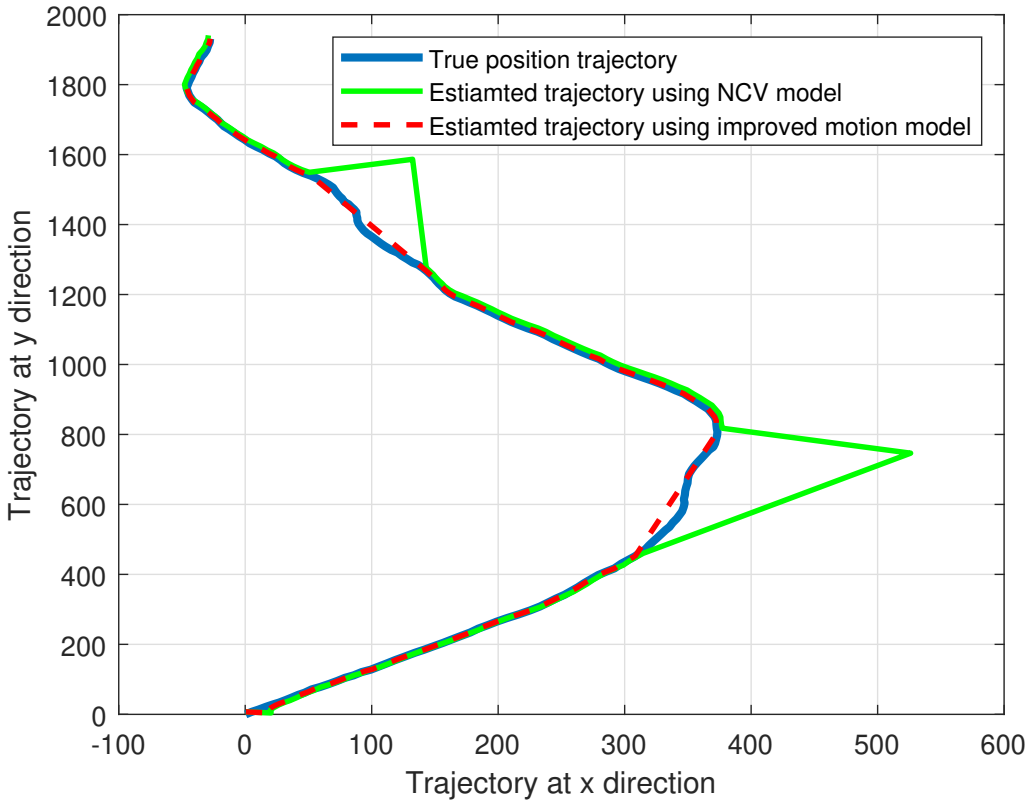

Figure 4.2: Estimation result of the simulated trajectory using the regular NCV and the improved motion model

In terms of the MOT problem where the "xyxy" or "xywh" type of NCV model is usually used, the improved "xyxy" or "xywh" type of motion model can be repre-

sented as

$$x_{k+1} = \begin{bmatrix} T & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & T & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & T & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & T & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \phi & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \phi & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \phi & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \phi \end{bmatrix} x_k + \begin{bmatrix} \frac{T}{2} \\ \frac{T}{2} \\ \frac{T}{2} \\ \frac{T}{2} \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} v_k \tag{4.13}$$

$$y_k = \begin{bmatrix} I_{4\times4} & 0_{4\times4} \end{bmatrix} x_k \tag{4.14}$$

where $x = [x_c, y_c, h, w, \dot{x}_c, \dot{y}_c, \dot{h}, \dot{w}]^T$ for "xywh" model, and $x = [x_1, y_1, x_2, y_2, \dot{x}_1, \dot{y}_1, \dot{x}_2, \dot{y}_2]^T$ for "xyxy" model. The tuning parameter $\phi = \sqrt{\frac{S^2 - \sigma_s^2}{S^2}}$ as shown before. Note this change does not increase the model complexity.

4.3.3 Reduced-order Kalman Filter

Within the track-by-detection framework, vision-based detectors play a pivotal role by furnishing the object's location to the motion model. This, in turn, facilitates the requisite updates for each assigned Kalman filter. Nevertheless, when deploying the aforementioned model for configuring the Kalman filter, the inclusion of large yet sparse matrices can give rise to heightened computation costs. These increased

computational demands can exert an adverse impact on the tracker's efficacy, particularly in scenarios characterized by congestion, where multiple Kalman filters are required to collaborate harmoniously.

Moreover, considering the motion model outlined earlier, it becomes evident that the utilization of the Kalman filter to estimate the position of each bounding box is superfluous, as it essentially duplicates the sensor measurements. In response to these multifaceted challenges and with the overarching objective of enhancing tracking speed, a more streamlined variant of the Kalman filter has been introduced. This refined variant is adept at curtailing computational overhead, thereby contributing to an overall improvement in tracking performance.

Numerous applications have explored the utilization of a reduced-order Kalman filter, aiming to accommodate limited computational resources. In [117, 118], the implementation of reduced-order filtering involves approximating the covariance through a lower-rank Singular Value Decomposition (SVD)-like decomposition, as elucidated in [13]. Additionally, a comprehensive methodology for the reduced-order Kalman filter is delineated in [13, 119]. Conversely, in works like [120, 121, 122], a simplified reduced-order Kalman filter is derived under the condition that the system states are all decoupled from each other.

However, these approaches either incur substantial computational expenses ([13, 119]), thereby conflicting with the primary intent of this undertaking, or prove inapplicable to the tracking problem ([120, 121, 122]), since the system states are not decoupled in MOT's motion model introduced before. To address these drawbacks, this section introduces a novel approach to the reduced-order Kalman filter tailored

specifically for the aforementioned improved motion model. This new approach is notably simpler when juxtaposed with [120, 121, 122]. Crucially, it circumvents the requirement for system states to be completely decoupled. Its hallmark lies in its simplicity, flexibility, and ease of implementation. In addition, the proposed reduced-order filter is not limited to the tracking problem but can be used for other applications as well.

4.3.3.1 Reduced-order Kalman Filter

Consider the following system [123]:

$$\begin{bmatrix} x_{1,k+1} \\ x_{2,k+1} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + \begin{bmatrix} v_{1,k} \\ v_{2,k} \end{bmatrix} \tag{4.15}$$

$$y_k = \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} \tag{4.16}$$

where the system is partitioned according to whether the state variables are (almost) perfectly measurable or not, and

$$v_{1,k} \sim N(0, V_1) \tag{4.17}$$

$$v_{2,k} \sim N(0, V_2) \tag{4.18}$$

equation (4.15) can be re-written as

$$x_{1,k+1} = A_{11}x_{1,k} + A_{12}x_{2,k} + v_{1,k} \tag{4.19}$$

$$x_{2,k+1} = A_{21}x_{1,k} + A_{22}x_{2,k} + v_{2,k} \tag{4.20}$$

Assuming that $x_{1,k+1}$ is also measurable (which will be relaxed later), the reduced order state estimate $\hat{x}_{2,k+1}$ can be written as

$$\begin{aligned} \hat{x}_{2,k+1} &= A_{21}x_{1,k} + A_{22}\hat{x}_{2,k} + K_k(x_{1,k+1} - A_{11}x_{1,k} - A_{12}\hat{x}_{2,k}) \\ &= (A_{22} - K_kA_{12})\hat{x}_{2,k} + (A_{21} - K_kA_{11})x_{1,k} + K_kx_{1,k+1} \end{aligned} \tag{4.21}$$

Therefore, the estimation error $e_{x_2,k+1}$ can be written as

$$\begin{aligned} e_{x_2,k+1} &= x_{2,k+1} - \hat{x}_{2,k+1} \\ &= A_{21}x_{1,k} + A_{22}x_{2,k} + v_{2,k} - (A_{22} - K_kA_{12})\hat{x}_{2,k} - \\ &\qquad (A_{21} - K_kA_{11})x_{1,k} - K_kx_{1,k+1} \\ &= K_kA_{11}x_{1,k} - K_kx_{1,k+1} + A_{22}x_2 + v_{2,k} - \\ &\qquad (A_{22} - K_kA_{12})\hat{x}_{2,k} \end{aligned} \tag{4.22}$$

Substituting for $x_{1,k+1}$, $e_{x_2,k+1}$ becomes

$$\begin{aligned} e_{x_2,k+1} &= K_k A_{11} x_{1,k} - K_k(A_{11}x_{1,k} + A_{12}x_{2,k} + v_{1,k}) + \\ &\qquad A_{22}x_2 + v_{2,k} - (A_{22} - K_k A_{12})\hat{x}_{2,k} \\ &= -K_k A_{12} x_{2,k} - K_k v_{1,k} + A_{22}x_2 + v_{2,k} - \\ &\qquad (A_{22} - K_k A_{12})\hat{x}_{2,k} \\ &= (A_{22} - K_k A_{12})x_{2,k} - (A_{22} - K_k A_{12})\hat{x}_{2,k} - \\ &\qquad K_k v_{1,k} + v_{2,k} \\ &= (A_{22} - K_k A_{12})e_{x_2,k} - K_k v_{1,k} + v_{2,k} \end{aligned} \tag{4.23}$$

The error covariance of x_2's estimation is found using the Riccati equation

$$P_{k+1} = (A_{22} - K_k A_{12})P_k(A_{22} - K_k A_{12})^T + V_2 + K_k V_1 K_k^T \tag{4.24}$$

and the Kalman gain is

$$K_k = A_{22}P_k A_{12}^T(A_{12}P_k A_{12}^T + V_1)^{-1} \tag{4.25}$$

Practical consideration: since $x_{1,k+1}$ is unknown at time k, instead of solving $\hat{x}_{2,k}$, $\hat{\mathscr{X}_k} = \hat{x}_{2,k} - K_{k-1}x_{1,k}$ is introduced, note $y_k = x_{1,k}$ from equation (4.16). In this case

$$\begin{aligned}
\hat{\mathscr{X}_{k+1}} &= \hat{x}_{2,k+1} - K_{k-1}x_{1,k+1} \\
&= (A_{22} - K_{k-1}A_{12})\hat{x}_{2,k} + (A_{21} - K_{k-1}A_{11})y_k + \\
&\qquad\qquad K_{k-1}x_{1,k+1} - K_{k-1}x_{1,k+1} \\
&= (A_{22} - K_{k-1}A_{12})\hat{x}_{2,k} + (A_{21} - K_{k-1}A_{11})y_k \\
&= (A_{22} - K_{k-1}A_{12})\hat{\mathscr{X}_k} + \\
&\quad [(A_{22} - K_{k-1}A_{12})K_{k-1} + A_{21} - K_{k-1}A_{11}]y_k
\end{aligned} \tag{4.26}$$

after solving $\hat{\mathscr{X}_k}$, $\hat{x}_{2,k}$ can be found from

$$\hat{x}_{2,k} = \hat{\mathscr{X}_k} + K_{k-1}y_k \tag{4.27}$$

Note $K_{k-1} = K_{-1}$ when $k = 0$, and it can be defined as $K_{-1} = 0$ for initialization. The estimation error covariance of $\hat{x}_{2,k}$ and $\hat{\mathscr{X}_k}$ are the same.

In summary, the reduced-order Kalman filter is shown as below:

Algorithm 7 One-step Reduced-order Kalman filter

1: **Initialization**: Initialize the augmented state estimate $\hat{\mathscr{X}_0} = \mathbb{E}[x_{2,0}]$ and error covariance matrix $P_0 = cov[x_{2,0}]$, define Kalman gain $K_{-1} = 0$ when $k = 0$

2: **for** $k = 0, 1, 2, \ldots$ **do**

3: $\quad \hat{x}_{2,k} = \hat{\mathscr{X}_k} + K_{k-1} y_k$

4: $\quad \hat{\mathscr{X}_{k+1}} = (A_{22} - K_{k-1}A_{12})\mathscr{X}_k + [(A_{22} - K_{k-1}A_{12})K_{k-1} + A_{21} - K_{k-1}A_{11}]y_k$

5: $\quad P_{k+1} = (A_{22} - K_{k-1}A_{12})P_k(A_{22} - K_{k-1}A_{12})^T + V_2 + K_{k-1}V_1K_{k-1}^T$

6: $\quad K_k = A_{22}P_kA_{12}^T(A_{12}P_kA_{12}^T + V_1)^{-1}$

7: **end for**

Since the tracking-by-detection framework requires the track-detection association between the predict and update step, the two-step version of the reduced-order Kalman filter is also introduced:

Algorithm 8 Two-step Reduced-order Kalman filter

1: **Initialization**: Initialize the augmented state estimate $\hat{\mathscr{X}_0^+} = \mathbb{E}[x_{2,0}]$ and error covariance matrix $P_0^+ = cov[x_{2,0}]$, define Kalman gain $K_{-1} = 0$ when $k = 0$

2: **for** $k = 0, 1, 2, \ldots$ **do**

3: $\quad \hat{\mathscr{X}_{k+1}^-} = A_{22}\hat{\mathscr{X}_k^+}$

4: $\quad P_{k+1}^- = A_{22}P_k^+A_{22}^T + V_2$

5: $\quad P_{k+1}^+ = P_k^- - P_k^-A_{12}^T(A_{12}P_k^-A_{12}^T + V_1)A_{12}P_k^-$

6: $\quad \hat{\mathscr{X}_{k+1}^+} = (I - K_{k-1}A_{12})\hat{\mathscr{X}_k^-} + [(I - K_{k-1}A_{12})A_{22}K_{k-1} + A_{21} - K_{k-1}A_{11}]y_k$

7: $\quad \hat{x}_{2,k}^- = \hat{\mathscr{X}_k^-} + A_{22}K_{k-1}y_k$

8: $\quad K_k = P_k^-A_{12}^T(A_{12}P_k^-A_{12}^T + V_1)^{-1}$

9: **end for**

In terms of the motion model shown in (4.13), parameter matrices are identified as $A_{11} = TI_{4\times4}$, $A_{12} = I_{4\times4}$, $A_{21} = 0_{4\times4}$, $A_{22} = \phi I_{4\times4}$, and the reduced-order Kalman filter can be set up accordingly for this application.

4.3.3.2 Steady-state Reduced-order Kalman Filter

In pursuit of further diminishing computational overhead, the option of implementing the steady-state solution becomes particularly pertinent, especially when computational resources are constrained. In this context, rather than iterating the time-variant Kalman gain, the constant Kalman gain can be derived through the offline resolution of the algebraic Riccati equation.

To expound further, for the motion model as delineated in equation (4.13), there are $A_{11} = TI_{4\times4} \stackrel{\text{def}}{=} D_1$, $A_{12} = I_{4\times4}$, $A_{21} = 0_{4\times4}$, and $A_{22} = \phi I_{4\times4} \stackrel{\text{def}}{=} D_2$. The algebraic Riccati equation, within this context, can be formulated as follows:

$$P = D_2 P D_2^T - D_2 P (P + V_1)^{-1} P D_2^T + V_2 \tag{4.28}$$

The constant Kalman gain is

$$K = D_2 P (P + V_1)^{-1} \tag{4.29}$$

After finding the constant Kalman gain K, the state estimation equation is

$$\hat{\mathscr{X}}_{k+1} = (D_2 - K)\hat{\mathscr{X}}_k + [(D_2 - K)K - KD_1]y_k \tag{4.30}$$

Finally, $\hat{x}_{2,k}$ can be found from

$$\hat{x}_{2,k} = \hat{\mathscr{X}}_k + Ky_k \tag{4.31}$$

The two-step steady-state filter is summarized as shown in Algorithm 3.

Algorithm 9 Two-step Steady-state Reduced-order Kalman filter

1: **Initialization**: Initialize the augmented state estimate $\hat{\mathscr{X}_0^+} = \mathbb{E}[x_{2,0}]$

2: Compute the Algebric Riccati equation for error covariance matrix P^- using:

3: $P^- = A_{22}P^+A_{22}^T + V_2$

4: $P^+ = P^- - P^-A_{12}^T(A_{12}P^-A_{12}^T + V_1)A_{12}P^-$

5: Compute the constant Kalman Gain K:

6: $K = P^-A_{12}^T(A_{12}P^-A_{12}^T + V_1)^{-1}$

7: **for** $k = 0, 1, 2, \ldots$ **do**

8: $\quad \hat{\mathscr{X}_{k+1}^-} = A_{22}\hat{\mathscr{X}_k^+}$

9: $\quad \hat{\mathscr{X}_{k+1}^+} = (I - KA_{12})\hat{\mathscr{X}_k^-} + [(I - KA_{12})A_{22}K + A_{21} - KA_{11}]y_k$

10: $\quad \hat{x}_{2,k}^- = \hat{\mathscr{X}_k^-} + A_{22}Ky_k$

11: **end for**

Note $(I-KA_{12})$ and $[(I-KA_{12})A_{22}K+A_{21}-KA_{11}]$ in step 9 can be pre-computed offline as well to further reduce the computation cost.

4.4 Simulation Results

4.4.1 Experimental Setup

In this work, the SORT [46] with the "xyxy" type motion model is employed as the baseline for assessing the proposed techniques. To ensure consistency in the simulation results, the YOLOX detector [124], is directly utilized without any supplementary modifications. All experimental evaluations, conducted on hardware con-

figurations [1], are grounded in identical detection results. Consequently, the reported tracking performance is indicative of the off-the-shelf tracking capabilities. It is pertinent to note that any lost tracklets are retained for a duration of 30 frames, allowing for the possibility of their re-appearance within the tracking process.

4.4.2 Ablation Results

The ablation results pertaining to the MOT17 dataset are presented in Table 4.1. An observation of note is the increase in IDF1 [125] by 0.3% when the improved model (IM) is applied, with the parameter ϕ fine-tuned to a value of 0.97. Furthermore, following the implementation of the reduced-order Kalman filter (ROKF), the frames per second (FPS) demonstrates a noteworthy increase of 11 compared to the full order Kalman filter. It is imperative to highlight that the introduction of the improved model does not entail any additional computational cost, thereby resulting in no degradation in the FPS.

[1]Hardware: Intel(R) Core(TM) i9-10850k @ 3.6GHz and NVIDIA GeForce RTX 3090 GPU.

Table 4.1: Ablation results on MOT17 validation set. Note the off-the-shelf FPS is reported. The best results are shown in **bold**.

IM	ROKF	IDF1↑	MOTA↑	IDs↓	FPS↑
		79.1	78.7	**260**	624
✓		**79.4**	**78.8**	272	624
	✓	79	78.7	**260**	**635**
✓	✓	**79.4**	**78.8**	271	**635**

Results on the MOT17 test set are shown in Table 4.2. The performance of the tracker, especially IDF1 [125], is still competitive compared to most of the previous work.

Table 4.2: Comparison of the state-of-the-art methods under the "private detector" protocol on MOT17 test set. The best results are shown in **bold**. Note the proposed tracker is based on SORT and only includes the use of improved model and reduced-order Kalman filter.

Tracker	MOTA↑	IDF1↑	HOTA↑	FP↓	FN↓	IDs↓	FPS↑
DAN [126]	52.4	49.5	39.3	25423	234592	8431	<3.9
Tube_TK [127]	63.0	58.6	48.0	27060	177483	4137	3.0
MOTR [128]	65.1	66.4	-	45486	149307	2049	-
CTracker [129]	66.6	57.4	49.0	22284	160491	5529	6.8
CenterTrack [114]	67.8	64.7	52.2	**18498**	160332	3039	17.5
QuasiDense [115]	68.7	66.3	53.9	26589	146643	3378	20.3
TraDes [130]	69.1	63.9	52.7	20892	150060	3555	17.5
MAT [131]	69.5	63.1	53.8	30660	138741	2844	9.0
SOTMOT [132]	71.0	71.9	-	39537	118983	5184	16.0
TransCenter [133]	73.2	62.2	54.5	23112	123738	4614	1.0
GSDT [134]	73.2	66.5	55.2	26397	120666	3891	4.9
Semi-TCL [135]	73.3	73.2	59.8	22944	124980	2790	-
FairMOT [136]	73.7	72.3	59.3	27507	117477	3303	25.9
RelationTrack [137]	73.8	74.7	61.0	27999	118623	**1374**	8.5
PermaTrackPr [138]	73.8	68.9	55.5	28998	115104	3699	11.9
CSTrack [139]	74.9	72.6	59.3	23847	114303	3567	15.8
TransTrack [140]	75.2	63.5	54.1	50157	86442	3603	10.0
FUFET [141]	76.2	68.0	57.9	32796	98475	3237	6.8
SiamMOT [142]	76.3	72.3	-	-	-	-	12.8
CorrTracker [143]	76.5	73.6	60.7	29808	99510	3369	15.6
TransMOT [144]	76.7	75.1	61.7	36231	93150	2346	9.6
ReMOT [145]	77.0	72.0	59.7	33204	93612	2853	1.8
ByteTrack [55]	**80.3**	77.3	**63.1**	25491	**83721**	2196	29.6
Ours	76.4	**77.4**	56.7	32660	110124	2567	**589** [2]

To further investigate effectiveness of the proposed methods, the experiment is repeated on the MOT17 validation set with ByteTrack as the base tracker, and results are shown in Table 4.3.

Table 4.3: Ablation results on MOT17-val based on ByteTrack.

IM	ROKF	IDF1↑	MOTA↑	IDs↓	FPS↑
		80.2	78.3	**235**	335
✓		**80.4**	**78.8**	272	335
	✓	80.2	78	256	**342**
✓	✓	**80.4**	**78.8**	236	**342**

As depicted in Table 4.3, the ablation results conspicuously underscore the efficacy of the proposed methodologies in enhancing the tracking performance of ByteTrack. The introduction of the novel reduced-order Kalman filter and the improved motion model is discernibly associated with improvements in tracking accuracy and FPS. These observations accentuate the practical advantages that the proposed techniques confer upon real-world MOT applications, endowing camera-based surveillance systems with enhanced robustness and efficiency.

[2]Off-the-shelf FPS for the proposed tracker is reported here

It is worth noting that the off-the-shelf FPS is noticeably lower in comparison to the proposed tracker, as presented in Table 4.2. This discrepancy arises from the additional complexity introduced by the second round of track-detection association.

4.5 Conclusions and Future Works

In this chapter, an improved motion model and a novel reduced-order Kalman filter are introduced with the overarching goal of enhancing tracking accuracy while concurrently mitigating the computational burden. The experimental results presented herein manifest the tangible benefits of these innovations: the improved model has been observed to bring about a marked enhancement in tracking performance, all without the imposition of any supplementary computational costs. Simultaneously, the integration of the reduced-order Kalman filter has led to a discernible reduction in computational expenses in comparison to its full-order counterpart. Notably, the steady-state version of the reduced-order Kalman filter is also introduced which can further reduce the computation cost.

Furthermore, the inherent flexibility of the proposed techniques lends itself to seamless integration into a spectrum of other tracking frameworks. The adaptability of these innovations serves as a testament to their versatility, facilitating their incorporation into various tracking systems and applications.

Future work includes the integration of the proposed technique into other applications. One notable trajectory of future research involves the utilization of the proposed methods in the context of Multi-Camera Multi-Object tracking. This is a realm in which the efficacy of these techniques can be further evaluated, with a

particular emphasis on their applicability in multi-camera setups.

In addition, the application of these methodologies in resource-constrained scenarios, such as edge processing, offers a compelling avenue for further investigation. In such environments, the computational resources are limited, and the ability to maintain robust tracking performance while operating within these constraints is of paramount importance. Therefore, future endeavors will be geared towards exploring the applicability and effectiveness of the proposed methods in such resource-constrained settings.

Chapter 5: Conclusions and Summary

This dissertation addresses three applications across different domains using the proposed state estimation techniques. The first application involves the intrusion detection problem for sensors and actuators, employing the adapted MMAE technique. Specifically, the objective is to identify potential cyber attacks on industrial control systems. To achieve this, an intrusion model is initially proposed and then implemented into the MMAE technique to detect various types of attack signals.

The second application focuses on SOC and total cell capacity estimation for Lithium-ion cells. In this application, three techniques are devised to estimate SOC individually and SOC/cell total capacity simultaneously.

The third application seeks to reduce computation costs and enhance model accuracy in the multi-object tracking problem. An improved motion model is utilized to achieve higher tracking accuracy, particularly in scenarios with intermittent sensor measurements. Additionally, a reduced-order Kalman filter is designed to analytically minimize computation costs. A steady-state reduced-order Kalman filter is also introduced to further decrease computation costs. The versatility of the proposed technique extends beyond these applications, allowing adaptation to other problems to achieve computational cost reduction.

5.1 Sensor and Actuator Intrusion Detection Problem

In terms of the CPS intrusion detection problem under sensor, actuator, or dual attacks, the impact of various intrusion signals on system performance, specifically on false data injection attack with step and ramp type attack signals is analyzed. After that, strategies for detecting these intrusions are developed, and the MMAE technique is adapted for effective detection.

The approach involves considering the generalized healthy CPS model and models under different false data injection attacks. Simulation results, using a DC motor CPS model, demonstrate the technique's efficacy in detecting unknown intrusion signals and identifying their nature (sensor, actuator, or both). Addressing detection time delays, the exponential data weighting technique is applied to the Riccati equation.

Future work will focus on scalability and efficiency for large-scale CPS networks. Developing strategies for automated or semi-automated responses to detected intrusions which could further enhance CPS security.

5.2 Online SOC and cell total capacity estimation problem

In terms of the battery management system cell staus estimation problem, the MMAE technique is introduced as an effective solution for LiFePO4 cell SOC estimation with simulations at 22°C and a 1C rate.

Results demonstrate MMAE's adaptive and accurate SOC estimation, even with a poor initial estimate, outperforming the EKF in convergence speed. Folloiwng

this, an improved method combining the EKF and the MMAE is proposed, leveraging MMAE's rapid convergence for an accurate initial SOC estimate. This hybrid approach reduces computational costs and surpasses individual techniques, as validated in LiFePO4 cell simulations.

The MMAE technique is further extended for simultaneous SOC and cell total capacity estimation. Specifically, cell total capacity is first discretized, and a bank of EKFs is then set up for conditional SOC estimation, followed by Bayes data fusion. This technique results in superior accuracy compared to dual and joint estimation techniques.

Future work includes exploring temperature-dependent SOC, simultaneous SOC/-cell capacity estimation, and detailed investigations into Lithium-ion cell SOP estimation, cell pack SOC and SOH estimation, and cell balancing problems.

5.3 Multi-Object Tracking Problem

In terms of the Multi-object tracking problem proposed in chapter 4, it introduces an improved motion model and a novel reduced-order Kalman filter to enhance tracking accuracy while mitigating computational burdens. Experimental results demonstrate the enhanced tracking performance without additional computational costs. The integration of the reduced-order Kalman filter reduces computational expenses, and the steady-state version further reduces costs.

The proposed techniques are adaptable and can seamlessly integrate into various tracking frameworks, showcasing their versatility for different systems and applications.

Future work involves integrating the technique into other applications, particularly exploring Multi-Camera Multi-Object tracking. In addition, the extension of the proposed reduced-order Kalman filter will also be included [146, 147]. Evaluating the effectiveness of these techniques in resource-constrained scenarios, like edge processing, is a promising avenue for further investigation, considering the importance of maintaining robust tracking performance within limited computational resources.

Bibliography

[1] G. L. Plett, *Battery management systems, Volume II: Equivalent-circuit methods*. Artech House, 2015.

[2] J. Su, S. Schneider, E. Yaz, and F. Josse, "Online state of charge estimation of lithium-ion battery cells: A multiple model adaptive estimation approach," in *2021 American Control Conference (ACC)*, 2021, pp. 4447–4452.

[3] ——, "Simultaneous state of charge and total capacity estimation of lithium-ion cells using multiple model adaptive estimation," in *2022 IEEE Conference on Control Technology and Applications (CCTA)*, 2022, pp. 778–784.

[4] S. Skaff, A. A. Rizzi, H. Choset, and M. Tesch, "Context identification for efficient multiple-model state estimation of systems with cyclical intermittent dynamics," *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 14–28, 2011.

[5] J. Sun, J. Song, H. Chen, X. Huang, and Y. Liu, "Autonomous state estimation and mapping in unknown environments with onboard stereo camera for micro aerial vehicles," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 5746–5756, 2020.

[6] Y. Y. Lim, S. L. Kek, and K. L. Teo, "Efficient state estimation strategies for stochastic optimal control of financial risk problems," *Data Science in Finance and Economics*, vol. 2, no. 4, pp. 356–370, 2022.

[7] E. Beadle and P. Djuric, "A fast-weighted bayesian bootstrap filter for nonlinear model state estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 1, pp. 338–343, 1997.

[8] J. Su, Y. Weng, S. C. Schneider, and E. E. Yaz, "Sensor and actuator intrusion detection for cyber-physical systems via adaptive estimation algorithm," in *Dynamic Systems and Control Conference*, vol. 84287. American Society of Mechanical Engineers, 2020, p. V002T33A002.

[9] J. Su, Y. Weng, S. Schneider, and E. Yaz, "Accelerated detection method for sensor and actuator intrusions in cyber-physical systems using multiple model estimation algorithm," pp. 1–8, 2020.

[10] J. Su, "Sensor intrusion detection in control systems using estimation theory," Master's thesis, Marquette University, 2019.

[11] J. Su, A. Strandt, S. Schneider, E. Yaz, and F. Josse, "Improved state of charge estimation of lithium-ion battery cells," in *2021 European Control Conference (ECC)*, 2021, pp. 1645–1650.

[12] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[13] B. D. Anderson and J. B. Moore, *Optimal filtering*. Courier Corporation, 2012.

[14] G. C. Goodwin and K. S. Sin, *Adaptive filtering prediction and control*. Courier Corporation, 2014.

[15] G. Ellis, *Observers in control systems: a practical guide*. Elsevier, 2002.

[16] R. J. Vaccaro, "Digital control: A state space approach," *IEEE Control Systems*, vol. 16, no. 1, p. 90, 1996.

[17] E. A. Wan and R. Van Der Merwe, "The unscented kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*. Ieee, 2000, pp. 153–158.

[18] C. Yang, R. Duraiswami, and L. Davis, "Fast multiple object tracking via a hierarchical particle filter," in *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, vol. 1. IEEE, 2005, pp. 212–219.

[19] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 425–437, 2002.

[20] H. F. Lopes and R. S. Tsay, "Particle filters and bayesian inference in financial econometrics," *Journal of Forecasting*, vol. 30, no. 1, pp. 168–209, 2011.

[21] S. Thrun, "Particle filters in robotics." in *UAI*, vol. 2. Citeseer, 2002, pp. 511–518.

[22] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerospace and Electronic Systems Magazine*, vol. 25, no. 7, pp. 53–82, 2010.

[23] P. M. Djuric, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Miguez, "Particle filtering," *IEEE signal processing magazine*, vol. 20, no. 5, pp. 19–38, 2003.

[24] B. F. Farrell and P. J. Ioannou, "State estimation using a reduced-order kalman filter," *Journal of the Atmospheric Sciences*, vol. 58, no. 23, pp. 3666–3680, 2001.

[25] C. A. Greenhall, "A review of reduced kalman filters for clock ensembles," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 59, no. 3, pp. 491–496, 2012.

[26] Z.-L. Deng, Y. Gao, L. Mao, Y. Li, and G. Hao, "New approach to information fusion steady-state kalman filtering," *Automatica*, vol. 41, no. 10, pp. 1695–1707, 2005.

[27] P. D. Hanlon and P. S. Maybeck, "Multiple-model adaptive estimation using a residual correlation kalman filter bank," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 2, pp. 393–406, 2000.

[28] X. Liu, X. Liu, W. Zhang, and Y. Yang, "Interacting multiple model uav navigation algorithm based on a robust cubature kalman filter," *IEEE Access*, vol. 8, pp. 81 034–81 044, 2020.

[29] M. Kheirandish, E. A. Yazdi, H. Mohammadi, and M. Mohammadi, "A fault-tolerant sensor fusion in mobile robots using multiple model kalman filters," *Robotics and Autonomous Systems*, vol. 161, p. 104343, 2023.

[30] G. Terejanu, P. Singla, T. Singh, and P. D. Scott, "Adaptive gaussian sum filter for nonlinear bayesian estimation," *IEEE Transactions on Automatic Control*, vol. 56, no. 9, pp. 2151–2156, 2011.

[31] E. A. Lee, "Cyber physical systems: Design challenges," in *2008 11th IEEE international symposium on object and component-oriented real-time distributed computing (ISORC)*. IEEE, 2008, pp. 363–369.

[32] R. Alguliyev, Y. Imamverdiyev, and L. Sukhostat, "Cyber-physical systems and their security issues," *Computers in Industry*, vol. 100, pp. 212–223, 2018.

[33] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE transactions on automatic control*, vol. 58, no. 11, pp. 2715–2729, 2013.

[34] R. Mitchell and I.-R. Chen, "A survey of intrusion detection techniques for cyber-physical systems," *ACM Computing Surveys (CSUR)*, vol. 46, no. 4, pp. 1–29, 2014.

[35] J.-P. A. Yaacoub, O. Salman, H. N. Noura, N. Kaaniche, A. Chehab, and M. Malli, "Cyber-physical systems security: Limitations, issues and future trends," *Microprocessors and microsystems*, vol. 77, p. 103201, 2020.

[36] A. Yeboah-Ofori, J. Abdulai, and F. Katsriku, "Cybercrime and risks for cyber physical systems," *International Journal of Cyber-Security and Digital Forensics (IJCSDF)*, vol. 8, no. 1, pp. 43–57, 2019.

[37] J. Wurm, Y. Jin, Y. Liu, S. Hu, K. Heffner, F. Rahman, and M. Tehranipoor, "Introduction to cyber-physical system security: A cross-layer perspective," *IEEE Transactions on Multi-Scale Computing Systems*, vol. 3, no. 3, pp. 215–227, 2016.

[38] J. Giraldo, E. Sarkar, A. A. Cardenas, M. Maniatakos, and M. Kantarcioglu, "Security and privacy in cyber-physical systems: A survey of surveys," *IEEE Design & Test*, vol. 34, no. 4, pp. 7–17, 2017.

[39] L. Cao, X. Jiang, Y. Zhao, S. Wang, D. You, and X. Xu, "A survey of network attacks on cyber-physical systems," *IEEE Access*, vol. 8, pp. 44 219–44 227, 2020.

[40] P. M. Lima, M. V. S. Alves, L. K. Carvalho, and M. V. Moreira, "Security against communication network attacks of cyber-physical systems," *Journal of Control, Automation and Electrical Systems*, vol. 30, pp. 125–135, 2019.

[41] Y. Wang, J. Tian, Z. Sun, L. Wang, R. Xu, M. Li, and Z. Chen, "A comprehensive review of battery modeling and state estimation approaches for advanced battery management systems," *Renewable and Sustainable Energy Reviews*, vol. 131, p. 110015, 2020.

[42] M. Shen and Q. Gao, "A review on battery management system from the modeling efforts to its multiapplication and integration," *International Journal of Energy Research*, vol. 43, no. 10, pp. 5042–5075, 2019.

[43] K. Liu, K. Li, Q. Peng, and C. Zhang, "A brief review on key technologies in the battery management system of electric vehicles," *Frontiers of mechanical engineering*, vol. 14, pp. 47–64, 2019.

[44] D. Roman, S. Saxena, V. Robu, M. Pecht, and D. Flynn, "Machine learning pipeline for battery state-of-health estimation," *Nature Machine Intelligence*, vol. 3, no. 5, pp. 447–456, 2021.

[45] F. Zhu, X. Wang, and N. Yu, "Crowd tracking with dynamic evolution of group structures," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*. Springer, 2014, pp. 139–154.

[46] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE international conference on image processing (ICIP)*. IEEE, 2016, pp. 3464–3468.

[47] J. Su, P. Her, E. Clemens, E. Yaz, S. Schneider, and H. Medeiros, "Violence detection using 3d convolutional neural networks," in *2022 18th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2022, pp. 1–8.

[48] W. Ren, X. Wang, J. Tian, Y. Tang, and A. B. Chan, "Tracking-by-counting: Using network flows on crowd density maps for tracking multiple targets," *IEEE Transactions on Image Processing*, vol. 30, pp. 1439–1452, 2020.

[49] X. Chang, H. Pan, W. Sun, and H. Gao, "Yoltrack: Multitask learning based real-time multiobject tracking and segmentation for autonomous vehicles," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 12, pp. 5323–5333, 2021.

[50] J. Xing, H. Ai, L. Liu, and S. Lao, "Multiple player tracking in sports video: A dual-mode two-way bayesian inference approach with progressive observation modeling," *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp. 1652–1667, 2010.

[51] A. Generosi, S. Ceccacci, and M. Mengoni, "A deep learning-based system to track and analyze customer behavior in retail store," in *2018 IEEE 8th International Conference on Consumer Electronics-Berlin (ICCE-Berlin)*. IEEE, 2018, pp. 1–6.

[52] X. Hou, Y. Wang, and L.-P. Chau, "Vehicle tracking using deep sort with low confidence track filtering," in *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2019, pp. 1–6.

[53] A. Pramanik, S. K. Pal, J. Maiti, and P. Mitra, "Granulated rcnn and multi-class deep sort for multi-object detection and tracking," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 6, no. 1, pp. 171–181, 2021.

[54] J. Janai, F. Güney, A. Behl, A. Geiger *et al.*, "Computer vision for autonomous vehicles: Problems, datasets and state of the art," *Foundations and Trends® in Computer Graphics and Vision*, vol. 12, no. 1–3, pp. 1–308, 2020.

[55] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "Bytetrack: Multi-object tracking by associating every detection box," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*. Springer, 2022, pp. 1–21.

[56] J. Cao, J. Pang, X. Weng, R. Khirodkar, and K. Kitani, "Observation-centric sort: Rethinking sort for robust multi-object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9686–9696.

[57] G. Maggiolino, A. Ahmad, J. Cao, and K. Kitani, "Deep oc-sort: Multi-pedestrian tracking by adaptive re-identification," *arXiv preprint arXiv:2302.11813*, 2023.

[58] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky, "Bot-sort: Robust associations multi-pedestrian tracking," *arXiv preprint arXiv:2206.14651*, 2022.

[59] A. Cardenas, S. Amin, B. Sinopoli, A. Giani, A. Perrig, S. Sastry *et al.*, "Challenges for securing cyber physical systems," in *Workshop on future directions in cyber-physical systems security*, vol. 5, no. 1. Citeseer, 2009.

[60] A. Teixeira, S. Amin, H. Sandberg, K. H. Johansson, and S. S. Sastry, "Cyber security analysis of state estimators in electric power systems," in *49th IEEE Conference on Decision and Control (CDC)*, 2010, pp. 5991–5998.

[61] F. Pasqualetti, F. Dörfler, and F. Bullo, "Cyber-physical attacks in power networks: Models, fundamental limitations and monitor design," in *2011 50th IEEE Conference on Decision and Control and European Control Conference*, 2011, pp. 2195–2201.

[62] T. Keijzer and R. M. Ferrari, "A sliding mode observer approach for attack detection and estimation in autonomous vehicle platoons using event triggered communication," in *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2019, pp. 5742–5747.

[63] M. Fauser and P. Zhang, "Detection of cyber attacks in encrypted control systems," in *2022 American Control Conference (ACC)*, 2022, pp. 4992–4997.

[64] T. Roy and S. Dey, "Secure traffic networks in smart cities: Analysis and design of cyber-attack detection algorithms," in *2020 American Control Conference (ACC)*, 2020, pp. 4102–4107.

[65] K. Ghorbani, N. Orouji, and M. R. Mosavi, "Navigation message authentication based on one-way hash chain to mitigate spoofing attacks for gps l1," *Wireless Personal Communications*, vol. 113, pp. 1743–1754, 2020.

[66] A. A. Cárdenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, "Attacks against process control systems: risk assessment, detection, and response," in *Proceedings of the 6th ACM symposium on information, computer and communications security*, 2011, pp. 355–366.

[67] J.-S. Wang and G.-H. Yang, "Data-driven methods for stealthy attacks on tcp/ip-based networked control systems equipped with attack detectors," *IEEE Transactions on Cybernetics*, vol. 49, no. 8, pp. 3020–3031, 2019.

[68] A. Kundu, A. Sahu, K. Davis, and E. Serpedin, "Learning-based defense of false data injection attacks in power system state estimation," in *2019 North American Power Symposium (NAPS)*, 2019, pp. 1–6.

[69] Y. Mo, T. H.-J. Kim, K. Brancik, D. Dickinson, H. Lee, A. Perrig, and B. Sinopoli, "Cyber–physical security of a smart grid infrastructure," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 195–209, 2011.

[70] Z. Zhang, R. Deng, D. K. Y. Yau, P. Cheng, and J. Chen, "Zero-parameter-information fdi attacks against power system state estimation," in *2020 American Control Conference (ACC)*, 2020, pp. 2987–2992.

[71] Y. Chakhchoukh and H. Ishii, "Cyber attacks scenarios on the measurement function of power state estimation," in *2015 American Control Conference (ACC)*, 2015, pp. 3676–3681.

[72] D. Martynova and P. Zhang, "Completely stealthy attacks on cyber-physical system with parity space based monitoring," in *2019 American Control Conference (ACC)*, 2019, pp. 4424–4429.

[73] Y. Mo and B. Sinopoli, "On the performance degradation of cyber-physical systems under stealthy integrity attacks," *IEEE Transactions on Automatic Control*, vol. 61, no. 9, pp. 2618–2624, 2015.

[74] A. Rosich, H. Voos, Y. Li, and M. Darouach, "A model predictive approach for cyber-attack detection and mitigation in control systems," in *52nd IEEE Conference on Decision and Control*, 2013, pp. 6621–6626.

[75] R. A. Biroon, P. Pisu, and Z. Abdollahi, "Real-time false data injection attack detection in connected vehicle systems with pde modeling," in *2020 American Control Conference (ACC)*, 2020, pp. 3267–3272.

[76] S. D. Vyas, S. Kumar Padisala, and S. Dey, "A physics-informed neural network approach towards cyber attack detection in vehicle platoons," in *2023 American Control Conference (ACC)*, 2023, pp. 4537–4542.

[77] M. Sun, L. Ren, and N.-y. Chiang, "Data-driven probabilistic anomaly detection for electricity market under cyber attacks," in *2021 American Control Conference (ACC)*, 2021, pp. 4586–4591.

[78] B. Sundberg and D. B. Pourkargar, "Cyberattack awareness and resiliency of integrated moving horizon estimation and model predictive control of complex process networks," in *2023 American Control Conference (ACC)*, 2023, pp. 3815–3820.

[79] A. Jevtić and M. Ilić, "A dynamic strategy for cyber-attack detection in large-scale power systems via output clustering," in *2020 American Control Conference (ACC)*, 2020, pp. 4231–4236.

[80] E. Kontouras, A. Tzes, and L. Dritsas, "Set-theoretic detection of bias injection cyber-attacks on networked power systems," in *2018 Annual American Control Conference (ACC)*, 2018, pp. 165–170.

[81] G. Sun, T. Alpcan, B. I. P. Rubinstein, and S. Camtepe, "A communication security game on switched systems for autonomous vehicle platoons," in *2021 60th IEEE Conference on Decision and Control (CDC)*, 2021, pp. 2690–2695.

[82] T. Kobayashi and D. L. Simon, "Application of a bank of kalman filters for aircraft engine fault diagnostics," in *Turbo Expo: Power for Land, Sea, and Air*, vol. 36843, 2003, pp. 461–470.

[83] I. Idrissi, H. Chafouk *et al.*, "A bank of kalman filters for current sensors faults detection and isolation of dfig for wind turbine," in *2017 International Renewable and Sustainable Energy Conference (IRSEC)*. IEEE, 2017, pp. 1–6.

[84] T. E. Menke and P. S. Maybeck, "Sensor/actuator failure detection in the vista f-16 by multiple model adaptive estimation," *IEEE Transactions on aerospace and electronic systems*, vol. 31, no. 4, pp. 1218–1229, 1995.

[85] R. Isermann, "Model-based fault-detection and diagnosis–status and applications," *Annual Reviews in control*, vol. 29, no. 1, pp. 71–85, 2005.

[86] D. T. Bill Messner, "Control tutorials for matlab and simulink - motor speed: System modeling," 2019. [Online]. Available: http://ctms.engin.umich.edu/CTMS/index.php?example=MotorSpeed&section=SystemModeling

[87] P. Shen, M. Ouyang, L. Lu, J. Li, and X. Feng, "The co-estimation of state of charge, state of health, and state of function for lithium-ion batteries in electric vehicles," *IEEE Transactions on vehicular technology*, vol. 67, no. 1, pp. 92–103, 2017.

[88] M. Kwak, B. Lkhagvasuren, J. Park, and J.-H. You, "Parameter identification and soc estimation of a battery under the hysteresis effect," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 11, pp. 9758–9767, 2019.

[89] X. Hu, H. Yuan, C. Zou, Z. Li, and L. Zhang, "Co-estimation of state of charge and state of health for lithium-ion batteries based on fractional-order calculus," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 10 319–10 329, 2018.

[90] S. Afshar, K. Morris, and A. Khajepour, "State-of-charge estimation using an ekf-based adaptive observer," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 5, pp. 1907–1923, 2018.

[91] G. L. Plett, "Extended kalman filtering for battery management systems of lipb-based hev battery packs: Part 3. state and parameter estimation," *Journal of Power sources*, vol. 134, no. 2, pp. 277–292, 2004.

[92] M. Partovibakhsh and G. Liu, "An adaptive unscented kalman filtering approach for online estimation of model parameters and state-of-charge of lithium-ion batteries for autonomous mobile robots," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 1, pp. 357–363, 2014.

[93] G. L. Plett, "Sigma-point kalman filtering for battery management systems of lipb-based hev battery packs: Part 2: Simultaneous state and parameter estimation," *Journal of power sources*, vol. 161, no. 2, pp. 1369–1384, 2006.

[94] A. Chanthakett, M. T. Arif, M. Khan, and A. M. Oo, "Improvement of hydrogen production from coal gasification for power generation application," in *2020 Australasian Universities Power Engineering Conference (AUPEC)*. IEEE, 2020, pp. 1–7.

[95] L. Timofeeva and L. Mallinson, "Ageing studies and lifetime extension of materials," *New York*, pp. 191–198, 2001.

[96] Y. Zou, X. Hu, H. Ma, and S. E. Li, "Combined state of charge and state of health estimation over lithium-ion battery cell cycle lifespan for electric vehicles," *Journal of Power Sources*, vol. 273, pp. 793–803, 2015.

[97] B. S. Bhangu, P. Bentley, D. A. Stone, and C. M. Bingham, "Nonlinear observers for predicting state-of-charge and state-of-health of lead-acid batteries for hybrid-electric vehicles," *IEEE transactions on vehicular technology*, vol. 54, no. 3, pp. 783–794, 2005.

[98] Y. Gao, K. Liu, C. Zhu, X. Zhang, and D. Zhang, "Co-estimation of state-of-charge and state-of-health for lithium-ion batteries using an enhanced electrochemical model," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 3, pp. 2684–2696, 2021.

[99] S. Li, K. Li, E. Xiao, and C.-K. Wong, "Joint soc and soh estimation for zinc–nickel single-flow batteries," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 10, pp. 8484–8494, 2019.

[100] D. N. How, M. A. Hannan, M. S. H. Lipu, K. S. Sahari, P. J. Ker, and K. M. Muttaqi, "State-of-charge estimation of li-ion battery in electric vehicles: A deep neural network approach," *IEEE Transactions on Industry Applications*, vol. 56, no. 5, pp. 5565–5574, 2020.

[101] M. A. Hannan, D. N. How, M. H. Lipu, P. J. Ker, Z. Y. Dong, M. Mansur, and F. Blaabjerg, "Soc estimation of li-ion batteries with learning rate-optimized deep fully convolutional network," *IEEE Transactions on Power Electronics*, vol. 36, no. 7, pp. 7349–7353, 2020.

[102] M. A. Hannan, D. N. How, M. B. Mansor, M. S. H. Lipu, P. J. Ker, and K. M. Muttaqi, "State-of-charge estimation of li-ion battery using gated recurrent unit with one-cycle learning rate policy," *IEEE Transactions on Industry Applications*, vol. 57, no. 3, pp. 2964–2971, 2021.

[103] U. EPA, "Epa urban dynamometer driving schedule (udds)," 2017.

[104] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler, "Mot16: A benchmark for multi-object tracking," *arXiv preprint arXiv:1603.00831*, 2016.

[105] H. Shi, H. Ghahremannezhad, and C. Liu, "Anomalous driving detection for traffic surveillance video analysis," in *2021 IEEE International Conference on Imaging Systems and Techniques (IST)*. IEEE, 2021, pp. 1–6.

[106] T. Han, L. Bai, J. Gao, Q. Wang, and W. Ouyang, "Dr. vic: Decomposition and reasoning for video individual counting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 3083–3092.

[107] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE international conference on image processing (ICIP)*. IEEE, 2017, pp. 3645–3649.

[108] Y.-H. Wang, "Smiletrack: Similarity learning for multiple object tracking," *arXiv preprint arXiv:2211.08824*, 2022.

[109] Y. Du, Z. Zhao, Y. Song, Y. Zhao, F. Su, T. Gong, and H. Meng, "Strongsort: Make deepsort great again," *IEEE Transactions on Multimedia*, 2023.

[110] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3464–3468.

[111] Z. Liu, X. Wang, C. Wang, W. Liu, and X. Bai, "Sparsetrack: Multi-object tracking by performing scene decomposition based on pseudo-depth," *arXiv preprint arXiv:2306.05238*, 2023.

[112] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "Fairmot: On the fairness of detection and re-identification in multiple object tracking," *International Journal of Computer Vision*, vol. 129, pp. 3069–3087, 2021.

[113] T. Meinhardt, A. Kirillov, L. Leal-Taixe, and C. Feichtenhofer, "Trackformer: Multi-object tracking with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 8844–8854.

[114] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV*. Springer, 2020, pp. 474–490.

[115] J. Pang, L. Qiu, X. Li, H. Chen, Q. Li, T. Darrell, and F. Yu, "Quasi-dense similarity learning for multiple object tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 164–173.

[116] Z. Wang, L. Zheng, Y. Liu, Y. Li, and S. Wang, "Towards real-time multi-object tracking," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*. Springer, 2020, pp. 107–122.

[117] D. T. Pham, J. Verron, and M. C. Roubaud, "A singular evolutive extended kalman filter for data assimilation in oceanography," *Journal of Marine systems*, vol. 16, no. 3-4, pp. 323–340, 1998.

[118] J. Ballabrera-Poy, A. J. Busalacchi, and R. Murtugudde, "Application of a reduced-order kalman filter to initialize a coupled atmosphere–ocean model: Impact on the prediction of el nino," *Journal of climate*, vol. 14, no. 8, pp. 1720–1737, 2001.

[119] D. Simon, "Reduced order kalman filtering without model reduction." *Control & Intelligent Systems*, vol. 35, no. 2, 2007.

[120] S. F. Schmidt, "Application of state-space methods to navigation problems," in *Advances in control systems*. Elsevier, 1966, vol. 3, pp. 293–340.

[121] R. G. Brown and P. Y. Hwang, "Introduction to random signals and applied kalman filtering: with matlab exercises and solutions," *Introduction to random signals and applied Kalman filtering: with MATLAB exercises and solutions*, 1997.

[122] M. S. Grewal and A. P. Andrews, *Kalman filtering: Theory and Practice with MATLAB*. John Wiley & Sons, 2014.

[123] J. Su, S. Schneider, and E. Yaz, "Multi-object tracking using video sequences with improved performance and reduced computation cost," in *2024 IEEE Conference on Control Technology and Applications (CCTA)*, 2024.

[124] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," *arXiv preprint arXiv:2107.08430*, 2021.

[125] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *European conference on computer vision*. Springer, 2016, pp. 17–35.

[126] S. Sun, N. Akhtar, H. Song, A. S. Mian, and M. Shah, "Deep affinity network for multiple object tracking," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[127] B. Pang, Y. Li, Y. Zhang, M. Li, and C. Lu, "Tubetk: Adopting tubes to track multi-object in a one-step training model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6308–6318.

[128] F. Zeng, B. Dong, T. Wang, C. Chen, X. Zhang, and Y. Wei, "Motr: End-to-end multiple-object tracking with transformer," *arXiv preprint arXiv:2105.03247*, 2021.

[129] J. Peng, C. Wang, F. Wan, Y. Wu, Y. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and Y. Fu, "Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking," in *European Conference on Computer Vision*. Springer, 2020, pp. 145–161.

[130] J. Wu, J. Cao, L. Song, Y. Wang, M. Yang, and J. Yuan, "Track to detect and segment: An online multi-object tracker," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 352–12 361.

[131] S. Han, P. Huang, H. Wang, E. Yu, D. Liu, X. Pan, and J. Zhao, "Mat: Motion-aware multi-object tracking," *arXiv preprint arXiv:2009.04794*, 2020.

[132] L. Zheng, M. Tang, Y. Chen, G. Zhu, J. Wang, and H. Lu, "Improving multiple object tracking with single object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2453–2462.

[133] Y. Xu, Y. Ban, G. Delorme, C. Gan, D. Rus, and X. Alameda-Pineda, "Transcenter: Transformers with dense queries for multiple-object tracking," *arXiv preprint arXiv:2103.15145*, 2021.

[134] Y. Wang, K. Kitani, and X. Weng, "Joint object detection and multi-object tracking with graph neural networks," *arXiv preprint arXiv:2006.13164*, 2020.

[135] W. Li, Y. Xiong, S. Yang, M. Xu, Y. Wang, and W. Xia, "Semi-tcl: Semi-supervised track contrastive representation learning," *arXiv preprint arXiv:2107.02396*, 2021.

[136] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "Fairmot: On the fairness of detection and re-identification in multiple object tracking," *arXiv preprint arXiv:2004.01888*, 2020.

[137] E. Yu, Z. Li, S. Han, and H. Wang, "Relationtrack: Relation-aware multiple object tracking with decoupled representation," *arXiv preprint arXiv:2105.04322*, 2021.

[138] P. Tokmakov, J. Li, W. Burgard, and A. Gaidon, "Learning to track with object permanence," *arXiv preprint arXiv:2103.14258*, 2021.

[139] C. Liang, Z. Zhang, Y. Lu, X. Zhou, B. Li, X. Ye, and J. Zou, "Rethinking the competition between detection and reid in multi-object tracking," *arXiv preprint arXiv:2010.12138*, 2020.

[140] P. Sun, Y. Jiang, R. Zhang, E. Xie, J. Cao, X. Hu, T. Kong, Z. Yuan, C. Wang, and P. Luo, "Transtrack: Multiple-object tracking with transformer," *arXiv preprint arXiv:2012.15460*, 2020.

[141] C. Shan, C. Wei, B. Deng, J. Huang, X.-S. Hua, X. Cheng, and K. Liang, "Tracklets predicting based adaptive graph tracking," *arXiv preprint arXiv:2010.09015*, 2020.

[142] C. Liang, Z. Zhang, X. Zhou, B. Li, Y. Lu, and W. Hu, "One more check: Making" fake background" be tracked again," *arXiv preprint arXiv:2104.09441*, 2021.

[143] Q. Wang, Y. Zheng, P. Pan, and Y. Xu, "Multiple object tracking with correlation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3876–3886.

[144] P. Chu, J. Wang, Q. You, H. Ling, and Z. Liu, "Transmot: Spatial-temporal graph transformer for multiple object tracking," *arXiv preprint arXiv:2104.00194*, 2021.

[145] F. Yang, X. Chang, S. Sakti, Y. Wu, and S. Nakamura, "Remot: A model-agnostic refinement for multiple object tracking," *Image and Vision Computing*, vol. 106, p. 104091, 2021.

[146] J. Su, S. Schneider, and E. Yaz, "Reduced-order H_∞ filter for linear systems," in *IECON*, 2024.

[147] ——, "Permanent magnet synchronous motor speed and position estimation using reduced-order extended kalman filter," in *IFAC-PapersOnLine*, 2024.

Appendix: Code

```
% Code for this dissertation can be found on jiayisu.com
```