# Finite Interval Parameter and State Estimation in LTI Systems Using Kernel-Based Multiple Regression

*Surya Kumar Devarajan*

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

November 2020

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Master of Engineering. 

**Abstract**

A recursive version of a generalized least squares for parameter estimation in Reproducing Kernel Hilbert Space (RKHS) is presented in this thesis. It begins with the understanding and derivation of a special construction of a forward-backward kernel representation of linear differential invariants for a third-order linear system. Methods for parameter and state estimation from single noisy realizations of the system output on a time interval $[a, b]$ is discussed. Parameter estimation is solved by the way of stochastic regression. Generalized Least squares with covariance weighting is employed to deal with high noise. Once the recursive approach estimates the parameters, the output and time derivatives are reconstructed by projection onto the span of fundamental solutions.

**Résumé**

Une version récursive des moindres carrés généralisés pour l'estimation des paramètres dans Reproducing Kernel Hilbert Space (RKHS) est présentée dans cette thèse. Il commence par la compréhension et la dérivation d'une construction spéciale d'une représentation de noyau avant-arrière d'invariants différentiels linéaires pour un système linéaire du troisième ordre. Les méthodes d'estimation des paramètres et des états à partir de réalisations bruyantes uniques de la sortie système sur un intervalle de temps $[a, b]$ sont discutées. L'estimation des paramètres est résolue par la voie de la régression stochastique. Les moindres carrés généralisés avec pondération par covariance sont utilisés pour traiter le bruit élevé. Une fois les paramètres estimés par l'approche récusive, la sortie et leurs dérivées temporelles sont reconstruites par projection sur l'étendue des solutions fondamentales.

## Acknowledgment

I would like to express my sincere gratitude to my academic supervisor Prof. Hannah Michalska for the continuous support of my Master's thesis, for her motivation, enthusiasm, and immense knowledge. Her guidance helped me throughout the research and writing of this thesis. I thank her for supporting and guiding me.

I am grateful to my friend and fellow research partner Santosh Devapati, who has been a great support and a source of encouragement in my academic journey in the last two years.

I also thank Debarshi Ghoshal and fellow lab mates for the healthy discussions and reviewing of the codes.

I would thank my parents, friends and family who cheered me up and kept me going in the times of ups and downs.

## Preface

This is to declare that the work presented in this document was completed and carried out by Surya Kumar Devarajan under the guidance of Professor Hannah Michalska. It builds on the efforts of Debarshi Patanjali Ghoshal, the Ph.D. scholar in the research group who carried out the parameter estimation for linear systems by least-squares and recursive least-squares. The forward-backward kernel-based state and parameter estimation using multiple regression equations, which are very efficient in the presence of heteroskedasticity for a third-order system, were verified and coded. The theoretical background for the RKHS approaches is based on the research notes by Professor Hannah Michalska, which is duly acknowledged.

# List of Figures

# List of Tables

# List of Acronyms

| | |
|---|---|
| RKHS | Reproducing Kernel Hilbert Space |
| KS | Kolmogorov Smirnov |
| LTI | Linear Time Invariant |
| RMSD | Root Mean Square Difference |
| SISO | Single Input Single Output |
| BLUE | Best Linear Unbiased Estimator |
| OLS | Ordinary Least Squares |
| RLS | Recursive Least Squares |
| GLS | Generalised Least Squares |
| IV | Instrument Variable |

# Chapter 1

# Introduction

Parameter estimation and system identification have important applications in system modelling, control, and analysis [1] [2] and thus have received research attention in past few decades [3] [4]. The mathematics behind this is the Control Theory. A controller is used to provide the corrective action to achieve the desired output. As it is known, a controller is of two types: open-loop and closed-loop control. The open-loop system does not have feedback, and the output is solely dependent on the input, whereas a closed-loop system involves a control scheme that incorporates a sensor that measures the error. The open-loop system can be converted into a closed-loop system by providing feedback to the system. The system achieves and maintains the desired output by comparing it with the actual output. It does this by generating an error signal which is the difference between the reference input and the output. A general control system employing feedback is shown in the form of a block diagram below.

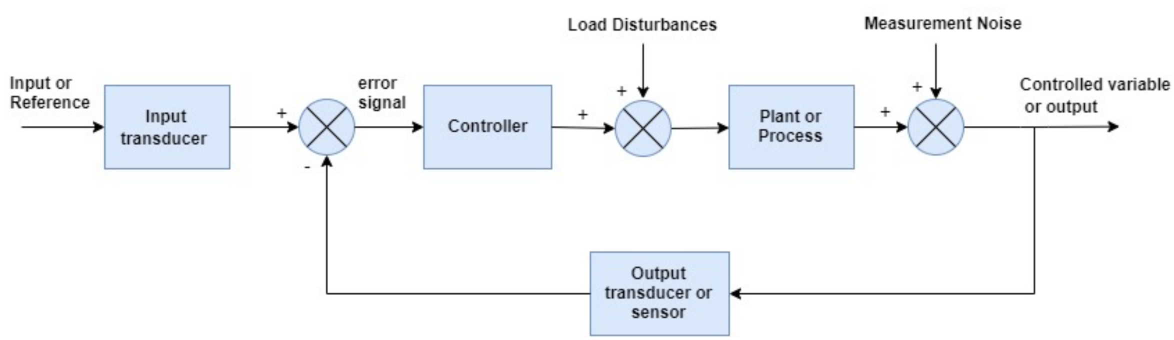


**Figure 1.1** Block diagram of closed-loop control systems

The minimum set of variables (state variables) with their current values, along with the input signal, predict the future behavior of the system [5]. The knowledge of all the system states is usually required to generate a control law. The input transducer, which acts as feedforward, gives the desired response to the input. The feedback system handles process uncertainties, load disturbances, and measurement noise. The load disturbance drives the working of the plant away from its desired behavior, and measurement noise distorts the process variable. It is expected that the controller can compensate for such noise and disturbances. For a large system, it is not possible to measure all the state variables. An estimator must hence be designed that can estimate the states from both output and input. The problem of estimator design for an observable system has been extensively dealt with by mathematicians, especially, Kalman and Luenberger [6], [7], [8], [9].

Recently, new non-asymptotic methods evolved for state and parameter estimation on finite time intervals of time [5], comprising a specific set of methods known as algebraic estimation methods. Algebraic estimation is based on the idea of repeated differentiation of measured system output. The original algebraic differentiator was based on the truncated Taylor series signal approximation. The operation of differentiation is expressed in terms of the operation of integration. Noise attenuation is attempted by repeated integration and shaping of the annihilator functions used to eliminate the effect of the initial conditions [10] [11] [12]. However, this method requires frequent re-initialization when used forward in time, and its noise rejection properties are characterized as non-standard [13]. These methods are sensitive to measurement noise.

A novel estimation scheme for the state and its time derivatives is implemented using the knowledge of the system characteristic equation in [14]. In this approach, the state equations are replaced with an output reproducing property on an arbitrary time interval, which follows directly from the knowledge of the system characteristic equation. Such a behavioral model is derived from the differential invariance, which is characteristic to the system [15]. This eliminates the need for initial conditions, and the mathematical model is in the form of a homogeneous Fredholm integral equation of the second kind with a Hilbert Schmidt kernel [16] [17]. The mathematical interpretation of the behavioral model in a RKHS allows extracting signal and its time derivatives from output measurement, despite the presence of noise [16]. The Double-sided kernel approach is discussed in chapter 2.

Algebraic state and parameter estimation of linear systems based on a special construction of a forward-backward kernel representation of linear differential invariants are extended to handle large noise in output measurement [18] [19] [20] [21] [14]. The parameter estimation is solved by stochastic regression. The regression model does not satisfy the Gauss-Markov theorem's assumptions in that the random regressor is correlated with the regression error, which is also heteroskedastic. These complications do not impede achieving high accuracy of estimation.

## 1.1 Thesis summary: objectives and organization

Parameter estimation of the differential invariance is carried out as a least-squares approach. In the OLS approach, the variance of the error term is the same across observations (homoscedasticity). Heteroskedasticity occurs when the variance of the error term differs across different observations. OLS estimator is inefficient with heteroskedasticity as it gives equal weight to all observations. Hence, the OLS estimator is no longer BLUE, and in addition to this, confidence intervals and hypothesis tests are no longer valid. Generalized least squares (GLS) works well with heteroskedasticity as it takes into account the inequality of variance in the observations. Recursive Least Squares provides an efficient way to update the approximate solution of least squares and has faster parameter convergence [14] [22] [23] [24]. The objectives of this thesis are stated below.

- Understand the mathematical background behind the parameter and state estimation by algebraic methods.
- Study and understand the development of novel kernel development for SISO LTI systems [19]. A third-order system is considered.
- Study and understand the previous works on the parameter and state estimation carried out using the Double-Sided Kernel Approach.
- Develop a forward-backward kernel-based state and parameter estimation using multiple regression equations to compensate for the heteroskedasticity. The Recursive Generalized Least Squares is used for regression, and it employs inverse covariance weighting.
- Present and discuss simulation results confirming the effectiveness of the proposed method.

The thesis is structured as follows:

Chapter 1 provides the literature review and the motivation behind this work. It also gives an introduction to parameter and state estimation by algebraic methods and the least-squares approach. The thesis objectives and organization are explained.

Chapter 2 is dedicated to the detailed derivation and understanding of the double-sided kernels for homogeneous SISO LTI systems. A third-order system is considered. The kernel representation for the derivatives of the third-order system's output is also derived.

Chapter 3 discusses the previous works done on state and parameter estimation. This thesis builds on the previous work done on the Recursive parameter estimation carried out by DP Ghoshal in [14] [24].

Chapter 4 provides a detailed summary of forward-backward kernel-based state and parameter estimation using multiple regression equations. Heteroskedasticity is dealt with by using Recursive Generalized Least Squares. Also, kernels with different dimensions are used to estimate the parameters of the same dimensions systems.

.

Chapter 5 presents the numerical results and simulation of the proposed mathematical approach. A comparative study with the previously developed methods is also provided.

Chapter 6 gives a summary of this thesis as well as discusses directions for future work.

# Chapter 2

# A Double Sided Kernel in SISO LTI Representation

As mentioned in the previous chapter, algebraic estimation as presented in [5] by M. Fliess and H. Sira-Ramirez has many advantages but also some shortcomings like frequent re-initialization when used forward in time, and falls short of standard noise rejection properties. To overcome that, in [19], a double-sided kernel representation of system invariance, ensuring no singularities was developed. In any observation window, the output reproducing property of the system invariance can be used to characterize system trajectories.

## 2.1 Development of Double Sided Kernel [19]

### 2.1.1 General n-th order SISO LTI system:

The estimation problem assumes a SISO LTI system:

$$\dot{x} = Ax; \quad y = Cx; \quad x \in \mathbb{R}^n \tag{2.1}$$

where the system matrix A is in canonical form,

$$\begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-1} \end{bmatrix} \tag{2.2}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \tag{2.3}$$

and,

$$C = \begin{bmatrix} 1 & 0 & 0 & \dots \end{bmatrix} \tag{2.4}$$

so that the characteristic equation of the system is,

$$y^{(n)}(t) + a_{n-1}y^{(n-1)}(t) + \cdots + a_1 y^{(1)}(t) + a_0 y(t) = 0 \tag{2.5}$$

Also the input-output equation of the system is represented as:

$$y^{(n)}(t) + a_{n-1}y^{(n-1)}(t) + \cdots + a_1 y^{(1)}(t) + a_0 y(t) = -b_{n-1}u^{(n-1)}(t) - \cdots - b_0 u(t) \tag{2.6}$$

The unknown parameters $a_i$ and $b_i$, $i = 0, .., n-1$ need to be estimated from noisy observations of the system's output $y_M(t)$ $t \in [a, b]$ for the given observation horizon $[a, b]$.

### 2.1.2 Kernel Representation [25]

The foundation of the finite interval estimation approach is the integral representation of the controlled differential invariance of the system in (2.6). The construction relies on the introduction of functions, which act as annihilators of initial or boundary conditions.

*Definition of Annihilators*
A pair of smooth (class $\mathcal{C}^{\infty}$) functions

$$(\alpha_a, \alpha_b), \alpha_s : [a, b] \to \mathbb{R}$$

$s = a$ or $b$ , is an annihiltor of the boundary conditions for system (2.1) , if the functions $\alpha_s$ are non-negative, monotonic and vanish with their derivatives up to order $n-1$ at the respective ends of the the interval $[a, b]$; i.e.

$$\alpha_s^{(i)}(s) = 0 \quad i = 0, \cdots, n-1; \quad s = a, b; \quad \alpha_s^{(0)} \equiv \alpha_s$$

such that their sum is strictly positive, i.e. for some constant c>0,

$$\alpha_{ab}(t) := \alpha_a(t) + \alpha_b(t) > c \text{ ; for } t \in [a, b]$$

An example of such an annihilator for system (2.1) is the pair,

$$\begin{aligned}
&\alpha_a(t) := (t-a)^n, \quad \alpha_b(t) := (b-t)^n \quad t \in [a, b] \\
&\alpha_{ab}(t) := \alpha_a(t) + \alpha_b(t) > 0 \quad t \in [a, b] \\
&\alpha_{ab}(s) = (b-a)^n, \quad s = a, b
\end{aligned}$$

Applying the concept of annihilators, the integral representation of the system (2.1) can be obtained [18], as stated below.

**Theorem 1.** *There exist Hilbert-Schmidt kernels $K_{DS,y}, K_{DS,u}$, such that the input and output functions $u$ and $y$ of (2.1) satisfy*

$$y(t) \quad = \quad \alpha_{ab}^{-1}(t) \left[ \int_a^b K_{DS,y}(n, t, \tau) y(\tau) d\tau + \int_a^b K_{DS,u}(n, t, \tau) u(\tau) d\tau \right] \tag{2.7}$$

*with*

$$\alpha_{ab}^{-1}(t) := \frac{1}{(t-a)^n + (b-t)^n} \tag{2.8}$$

*Hilbert-Schmidt double-sided kernels of* (2.7) *are square integrable functions on* $L^2[a,b] \times L^2[a,b]$ *and are expressed in terms of the "forward" and "backward" kernels for the respective integral representations of the operators* $\alpha_a F^*$ *and* $\alpha_b F^*$*, with the operator* $F^*$ *denoting the formal adjoint of* $F$ *:*

$$K_{DS,y}(n,t,\tau) \triangleq \begin{cases} K_{F,y}(n,t,\tau), & \text{for } \tau \leq t \\ K_{B,y}(n,t,\tau), & \text{for } \tau > t \end{cases} \tag{2.9}$$

$$K_{DS,u}(n,t,\tau) \triangleq \begin{cases} K_{F,u}(n,t,\tau), & \text{for } \tau \leq t \\ K_{B,u}(n,t,\tau), & \text{for } \tau > t \end{cases} \tag{2.10}$$

$$K_{DS,y}, K_{DS,u} \text{ are } n-1 \text{ times differentiable as functions of } t$$

The easiest proof of the representation theorem is conducted by mathematical induction on $n$, delivering explicit formulae for both kernels, as stated next.
The full proof corresponding to a third-order system is described in later sections.
The kernels of Theorem 1 have the following expression,

$$\begin{aligned} K_{F,y}(n,t,\tau) = &\sum_{j=1}^{n} (-1)^{j+1} \binom{n}{j} \frac{n!(t-\tau)^{j-1}(\tau-a)^{n-j}}{(n-j)!(j-1)!} \\ &+ \sum_{i=0}^{n-1} a_i \sum_{j=0}^{i} (-1)^{j+1} \binom{i}{j} \frac{n!(t-\tau)^{n-i+j-1}(\tau-a)^{n-j}}{(n-j)!(n-i+j-1)!} \end{aligned} \tag{2.11}$$

$$\begin{aligned} K_{B,y}(n,t,\tau) = &\sum_{j=1}^{n} \binom{n}{j} \frac{n!(t-\tau)^{j-1}(b-\tau)^{n-j}}{(n-j)!(j-1)!} \\ &+ \sum_{i=0}^{n-1} a_i \sum_{j=0}^{i} \binom{i}{j} \frac{n!(t-\tau)^{n-i+j-1}(b-\tau)^{n-j}}{(n-j)!(n-i+j-1)!} \end{aligned} \tag{2.12}$$

$$K_{F,u}(n,t,\tau) = \sum_{i=0}^{n-1} b_i \sum_{j=0}^{i} (-1)^{j+1} \begin{pmatrix} i \\ j \end{pmatrix} \frac{n!(t-\tau)^{n-i+j-1}(\tau-a)^{n-j}}{(n-j)!(n-i+j-1)!} \tag{2.13}$$

$$K_{B,u}(n,t,\tau) = \sum_{i=0}^{n-1} b_i \sum_{j=0}^{i} \begin{pmatrix} i \\ j \end{pmatrix} \frac{n!(t-\tau)^{n-i+j-1}(b-\tau)^{n-j}}{(n-j)!(n-i+j-1)!} \tag{2.14}$$

### 2.1.3 Kernel expression for derivatives of the output function

Due to the regularity properties of the kernel functions, it is fairly simple to obtain the corresponding recursive formula for the time derivatives of the output function, $y^{(i)}, i = 1, \cdots, n-1$

**Theorem 2.** *There exist Hilbert-Schmidt kernels* $K_{F,k,y}, K_{F,k,u}, K_{B,k,y}, K_{B,k,u}$, $k = 1, \cdots, n-1$ *such that the derivatives of the output function in* (2.1) *can be computed recursively as follows:*

$$y^{(k)}(t) = \tfrac{1}{(t-a)^n+(b-t)^n}\Bigg[\textstyle\sum_{i=1}^{k}(-1)^{i+1}\begin{pmatrix} p+i-1 \\ i \end{pmatrix}\tfrac{n!(t-a)^{n-i}y^{(k-i)}(t)}{(n-i)!}$$

$$+\textstyle\sum_{i=p}^{n-1}a_i\sum_{j=0}^{i-p}(-1)^{j+1}\begin{pmatrix} p+j-1 \\ j \end{pmatrix}\tfrac{n!(t-a)^{n-j}y^{(i-j-p)}(t)}{(n-j)!}+\int_a^t K_{F,k,y}(n,p,t,\tau)y(\tau)d\tau$$

$$+\textstyle\sum_{i=p}^{n-1}b_i\sum_{j=0}^{i-p}(-1)^{j+1}\begin{pmatrix} p+j-1 \\ j \end{pmatrix}\tfrac{n!(t-a)^{n-j}u^{(i-j-p)}(t)}{(n-j)!}+\int_a^t K_{F,k,u}(n,p,t,\tau)u(\tau)d\tau$$

$$-\textstyle\sum_{i=1}^{k}\begin{pmatrix} p+i-1 \\ i \end{pmatrix}\tfrac{n!(b-t)^{n-i}y^{(k-i)}(t)}{(n-i)!}-\sum_{i=p}^{n-1}a_i\sum_{j=0}^{i-p}\begin{pmatrix} p+j-1 \\ j \end{pmatrix}\tfrac{n!(b-t)^{n-j}y^{(i-j-p)}(t)}{(n-j)!}$$

$$+\textstyle\int_t^b K_{B,k,y}(n,p,t,\tau)y(\tau)d\tau-\sum_{i=p}^{n-1}b_i\sum_{j=0}^{i-p}\begin{pmatrix} p+j-1 \\ j \end{pmatrix}\tfrac{n!(b-t)^{n-j}u^{(i-j-p)}(t)}{(n-j)!} \tag{2.15}$$

$$+\textstyle\int_t^b K_{B,k,u}(n,p,t,\tau)u(\tau)d\tau]$$

$$where\ ,\ p=n-k$$

$$K_{F,k,y}(n,p,t,\tau)=\sum_{j=1}^{p}(-1)^{j+n-p+1}\begin{pmatrix} n \\ n-p+j \end{pmatrix}\frac{n!(t-\tau)^{j-1}(\tau-a)^{p-j}}{(p-j)!(j-1)!}$$

$$+\sum_{i=0}^{p-1}a_i\sum_{j=0}^{i}(-1)^{j+1}\begin{pmatrix} i \\ j \end{pmatrix}\frac{n!(t-\tau)^{p-i+j-1}(\tau-a)^{n-j}}{(n-j)!(p-i+j-1)!} \tag{2.16}$$

$$+\sum_{i=p}^{n-1}a_i\sum_{j=1}^{p}(-1)^{j+i-p+1}\begin{pmatrix} i \\ i-p+j \end{pmatrix}\frac{n!(t-\tau)^{j-1}(\tau-a)^{n-i+p-j}}{(n-i+p-j)!(j-1)!}$$

$$K_{F,k,u}(n,p,t,\tau) = \sum_{i=0}^{p-1} b_i \sum_{j=0}^{i} (-1)^{j+1} \begin{pmatrix} i \\ j \end{pmatrix} \frac{n!(t-\tau)^{p-i+j-1}(\tau-a)^{n-j}}{(n-j)!(p-i+j-1)!} + \sum_{i=p}^{n-1} b_i \sum_{j=1}^{p} (-1)^{j+i-p+1} \begin{pmatrix} i \\ i-p+j \end{pmatrix} \frac{n!(t-\tau)^{j-1}(\tau-a)^{n-i+p-j}}{(n-i+p-j)!(j-1)!} \tag{2.17}$$

$$\begin{aligned} K_{B,k,y}(n,p,t,\tau) = &\sum_{j=1}^{p} \begin{pmatrix} n \\ n-p+j \end{pmatrix} \frac{n!(t-\tau)^{j-1}(b-\tau)^{p-j}}{(p-j)!(j-1)!} \\ &+ \sum_{i=0}^{p-1} a_i \sum_{j=0}^{i} \begin{pmatrix} i \\ j \end{pmatrix} \frac{n!(t-\tau)^{p-i+j-1}(b-\tau)^{n-j}}{(n-j)!(p-i+j-1)!} \\ &+ \sum_{i=p}^{n-1} a_i \sum_{j=1}^{p} \begin{pmatrix} i \\ i-p+j \end{pmatrix} \frac{n!(t-\tau)^{j-1}(b-\tau)^{n-i+p-j}}{(n-i+p-j)!(j-1)!} \end{aligned} \tag{2.18}$$

$$\begin{aligned} K_{B,k,u}(n,p,t,\tau) = &\sum_{i=0}^{p-1} b_i \sum_{j=0}^{i} \begin{pmatrix} i \\ j \end{pmatrix} \frac{n!(t-\tau)^{p-i+j-1}(b-\tau)^{n-j}}{(n-j)!(p-i+j-1)!} \\ &+ \sum_{i=p}^{n-1} b_i \sum_{j=1}^{p} \begin{pmatrix} i \\ i-p+j \end{pmatrix} \frac{n!(t-\tau)^{j-1}(b-\tau)^{n-i+p-j}}{(n-i+p-j)!(j-1)!} \end{aligned} \tag{2.19}$$

## 2.2 Kernel development for a third order system [25]

The double-sided kernel for a general third order characteristic polynomial is derived in this section

$$y^{(3)}(t) + a_2 y^{(2)}(t) + a_1 y^{(1)}(t) + a_0 y(t) = 0 \tag{2.20}$$

on an interval $[a, b]$. By multiplying(2.20) with $(\xi - a)^3$ and $(b - \zeta)^3$ we get

$$(\xi - a)^3 y^{(3)}(t) + a_2(\xi - a)^3 y^{(2)}(t) + a_1(\xi - a)^3 y^{(1)}(t) + a_0(\xi - a)^3 y(t) = 0 \tag{2.21}$$

$$(b - \zeta)^3 y^{(3)}(t) + a_2(b - \zeta)^3 y^{(2)}(t) + a_1(b - \zeta)^3 y^{(1)}(t) + a_0(b - \zeta)^3 y(t) = 0 \tag{2.22}$$

Integrate (2.21) and (2.22) thrice on the interval $[a, a+\tau]$ and $[b-\sigma, b]$ , meaning we would integrate the (2.20) in the forward direction during the interval $[a, a+\tau]$ and in the backward direction during the interval $[b, b-\sigma]$.
Now consider that we integrate the first term (2.21) once,

$$\int_a^{a+\tau} (\xi - a)^3 y^{(3)}(\xi)\, \mathrm{d}\xi$$

$$\begin{aligned}
&= (\xi - a)^3 y^{(2)}(\xi) \mid_a^{a+\tau} - \int_a^{a+\tau} 3(\xi - a)^2 y^{(2)}(\xi)\, \mathrm{d}\xi \\
&= \tau^3 y^{(2)}(a+\tau) - \left[ 3(\xi - a)^2 y^{(1)}(\xi) \mid_a^{a+\tau} - \int_a^{a+\tau} 6(\xi - a) y^{(1)}(\xi) \mathrm{d}\xi \right] \\
&= \tau^3 y^{(2)}(a+\tau) - 3\tau^2 y^{(1)}(a+\tau) + 6(\xi - a) y^{(1)}(\xi) \mid_a^{a+\tau} - \int_a^{a+\tau} 6y(\xi)\, \mathrm{d}\xi \\
&= \tau^3 y^{(2)}(a+\tau) - 3\tau^2 y^{(1)}(a+\tau) + 6\tau y(a+\tau) - \int_a^{a+\tau} 6y(\xi)\, \mathrm{d}\xi
\end{aligned} \tag{2.23}$$

Integrating the same for a second time, we get the upper limit on the integral to be a 'dummy variable', by setting $\xi' = a + \tau$ ,
$\tau^3 y^{(2)}(a+\tau)$ becomes $(\xi' - a)^3 y^{(2)}(\xi')$
$3\tau^2 y^{(1)}(a+\tau)$ is now $3(\xi' - a)^2 y^{(1)}(\xi')$
$6\tau y(a+\tau)$ becomes $6(\xi' - a) y(\xi')$

Integrating (2.23) for the second time with the above changes,

$$\int_a^{a+\tau} \int_a^{\xi'} (\xi - a)^3 y^{(3)}(\xi)\, \mathrm{d}\xi \mathrm{d}\xi'$$

$$= \int_a^{a+\tau} (\xi' - a)^3 y^{(2)}(\xi')\, \mathrm{d}\xi' - \int_a^{a+\tau} 3(\xi' - a)^2 y^{(1)}(\xi')\, \mathrm{d}\xi' + \int_a^{a+\tau} 6(\xi' - a) y(\xi')\, \mathrm{d}\xi'$$

$$
\begin{aligned}
& - \int_a^{a+\tau} \int_a^{\xi'} 6y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi' \\
&= (\xi'-a)^3 y^{(1)}(\xi') \,|_a^{a+\tau} - \int_a^{a+\tau} 3(\xi'-a)^2 y^{(1)}(\xi')\,\mathrm{d}\xi' - \Bigg[ 3(\xi'-a)^2 y(\xi') \,|_a^{a+\tau} \\
&\quad - \int_a^{a+\tau} 6(\xi'-a)y(\xi')\,\mathrm{d}\xi' \Bigg] + \int_a^{a+\tau} 6(\xi'-a)y(\xi')\,\mathrm{d}\xi' - \int_a^{a+\tau}\int_a^{\xi'} 6y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi' \\
&= \tau^3 y^{(1)}(a+\tau) - \Bigg[ 3(\xi'-a)^2 y(\xi') \,|_a^{a+\tau} - \int_a^{a+\tau} 6(\xi'-a)y(\xi')\,\mathrm{d}\xi' \Bigg] \\
&\quad - 3\tau^2 y(a+\tau) + \int_a^{a+\tau} 12(\xi'-a)y(\xi')\,\mathrm{d}\xi' - \int_a^{a+\tau}\int_a^{\xi'} 6y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi' \\
&= \tau^3 y^{(1)}(a+\tau) - 6\tau^2 y(a+\tau) + \int_a^{a+\tau} 18(\xi'-a)y(\xi')\,\mathrm{d}\xi' - \int_a^{a+\tau}\int_a^{\xi'} 6y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'
\end{aligned}
\tag{2.24}
$$

As in the previous step, the upper limit is again a 'dummy variable' and now we set $\xi'' = a+\tau$. Integrating again,

$$
\begin{aligned}
& \int_a^{a+\tau}\int_a^{\xi''}\int_a^{\xi'} (\xi-a)^3 y^{(3)}(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\mathrm{d}\xi'' \\
&\quad = \int_a^{a+\tau} (\xi''-a)^3 y^{(1)}(\xi'')\,\mathrm{d}\xi'' - \int_a^{a+\tau} 6(\xi''-a)^2 y(\xi'')\,\mathrm{d}\xi'' + \int_a^{a+\tau}\int_a^{\xi''} 18(\xi'-a)y(\xi')\,\mathrm{d}\xi'\mathrm{d}\xi'' \\
&\qquad - \int_a^{a+\tau}\int_a^{\xi''}\int_a^{\xi'} 6y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\mathrm{d}\xi'' \\
&\quad = \tau^3 y(a+\tau) - \int_a^{a+\tau} 3(\xi''-a)^2 y(\xi'')\,\mathrm{d}\xi'' - \int_a^{a+\tau} 6(\xi''-a)^2 y(\xi'')\,\mathrm{d}\xi''
\end{aligned}
$$

$$+\int_a^{a+\tau}\int_a^{\xi''} 18(\xi'-a)y(\xi')\,\mathrm{d}\xi'\mathrm{d}\xi'' - \int_a^{a+\tau}\int_a^{\xi''}\int_a^{\xi'} 6y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\mathrm{d}\xi''$$

$$= \tau^3 y(a+\tau) - \int_a^{a+\tau} 9(\xi''-a)^2 y(\xi'')\,\mathrm{d}\xi'' + \int_a^{a+\tau}\int_a^{\xi''} 18(\xi'-a)y(\xi')\,\mathrm{d}\xi'\mathrm{d}\xi''$$

$$-\int_a^{a+\tau}\int_a^{\xi''}\int_a^{\xi'} 6y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\mathrm{d}\xi'' \tag{2.25}$$

Now repeating the above procedure for the second term in (2.21) we get the following steps,

$$\int_a^{a+\tau} a_2(\xi-a)^3 y^{(2)}(\xi)\,\mathrm{d}\xi$$

$$= a_2(\xi-a)^3 y^{(1)}(\xi)\mid_a^{a+\tau} - \int_a^{a+\tau} 3a_2(\xi-a)^2 y^{(1)}(\xi)\,\mathrm{d}\xi$$

$$= a_2\tau^3 y^{(1)}(a+\tau) - \left[3a_2(\xi-a)^2 y(\xi)\mid_a^{a+\tau} - \int_a^{a+\tau} 6a_2(\xi-a)y(\xi)\,\mathrm{d}\xi\right]$$

$$= a_2\tau^3 y^{(1)}(a+\tau) - 3a_2\tau^2 y(a+\tau) + \int_a^{a+\tau} 6a_2(\xi-a)y(\xi)\,\mathrm{d}\xi \tag{2.26}$$

Introducing 'dummy variable' and integrating (2.26) again,

$$\int_a^{a+\tau}\int_a^{\xi'} a_2(\xi-a)^3 y^{(2)}(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'$$

$$= \int_a^{a+\tau} a_2(\xi'-a)^3 y^{(1)}(\xi')\,\mathrm{d}\xi' - \int_a^{a+\tau} 3a_2(\xi'-a)^2 y(\xi')\,\mathrm{d}\xi' + \int_a^{a+\tau}\int_a^{\xi'} 6a_2(\xi-a)y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'$$

$$= a_2(\xi'-a)^3 y(\xi')\mid_a^{a+\tau} - \int_a^{a+\tau} 3a_2(\xi'-a)^2 y(\xi')\,\mathrm{d}\xi' - \int_a^{a+\tau} 3a_2(\xi'-a)^2 y(\xi')\mathrm{d}\xi'$$

$$+\int_a^{a+\tau}\int_a^{\xi'} 6a_2(\xi-a)y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'$$

$$=\tau^3 a_2(a+\tau)-\int_a^{a+\tau} 6a_2(\xi'-a)^2 y(\xi')\,\mathrm{d}\xi'+\int_a^{a+\tau}\int_a^{\xi'} 6a_2(\xi-a)y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'$$

finally, integrating for the third time,

$$\begin{aligned}
&\int_a^{a+\tau}\int_a^{\xi''}\int_a^{\xi'} a_2(\xi-a)^3 y^{(2)}(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\mathrm{d}\xi'' \\
&\quad=\int_a^{a+\tau} a_2(\xi''-a)^3 y(\xi'')\,\mathrm{d}\xi''-\int_a^{a+\tau}\int_a^{\xi''} 6a_2(\xi'-a)^2 y(\xi')\,\mathrm{d}\xi'\mathrm{d}\xi'' \\
&\qquad\qquad+\int_a^{a+\tau}\int_a^{\xi''}\int_a^{\xi'} 6a_2(\xi-a)y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\mathrm{d}\xi''
\end{aligned}\tag{2.27}$$

Similar steps are repeated for the third term in (2.21) ,

$$\int_a^{a+\tau} a_1(\xi-a)^3 y^{(1)}(\xi)\,\mathrm{d}\xi$$

$$=a_1(\xi-a)^3 y(\xi)\,\Big|_a^{a+\tau}-\int_a^{a+\tau} 3a_1(\xi-a)^2 y(\xi)\,\mathrm{d}\xi\tag{2.28}$$

$$=a_1\tau^3 y(a+\tau)-\int_a^{a+\tau} 3a_1(\xi-a)^2 y(\xi)\,\mathrm{d}\xi\tag{2.29}$$

Integrating (2.28) twice we get,

$$\int_a^{a+\tau}\int_a^{\xi'} a_1(\xi-a)^3 y^{(1)}(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'=\int_a^{a+\tau} a_1(\xi'-a)^3 y(\xi')\,\mathrm{d}\xi'-\int_a^{a+\tau}\int_a^{\xi'} 3a_1(\xi-a)^2 y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\tag{2.30}$$

and finally,

$$\begin{aligned}\int_a^{a+\tau}\int_a^{\xi''}\int_a^{\xi'} a_1(\xi-a)^3 y^{(1)}(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\mathrm{d}\xi'' &= \int_a^{a+\tau}\int_a^{\xi''} a_1(\xi'-a)^3 y(\xi')\,\mathrm{d}\xi' \\ &\quad - \int_a^{a+\tau}\int_a^{\xi''}\int_a^{\xi'} 3a_1(\xi-a)^2 y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\mathrm{d}\xi''\end{aligned} \tag{2.31}$$

Similarly for the last term of the equation (2.21)

$$\int_a^{a+\tau}\int_a^{\xi''}\int_a^{\xi'} a_0(\xi-a)^3 y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\mathrm{d}\xi'' \tag{2.32}$$

arranging (2.25) to (2.32) together, we end up

$$\begin{aligned}-\tau^3 y(a+\tau) &= \int_a^{a+\tau}\Big[-9(\xi''-a)^2 + a_2(\xi''-a)^3\Big] y(\xi'')\,\mathrm{d}\xi'' \\ &\quad + \int_a^{a+\tau}\int_a^{\xi''}\Big[+18(\xi'-a) - 6a_2(\xi'-a)^2 + a_1(\xi'-a)^3\Big] y(\xi')\,\mathrm{d}\xi'\mathrm{d}\xi'' \\ &\quad + \int_a^{a+\tau}\int_a^{\xi''}\int_a^{\xi'}\Big[-6 + 6a_2(\xi-a) - 3a_1(\xi-a)^2 + a_0(\xi-a)^3\Big] y(\xi)\,\mathrm{d}\xi\mathrm{d}\xi'\mathrm{d}\xi''\end{aligned} \tag{2.33}$$

This can be further simplified by recalling Cauchy formula for repeated integration. Let $f$ be a continuous function on the real line, then the $n$th repeated integral of $f$ based at $a$.

$$f^{(-n)}(x) = \int_a^x \int_a^{\sigma_1} \cdots \int_a^{\sigma_{n-1}} f(\sigma_n)\mathrm{d}\sigma_n \cdots \mathrm{d}\sigma_2\mathrm{d}\sigma_1 \tag{2.34}$$

is given by single integration

$$f^{(-n)}(x) = \frac{1}{(n-1)!}\int_a^x (x-t)^{n-1} f(t)\mathrm{d}t \tag{2.35}$$

let $a+\tau=t$, $\tau=t-a$ and applying the above for repeated integration on (2.33),we get,

$$\begin{aligned}
&-(t-a)^3y(t)\\
&\qquad=\int_a^t\Big[-9(\tau-a)^2+a_2(\tau-a)^3\Big]y(\tau)\,\mathrm{d}\tau\\
&\qquad\quad+\int_a^t(t-\tau)\Big[18(\tau-a)-6a_2(\tau-a)^2+a_1(\tau-a)^3\Big]y(\tau)\,\mathrm{d}\tau\\
&\qquad\quad+\frac{1}{2}\int_a^t(t-\tau)^2\Big[-6+6a_2(\tau-a)-3a_1(\tau-a)^2+a_0(\tau-a)^3\Big]y(\tau)\,\mathrm{d}\tau\\
&\qquad\triangleq\int_a^t K_F(t,\tau)y(\tau)\,\mathrm{d}\tau
\end{aligned}\tag{2.36}$$

with $K_F(t,\tau)$ defined by,

$$\begin{aligned}
K_F(t,\tau)\triangleq&\Big[-9(\tau-a)^2+a_2(\tau-a)^3\Big]+(t-\tau)\Big[18(\tau-a)-6a_2(\tau-a)^2+a_1(\tau-a)^3\Big]\\
&+(t-\tau)^2\Big[-6+6a_2(\tau-a)-3a_1(\tau-a)^2+a_0(\tau-a)^3\Big]
\end{aligned}\tag{2.37}$$

Consider the equation (2.22). Integrating the first term in (2.22) once,

$$\begin{aligned}
&\int_{b-\sigma}^{b}(b-\zeta)^3y^{(3)}(\zeta)\,\mathrm{d}\zeta\\
&\qquad=(b-\zeta)^3y^{(2)}(\zeta)\mid_{b-\sigma}^{b}+\int_{b-\sigma}^{b}3(b-\zeta)^2y^{(2)}(\zeta)\,\mathrm{d}\zeta\\
&\qquad=-\sigma^3y^{(2)}(b-\sigma)+\Big[3(b-\zeta)^2y^{(1)}(\zeta)\mid_{b-\sigma}^{b}+\int_{b-\sigma}^{b}6(b-\zeta)y^{(1)}(\zeta),\mathrm{d}\zeta\Big]
\end{aligned}$$

$$\tag{2.38}$$

$$= -\sigma^3 y^{(2)}(b-\sigma) - 3\sigma^2 y^{(1)}(b-\sigma) + 6(b-\zeta)y^{(1)}(\zeta)\,|_{b-\sigma}^{b} + \int\limits_{b-\sigma}^{b} 6y(\zeta)\,\mathrm{d}\zeta$$

$$= -\sigma^3 y^{(2)}(b-\sigma) - 3\sigma^2 y^{(1)}(b-\sigma) - 6\sigma y(b-\sigma) + \int\limits_{b-\sigma}^{b} 6y(\zeta)\,\mathrm{d}\zeta \tag{2.39}$$

Following the earlier steps i.e, introducing a 'dummy variable' and setting $\zeta' = b - \sigma$ ,

$$\int\limits_{b-\sigma}^{b}\int\limits_{b}^{\zeta'} (b-\zeta)^3 y^{(3)}(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'$$

$$= \int\limits_{b-\sigma}^{b} (b-\zeta')^3 y^{(2)}(\zeta')\,\mathrm{d}\zeta' + \int\limits_{b-\sigma}^{b} 3(b-\zeta')^2 y^{(1)}(\zeta')\,\mathrm{d}\zeta' + \int\limits_{b-\sigma}^{b} 6(b-\zeta')y(\zeta')\,\mathrm{d}\zeta'$$

$$- \int\limits_{b-\sigma}^{b}\int\limits_{b}^{\zeta'} 6y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'$$

$$= (b-\zeta')^3 y^{(1)}(\zeta')\,|_{b-\sigma}^{b} + \int\limits_{b-\sigma}^{b} 3(b-\zeta')^2 y^{(1)}(\zeta')\,\mathrm{d}\zeta' + \Bigg[3(b-\zeta')^2 y(\zeta')\,|_{b-\sigma}^{b}$$

$$+ \int\limits_{b-\sigma}^{b} 6(b-\zeta')y(\zeta')\,\mathrm{d}\zeta'\Bigg] + \int\limits_{b-\sigma}^{b} 6(b-\zeta')y(\zeta')\,\mathrm{d}\zeta' + \int\limits_{b}^{b-\sigma}\int\limits_{b}^{\zeta'} 6y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'$$

$$= -\sigma^3 y^{(1)}(b-\sigma) + \Bigg[3(b-\zeta')^2 y(\zeta')\,|_{b-\sigma}^{b} + \int\limits_{b-\sigma}^{b} 6(b-\zeta')y(\zeta')\,\mathrm{d}\zeta'\Bigg]$$

$$- 3\sigma^2 y(b-\sigma) - \int\limits_{b}^{b-\sigma} 12(b-\zeta')y(\zeta')\,\mathrm{d}\zeta' + \int\limits_{b}^{b-\sigma}\int\limits_{b}^{\zeta'} 6y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'$$

$$
\begin{aligned}
&= -\sigma^3 y^{(1)}(b-\sigma) + \left[ -3\sigma^2 y(b-\sigma) + \int_{b-\sigma}^{b} 6(b-\zeta')y(\zeta')\,\mathrm{d}\zeta' \right] \\
&\qquad - 3\sigma^2 y(b-\sigma) - \int_{b}^{b-\sigma} 12(b-\zeta')y(\zeta')\,\mathrm{d}\zeta' + \int_{b}^{b-\sigma}\int_{b}^{\zeta'} 6y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta' \\
&= -\sigma^3 y^{(1)}(b-\sigma) - 6\sigma^2 y(b-\sigma) - \int_{b}^{b-\sigma} 18(b-\zeta')y(\zeta')\,\mathrm{d}\zeta' \\
&\qquad + \int_{b}^{b-\sigma}\int_{b}^{\zeta'} 6y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'
\end{aligned} \tag{2.40}
$$

Again setting a 'dummy variable' , $\zeta'' = b-\sigma$.Integrate (2.40) third time,

$$
\begin{aligned}
&\int_{b-\sigma}^{b}\int_{b}^{\zeta''}\int_{b}^{\zeta'} (b-\zeta)^3 y^{(3)}(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta'' \\
&\quad = \int_{b-\sigma}^{b} (b-\zeta'')^3 y^{(1)}(\zeta'')\,\mathrm{d}\zeta'' + \int_{b-\sigma}^{b} 6(b-\zeta'')^2 y(\zeta'')\,\mathrm{d}\zeta'' - \int_{b}^{b-\sigma}\int_{b}^{\zeta''} 18(b-\zeta')y(\zeta')\,\mathrm{d}\zeta'\mathrm{d}\zeta'' \\
&\qquad - \int_{b}^{b-\sigma}\int_{b}^{\zeta''}\int_{b}^{\zeta'} 6y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta'' \\
&\quad = \left[ (b-\zeta'')^3 y(\zeta'')\,\Big|_{b-\sigma}^{b} + \int_{b-\sigma}^{b} 3(b-\zeta'')^2 y(\zeta'')\,\mathrm{d}\zeta'' \right] - \int_{b}^{b-\sigma} 6(b-\zeta'')^2 y(\zeta'')\,\mathrm{d}\zeta'' \\
&\qquad - \int_{b}^{b-\sigma}\int_{b}^{\zeta''} 18(b-\zeta')y(\zeta')\,\mathrm{d}\zeta'\mathrm{d}\zeta'' - \int_{b}^{b-\sigma}\int_{b}^{\zeta''}\int_{b}^{\zeta'} 6y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta'' \\
&\quad = -\sigma^3 y(b-\sigma) - \int_{b}^{b-\sigma} 9(b-\zeta'')^2 y(\zeta'')\,\mathrm{d}\zeta'' - \int_{b}^{b-\sigma}\int_{b}^{\zeta''} 18(b-\zeta')y(\zeta')\,\mathrm{d}\zeta'\mathrm{d}\zeta'' \\
&\qquad - \int_{b}^{b-\sigma}\int_{b}^{\zeta''}\int_{b}^{\zeta'} 6y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta''
\end{aligned} \tag{2.41}
$$

The above steps are repeated for all the terms in (2.22),

$$\int_{b-\sigma}^{b} a_2(b-\zeta)^3 y^{(2)}(\zeta)\,\mathrm{d}\zeta$$

$$
\begin{aligned}
&= a_2(b-\zeta)^3 y^{(1)}(\zeta)\mid_{b-\sigma}^{b} + \int_{b-\sigma}^{b} 3a_2(b-\zeta)^2 y^{(1)}(\zeta)\,\mathrm{d}\zeta \\
&= -a_2\sigma^3 y^{(1)}(b-\sigma) + \left[3a_2(b-\zeta)^2 y(\zeta)\mid_{b-\sigma}^{b} + \int_{b-\sigma}^{b} 6a_2(b-\zeta)y(\zeta)\,\mathrm{d}\zeta\right] \\
&= -a_2\sigma^3 y^{(1)}(b-\sigma) - 3a_2\sigma^2 y(b-\sigma) - \int_{b}^{b-\sigma} 6a_2(b-\zeta)y(\zeta)\,\mathrm{d}\zeta
\end{aligned}
\tag{2.42}
$$

Using 'dummy variable' and integrating (2.42),

$$\int_{b-\sigma}^{b}\int_{b}^{\zeta'} a_2(b-\zeta)^3 y^{(2)}(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'$$

$$
\begin{aligned}
&= \int_{b-\sigma}^{b} a_2(b-\zeta')^3 y^{(1)}(\zeta')\,\mathrm{d}\zeta' + \int_{b-\sigma}^{b} 3a_2(b-\zeta')^2 y(\zeta')\,\mathrm{d} - \int_{b-\sigma}^{b}\int_{b} 6a_2(b-\sigma)y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta' \\
&= a_2(b-\zeta')^3 y(\zeta')\mid_{b-\sigma}^{b} + \int_{b-\sigma}^{b} 3a_2(b-\zeta')^2 y(\zeta')\,\mathrm{d}\zeta' - \int_{b}^{b-\sigma} 3a_2(b-\zeta')^2 y(\zeta') \\
&\quad - \int_{b-\sigma}^{b}\int_{b}^{\zeta'} 6a_2(b-\zeta)y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta' \\
&= -\sigma^3 a_2(b-\sigma) - \int_{b}^{b-\sigma} 6a_2(b-\zeta')^2 y(\zeta')\,\mathrm{d}\zeta' - \int_{b-\sigma}^{b}\int_{b}^{\zeta'} 6a_2(b-\zeta)y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'
\end{aligned}
\tag{2.43}
$$

and for the third time,

$$\int_{b-\sigma}^{b}\int_{b}^{\zeta''}\int_{b}^{\zeta'} a_2(b-\zeta-a)^3 y^{(2)}(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta''$$

$$= \int_{b-\sigma}^{b} a_2(b-\zeta'')^3 y(\zeta'')\,\mathrm{d}\zeta'' + \int_{b-\sigma}^{b}\int_{b}^{\zeta''} 6a_2(b-\zeta')^2 y(\zeta')\,\mathrm{d}\zeta'\mathrm{d}\zeta''$$
$$+ \int_{b-\sigma}^{b}\int_{b}^{\zeta''}\int_{b}^{\zeta'} 6a_2(b-\zeta)y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta''$$
$$= -\int_{b}^{b-\sigma} a_2(b-\zeta'')^3 y(\zeta'')\,\mathrm{d}\zeta'' - \int_{b}^{b-\sigma}\int_{b}^{\zeta''} 6a_2(b-\zeta')^2 y(\zeta')\,\mathrm{d}\zeta'\mathrm{d}\zeta'' -$$
$$\int_{b}^{b-\sigma}\int_{b}^{\zeta''}\int_{b}^{\zeta'} 6a_2(b-\zeta)y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta'' \tag{2.44}$$

$$\int_{b-\sigma}^{b} a_1(b-\zeta)^3 y^{(1)}(\zeta)\,\mathrm{d}\zeta$$
$$= a_1(b-\zeta)^3 y(\zeta)\,\big|_{b-\sigma}^{b} + \int_{b-\sigma}^{b} 3a_1(b-\zeta)^2 y^{(2)}(\zeta)\,\mathrm{d}\zeta$$
$$= -a_1\sigma^3 y(b-\sigma) - \int_{b}^{b-\sigma} 3a_1(b-\zeta)^2 y^{(2)}(\zeta)\,\mathrm{d}\zeta \tag{2.45}$$

Integrating (2.45) second time,

$$\int_{b-\sigma}^{b}\int_{b}^{\zeta'} a_1(b-\zeta)^3 y^{(1)}(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta' = \int_{b-\sigma}^{b} a_1(b-\zeta)^3 y(\zeta')\,\mathrm{d}\zeta' - \int_{b-\sigma}^{b}\int_{b}^{\zeta'} 3a_1(b-\zeta)^2 y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta' \tag{2.46}$$

$$= -\int_{b}^{b-\sigma} a_1(b-\zeta)^3 y(\zeta')\,\mathrm{d}\zeta' + \int_{b}^{b-\sigma}\int_{b}^{\zeta'} 3a_1(b-\zeta)^2 y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta' \tag{2.47}$$

Integrating (2.46) the final time,

$$\begin{aligned}
&\int_{b-\sigma}^{b}\int_{b}^{\zeta''}\int_{b}^{\zeta'} a_1(b-\zeta)^3 y^{(1)}(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta'' \\
&\qquad = \int_{b-\sigma}^{b}\int_{b}^{\zeta''} a_1(b-\zeta')^3 y(\zeta')\,\mathrm{d}\zeta' + \int_{b-\sigma}^{b}\int_{b}^{\zeta''}\int_{b}^{\zeta'} 3a_1(b-\zeta)^2 y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta'' \\
&\qquad = -\int_{b}^{b-\sigma}\int_{b}^{\zeta''} a_1(b-\zeta')^3 y(\zeta')\,\mathrm{d}\zeta' - \int_{b}^{b-\sigma}\int_{b}^{\zeta''}\int_{b}^{\zeta'} 3a_1(b-\zeta)^2 y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta''
\end{aligned} \tag{2.48}$$

the last term in (2.22)

$$\int_{b-\sigma}^{b}\int_{b}^{\zeta''}\int_{b}^{\zeta'} a_0(b-\zeta)^3 y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta'' = -\int_{b}^{b-\sigma}\int_{b}^{\zeta''}\int_{b}^{\zeta'} a_0(b-\zeta)^3 y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta'' \tag{2.49}$$

all the integrated terms from (2.40) to (2.49) together form:

$$\begin{aligned}
\sigma^3 y(b-\sigma) = &\int_{b}^{b-\sigma}\Big[-9(b-\zeta'')^2 - a_2(b-\zeta'')^3\Big] y(\zeta'')\,\mathrm{d}\zeta'' \\
&+ \int_{b}^{b-\sigma}\int_{b}^{\zeta''}\Big[-18(b-\zeta') - 6a_2(b-\zeta')^2 - a_1(b-\zeta')^3\Big] y(\zeta')\,\mathrm{d}\zeta'\mathrm{d}\zeta'' \\
&+ \int_{b}^{b-\sigma}\int_{b}^{\zeta''}\int_{b}^{\zeta'}\Big[-6 - 6a_2(b-\zeta) - 3a_1(b-\zeta)^2 - a_0(b-\zeta)^3\Big] y(\zeta)\,\mathrm{d}\zeta\mathrm{d}\zeta'\mathrm{d}\zeta''
\end{aligned} \tag{2.50}$$

let $b-\sigma = t$ and $\sigma = b-t$ , applying the Cauchy formula for repeated integration as stated earlier,

$$
\begin{aligned}
(b-t)^3 y(t) = &\int_b^t \left[ -9(b-\sigma)^2 - a_2(b-\sigma)^3 \right] y(\sigma)\,\mathrm{d}\sigma \\
&+ \int_b^t (t-\sigma)\left[ -18(b-\sigma) - 6a_2(b-\sigma)^2 - a_1(b-\sigma)^3 \right] y(\sigma)\,\mathrm{d}\sigma \\
&+ \frac{1}{2}\int_b^t (t-\sigma)^2\left[ -6 - 6a_2(b-\sigma) - 3a_1(b-\sigma)^2 - a_0(b-\sigma)^3 \right] y(\sigma)\,\mathrm{d}\sigma
\end{aligned}
\tag{2.51}
$$

flipping the limits on integrals (2.51) can be written as,

$$
\begin{aligned}
(b-t)^3 y(t) = &-\int_t^b \left[ -9(b-\tau)^2 - a_2(b-\tau)^3 \right] y(\tau)\,\mathrm{d}\tau \\
&- \int_t^b (t-\tau)\left[ -18(b-\tau) - 6a_2(b-\tau)^2 - a_1(b-\tau)^3 \right] y(\tau)\,\mathrm{d}\tau \\
&- \frac{1}{2}\int_t^b (t-\tau)^2\left[ -6 - 6a_2(b-\tau) - 3a_1(b-\tau)^2 - a_0(b-\tau)^3 \right] y(\tau)\,\mathrm{d}\tau \\
\triangleq &\int_b^t K_B(t,\tau) y(\tau)\mathrm{d}\tau
\end{aligned}
\tag{2.52}
$$

where $K_B(t,\tau)$ defined as,

$$
\begin{aligned}
K_B(t,\tau) \triangleq &\left[ 9(b-\tau)^2 + a_2(b-\tau)^3 \right] \\
&+ (t-\tau)\left[ 18(b-\tau) + 6a_2(b-\tau)^2 + a_1(b-\tau)^3 \right] \\
&+ \frac{1}{2}(t-\tau)^2\left[ 6 + 6a_2(b-\tau) + 3a_1(b-\tau)^2 + a_0(b-\tau)^3 \right]
\end{aligned}
\tag{2.53}
$$

‘ *The forward*’ & ‘*backward*’ partial kernels are redefined as

$$
\begin{aligned}
K_F(t,\tau) \triangleq \mu(\tau-a)&\Big[9(\tau-a)^2 - a_2(\tau-a)^3\Big] \\
&+ (t-\tau)\Big[-18(\tau-a) + 6a_2(\tau-a)^2 - a_1(\tau-a)^3\Big] \\
&+ \frac{1}{2}(t-\tau)^2\Big[6 - 6a_2(\tau-a) + 3a_1(\tau-a)^2 - a_0(\tau-a)^3\Big]
\end{aligned} \tag{2.54}
$$

$$
\begin{aligned}
K_B(t,\tau) \triangleq \mu(b-\tau)&\Big[9(b-\tau)^2 + a_2(b-\tau)^3\Big] \\
&+ (t-\tau)\Big[18(b-\tau) + 6a_2(b-\tau)^2 + a_1(b-\tau)^3\Big] \\
&+ \frac{1}{2}(t-\tau)^2\Big[6 + 6a_2(b-\tau) + 3a_1(b-\tau)^2 + a_0(b-\tau)^3\Big]
\end{aligned} \tag{2.55}
$$

where

$$\mu(\tau-a) = \begin{cases} 1 & : \tau \geq a \\ 0 & : \tau < a \end{cases}$$

and

$$\mu(b-\tau) = \begin{cases} 1 & : \tau \leq b \\ 0 & : \tau > b \end{cases}$$

The equations (2.54) and (2.55) can now be expressed simply as :

$$(t-a)^3 y(t) = \int_a^t K_F(t,\tau) y(\tau)\,\mathrm{d}\tau \tag{2.56}$$

$$(b-t)^3 y(t) = \int_t^b K_B(t,\tau) y(\tau)\,\mathrm{d}\tau \tag{2.57}$$

Hence,

$$K_{DS}(t,\tau) \triangleq \begin{cases} K_F(t,\tau) & : \tau \leq t \\ K_B(t,\tau) & : \tau > t \end{cases} \tag{2.58}$$

Combining (2.57) and (2.58) and dividing both sides by $[(t-a)^3+(b-t)^3]$ results in:

$$y(t) = \frac{1}{[(t-a)^3+(b-t)^3]} \int\limits_a^b K_{DS}(t,\tau)y(\tau)\,\mathrm{d}\tau \tag{2.59}$$

The recursive expressions of the derivatives can be derived by following similar steps as in the derivation of $K_{DS}$. In order to obtain the expression for $y^{(1)}(t)$ (2.21) and (2.22) are integrated twice.

$$\begin{aligned}(t-a)^3y^{(1)}(t) = {}& 6(t-a)^2y(t) - a_2(t-a)^3y(t)\\ &+ \int_t^a \Big[-18(\tau-a) + 6a_2(\tau-a)^2 - a_1(\tau-a)^3\Big]y(\tau)\mathrm{d}\tau\\ &+ \int_t^a (t-\tau)\Big[6 - 6a_2(\tau-a)^2 - 3a_1(\tau-a)^2 - a_0(\tau-a)^3\Big]y(\tau)\mathrm{d}\tau\end{aligned} \tag{2.60}$$

and

$$\begin{aligned}(b-t)^3y^{(1)}(t) = {}& -6(b-t)^2y(t) - a_2(b-t)^3y(t)\\ &+ \int_t^b \Big[18(b-\tau) + 6a_2(b-\tau)^2 + a_1(b-\tau)^3\Big]y(\tau)\mathrm{d}\tau\\ &+ \int_t^b (t-\tau)\Big[6 + 6a_2(b-\tau)^2 + 3a_1(b-\tau)^2 + a_0(b-\tau)^3\Big]y(\tau)\mathrm{d}\tau\end{aligned} \tag{2.61}$$

$y^{(1)}(t)$ is obtained by adding (2.60) and (2.61) and dividing by $[(t-a)^3+(b-t)^3]$.
For $y^{(2)}(t)$, (2.21) and (2.22) are integrated only once , hence resulting in

$$\begin{aligned}(t-a)^3y^{(2)}(t) = {}& 3(t-a)^2y^{(1)}(t) - a_2(t-a)^3y^{(1)}(t)\\ &- 6(t-a)y(t) + 3a_2(t-a)^2y(t) - a_1(t-a)^3y(t)\\ &+ \int_t^a \Big[6 - 6a_2(\tau-a) + 3a_1(\tau-a)^2 - a_0(\tau-a)^3\Big]y(\tau)\mathrm{d}\tau\end{aligned} \tag{2.62}$$

and

$$
\begin{aligned}
(b-t)^3 y^{(2)}(t) = & -3(b-t)^2 y^{(1)}(t) - a_2(b-t)^3 y^{(1)}(t) \\
& - 6(b-t)y(t) - 3a_2(b-t)^2 y(t) - a_1(b-t)^3 y(t) \\
& + \int_t^b \left[6 + 6a_2(b-\tau) + 3a_1(b-\tau)^2 + a_0(b-\tau)^3\right] y(\tau)\mathrm{d}\tau
\end{aligned}
\tag{2.63}
$$

$y^{(2)}(t)$ is also obtained by following similar steps as above and hence adding (2.62) and (2.63) and also dividing by $[(t-a)^3 + (b-t)^3]$.

# Chapter 3

# Overview of Different Parameter Estimation Methods Using the Double-Sided Kernel Approach

In this chapter, we study and understand the previous works done on parameter and state estimation using double sided kernel approach.

## 3.1 Parameter Estimation Using L-splines in RKHS [25] [26]

Consider a general third order system below:

$$y^{(3)}(t) + a_2 y^{(2)}(t) + a_1 y^{(1)}(t) + a_0 y(t) = 0 \tag{3.1}$$

We want to estimate the parameter vector $a \triangleq (a_0, a_1, a_2)$ of the above system from noisy observation $z(\tau); \tau \in [a, b]$ of the function $y(\tau); \tau \in [a, b]$. The expression

$$L(y) = \int_a^b K_{DS}^a(t, \tau) y(\tau) d\tau \tag{3.2}$$

in which the kernel $K_{DS}$ is dependent on the unknown parameter vector $a$ represents an operator which maps the function $y \in L^2[a, b]$ into the RKHS with a positive symmetric kernel defined as

$$K^a(t_1, t_2) = \int_a^b K^a_{DS}(t_1, \tau) K^a_{DS}(t_2, \tau) d\tau \tag{3.3}$$

The "noisy measurement function" $z(t)$,$t \in [a, b]$ is projected onto the subspace in RKHS [26] defined by,

$$S^a \triangleq \{y \in Ker^{\perp}L \mid s.t\ y(t) = \int_a^b K^a_{DS}(t, z) y(z) dz\} \tag{3.4}$$

where, $Ker^{\perp}L$ denotes the orthogonal compliment to the null space of the operator $L$ in (3.2). Since we are looking for projections $\hat{y} \in Ker^{\perp}L$ , their approximate expressions as finite linear combinations of the vectors in the subspace span$\{K(t, .), t \in [a, b]\}$, or span$\{K_{DS}(t, .), t \in [a, b]\}$ needs to be found out. These finite linear combinations will have the following form:

$$\hat{y}(.) \cong \sum_{j=1}^{n} c_j K^a_{DS}(t_j, \cdot) \tag{3.5}$$

for some distinct time instants $t_i \in [a, b], i = 1, ..., n$ chosen in such a way that, $K^a_{DS}(t_j, \cdot)$ are linearly independent for $t_j, j = 1, ..., n$ for distinct $t_j$.The linear combination in (3.5) should approximately satisfy the reproducing equation below:

$$\hat{y}(t) = \int_a^b K^a_{DS}(t, \tau) \hat{y}(\tau) d\tau \tag{3.6}$$

The approximation should satisfy the above equation
By substituting (3.5) in (3.6) we get,

$$\begin{aligned}
\sum_{j=1}^{n} c_j K^a_{DS}(t_j, t) &= \int_a^b K^a_{DS}(t, \tau) \sum_{j=1}^{n} c_j K^a_{DS}(t_j, \tau) d\tau \\
&= \sum_{j=1}^{n} c_j \int_a^b K^a_{DS}(t, \tau) K^a_{DS}(t_j, \tau) d\tau \\
&= \sum_{j=1}^{n} c_j K^a(t, t_j) \\
&= \sum_{j=1}^{n} c_j K^a(t_j, t)
\end{aligned} \tag{3.7}$$

As RKHS kernels are symmetric we can re-write the above equation as,

$$\sum_{j=1}^{n} c_j[K_{DS}^a - K^a](t_j, t) = 0 \tag{3.8}$$

or, equivalently,

$$c^T vec\{[K_{DS}^a - K^a](t_j, t)\}_{j=1}^n = 0 \tag{3.9}$$

with $c^T \triangleq [c_1, ..., c_n]$,

$$vec\{[K_{DS}^a - K^a](t_j, t)\}_{j=1}^n \triangleq [(K_{DS}^a - K^a)(t_1, t), ..., (K_{DS}^a - K^a)(t_n, t)]^T$$

The projection has to minimize

$$min\{\frac{1}{2}\|z - \hat{y}\|^2 \ ; c_j, j = 1, \cdots, n\} \tag{3.10}$$

subject to (3.8), where $z$ represents the noisy observation. The norm in (3.10) must be computed in the space $L^2[a, b]$. Recalling that the kernels $K_{DS}^a$ and $K^a$ both depend on the parameter vector $a$, then to estimate the system parameter vector $a$ the arg min solution in (3.10) of the above must further (or simultaneously ) be minimized w.r.t. $a$. For simplicity, the dependence of both $K_{DS}$ and $K_a$ is dropped from the notation in the sequel.
The cost function in (3.10) can be expressed as:

$$\frac{1}{2}\|z - \hat{y}\|^2 = \frac{1}{2}\langle z - \hat{y}|z - \hat{y}\rangle = \frac{1}{2}\|z\|^2 - \langle z|\hat{y}\rangle + \frac{1}{2}\|\hat{y}\|^2 \tag{3.11}$$

in which the scalar product is in the space $L^2[a, b]$. Using the approximation $\hat{y}$ in (3.5)

$$\begin{aligned} \langle z|\hat{y}\rangle &= \langle z(\cdot)| \sum_{j=1}^{n} c_j K_{DS}(t_j, \cdot)\rangle \\ &= \sum_{j=1}^{n} c_j \langle z(\cdot)|K_{DS}(t_j, \cdot)\rangle \\ &= c^T vec\{\langle z(\cdot)|K_{DS}(t_j, \cdot)\rangle\}_{j=1}^n \end{aligned} \tag{3.12}$$

and

$$\begin{aligned}\|\hat{y}\|^2 &= \langle\hat{y}|\hat{y}\rangle = \langle\sum_{j=1}^{n} c_j K_{DS}(t_j,\cdot)|\sum_{l=1}^{n} c_l K_{DS}(t_l,\cdot)\rangle \\ &= \sum_{j=1}^{n}\sum_{l=1}^{n} c_j c_l \langle K_{DS}(t_j,\cdot)|K_{DS}(t_l,\cdot)\rangle \\ &= \sum_{j=1}^{n}\sum_{l=1}^{n} c_j c_l K(t_j,t_l) \\ &= c^T\, matrix\{K(t_j,t_l)\}_{j,l=1}^{n,n}\, c\end{aligned} \tag{3.13}$$

The constraint (3.8) can be simplified, requiring that it holds point-wise at $t_l, l = 1, ..., m$, (for some selected integer $m < n$) so that (3.8) is written as

$$matrix\{[K_{DS} - K](t_j,t_l)\}_{j,l=1}^{n,m}\, c = 0 \tag{3.14}$$

Combining (3.13) and (3.12) , the fundamental optimization problem in RKHS is re-stated as:

$$\min_{c,a}\{\frac{1}{2}\|z\|^2 + \frac{1}{2}\|\hat{y}\|^2 - \langle z|\hat{y}\rangle = \min_{c,a}\{\frac{1}{2}\|z\|^2 + \frac{1}{2}c^T M^a(t_j,t_l)c - c^T V(z)\} \tag{3.15}$$

subject to the constraint $N^a(t_j,t_l)c = 0$ or $c \in Ker\{N^a(t_j,t_l)\}$ where

$$\begin{aligned}M^a(t_j,t_l) &\triangleq matrix\{K(t_j,t_l)\}_{j,l=1}^{n,n} \\ V(z) &\triangleq vec\{\langle z(\cdot)|K_{DS}(t_j,\cdot)\rangle\}_{j=1}^{n} \\ N^a(t_j,t_l) &\triangleq matrix\{(K_{DS} - K)(t_j,t_l)\}_{j,l=1}^{n,m}\end{aligned}$$

It should be noted that the stationary point with respect to $c$ of the unconstrained minimization problem in (3.15) is derived from the calculation of the gradient w.r.t $c$ of the cost function in (3.15). The equation for the stationary point, which is called the *normal equation for the solution of the least squares problem* is hence obtained as:

$$M^a(t_j,t_l)c - V(z) = 0 \tag{3.16}$$

and yields the stationary point:

$$c = [M^a(t_j, t_l)]^{-1} V(z) \tag{3.17}$$

In order to express $K(t_j, t_l), K_{DS}$ as linear functions of $a_0, a_1, a_3$, we consider the actual expression for $K_{DS}(t_j, \tau)$

$$\begin{aligned} K_{DS}(t_j, \tau) &= f_3(t_j, \tau) + a_2 f_2(t_j, \tau) + a_1 f_1(t_j, \tau) + a_0 f_0(t_j, \tau) \\ \langle z | K_{DS}(t_j, .) \rangle &= \langle z | f_3(t_j, .) \rangle + \langle z | a_2 f_2(t_j, .) \rangle \\ &\quad + \langle z | a_1 f_1(t_j, .) \rangle + \langle z | a_0 f_0(t_j, .) \rangle \end{aligned} \tag{3.18}$$

while

$$K(t_j, t_l) = \langle K_{DS}(t_j, .) | K_{DS}(t_l, .) = \sum_{k=1}^{3} \sum_{i=1}^{n} a_k a_i \langle f_k(t_j, .) | f_i(t_l, .) \rangle$$

in which $a_3$ is set to 1. It is important to note that the matrix $M^a(t_j, t_l)$ is symmetric and non negative definite by the properties of the RKHS kernel $K(t_j, t_l)$, so the minimization problem is convex in $c$.

### 3.1.1 RKHS approach No 1 with explicit null space constraint [16]

With reference to a SISO system of order three, constrained optimization of the cost function derived in the previous section is attempted, meaning the cost function is to be minimized with respect to both $a := [a_0, a_1, a_2]$ and $c$. The cost function, constraints and feasible set $F$ are as below:

$$\min\{J(a, c) \mid a \in \mathbb{R}^3; \quad c \in \mathbb{R}^n; \text{ subject to } \; N(a)c = 0\} \tag{3.19}$$

$$J(a, c) = \frac{1}{2} ||z||^2_{L_2[a,b]} - c^T V(a, z) + \frac{1}{2} c^T M(a) c \tag{3.20}$$

Constraints and feasible set $F$:

$$N(a)c = 0 \quad ; N(a) \in \mathbb{R}^{m \times n} \tag{3.21}$$

$$F := \{(a, c) \in \mathbb{R}^{3+n} \mid N(a)c = 0\} \tag{3.22}$$

In this section we use the notations $M(a)$,$N(a)$,$V(a,z)$ instead of $M^a(t_j,t_l)$ , $N^a(t_j,t_l)$ and $V(z)$ in section 3.1.
Constrained non-linear optimization algorithms such as 'sequential-quadratic programming' and 'active-set' were attempted on the above. However, these failed to provide a satisfactory result as the convergence rate of $a$ and $c$, were found to be vastly different. Hence there was a need to develop other optimization routines and modifying the problem as we proceeded.

### 3.1.2 RKHS approach No. 2 with implicit null space constraint [16]

In this approach the idea is to eliminate $c$ , to simplify the computational approach 1. To this effect , we represent the solution of by using the null space of $N(a)$. If $L(a)$ denotes the matrix whose columns are null space vectors of $N(a)$, then $c$ can be expressed as linear combinations of these columns as follows.

$$c = L(a)g \quad ; g \in \mathbb{R}^m; L(a) \in \mathbb{R}^{n\times m} \tag{3.23}$$

By substituting (3.23) into (3.20) we obtain an equivalent unconstrained minimization problem with respect to $(a,g) \in \mathbb{R}^{3+m}$:

$$\min\{J(a,g) \mid a \in \mathbb{R}^3; \quad g \in \mathbb{R}^n\} \tag{3.24}$$

$$J(a,g) := \frac{1}{2}||z||^2_{L_2[a,b]} - g^T L^T(a)V(a,z) + \frac{1}{2}g^T L(a)^T M(a)L(a)g \tag{3.25}$$

The necessary conditions for optimality of the pair of vectors $(a,g)$ in (3.24) are:

$$\frac{\partial J}{\partial a_i}(a,g) = 0 \; ; \quad i = 1,2 \tag{3.26}$$

$$\frac{\partial J}{\partial g}(a,g) = 0 \tag{3.27}$$

If we could find the solution of (3.26) - (3.27) we would have the exact minimizer $(a,g)$ and hence the optimal $a$.

The explicit formulae for the left hand sides of (3.26) - (3.27) are derived as follows. The dependence on parameter vector $a$ is omitted for brevity.

$$
\begin{aligned}
\frac{\partial J}{\partial a_i} &= -\frac{d[g^T L^T V]}{da_i} + \frac{1}{2}\frac{d[g^T L^T M L g]}{da_i} = -g^T \frac{d[L^T V]}{da_i} + \frac{1}{2} g^T \frac{d[L^T M L]}{da_i} g \\
&= -g^T \left[ \frac{d[L^T]}{da_i} V + L^T \frac{dV}{da_i} \right] + \frac{1}{2} g^T \left[ \frac{d[L^T]}{da_i} M L + L^T \frac{dM}{da_i} L + L^T M \frac{dL}{da_i} \right] g \\
&= -g^T \left\{ \left[ \frac{dL}{da_i} \right]^T V + L^T \frac{dV}{da_i} \right\} \\
&\qquad + \frac{1}{2} g^T \left\{ \left[ \frac{dL}{da_i} \right]^T M L + L^T \frac{dM}{da_i} L + L^T M \frac{dL}{da_i} \right\} g \ ; \quad i = 1, 2
\end{aligned}
\tag{3.28}
$$

$$
\frac{\partial J}{\partial g} = -\frac{d[g^T L^T V]}{dg} + \frac{1}{2}\frac{d[g^T L^T M L g]}{dg} = -V^T L + \frac{1}{2} g^T [L^T M L + L^T M^T L] \tag{3.29}
$$

Since $M$ is symmetric we have $M = M^T$ so

$$
\frac{\partial J}{\partial g} = -V^T L + g^T L^T M L \tag{3.30}
$$

Now (3.27) is equivalent to requesting that

$$
\begin{aligned}
&- V^T L + g^T L^T M L = 0 \quad \text{i.e. } L^T M L g - L^T V = 0 \\
&\text{equivalently} \quad g = \left[ L^T M L \right]^{-1} L^T V
\end{aligned}
\tag{3.31}
$$

Substituting (3.31) into (3.28) yields the gradient of $J$ with respect to $a$; component-wise calculated as

$$
\begin{aligned}
\frac{\partial J(a)}{\partial a_i} &= -V^T L \left[ L^T M L \right]^{-1} \left\{ \left[ \frac{dL}{da_i} \right]^T V + L^T \frac{dV}{da_i} \right\} \\
&\qquad + \frac{1}{2} V^T L \left[ L^T M L \right]^{-1} \left\{ \left[ \frac{dL}{da_i} \right]^T M L + L^T \frac{dM}{da_i} L + L^T M \frac{dL}{da_i} \right\} \left[ L^T M L \right]^{-1} L^T V
\end{aligned}
$$

for i=1,2, because for any matrix $U$, we have $[U^{-1}]^T = [U^T]^{-1}$ and because $L^T ML$ is symmetric. The above can be calculated by evaluating the following matrix sensitivity matrices by central finite differences:

$$\frac{dL}{da_i} \ ; \quad \frac{dV}{da_i} \ ; \quad \frac{dM}{da_i} \ ; \quad i = 1, 2 \tag{3.32}$$

This approach like the previous one did not yield an accurate estimation of parameter vector $a$.

### 3.1.3 RKHS approach No 3 using penalty term [16]

An alternative idea would be to solve the penalized problem (which uses penalization as a "soft" rather than "hard" constraint), by introducing a penalty of cost $\beta > 0$.

$$\min\{J_P(a, c, \beta) \mid a \in \mathbb{R}^3; \quad c \in \mathbb{R}^n\} \tag{3.33}$$

$$\begin{aligned} J_P &:= \frac{1}{2}||z||^2_{L_2[a,b]} - c^T V(a, z) + \frac{1}{2}c^T M(a)c + \frac{\beta}{2}c^T N(a)^T N(a)c \\ &= \frac{1}{2}||z||^2_{L_2[a,b]} - c^T V(a, z) + \frac{1}{2}c^T [M(a) + \beta N(a)^T N(a)]c \end{aligned}$$

The necessary optimality conditions in this case are

$$\frac{\partial J_P}{\partial a_i}(a, c) = 0 \ ; \quad i = 1, 2 \tag{3.34}$$

$$\frac{\partial J_P}{\partial c}(a, c) = 0 \tag{3.35}$$

Defining

$$M_{\beta N}(a, \beta) := M(a) + \beta N(a)^T N(a)$$

Note that $M_{\beta N}$ is symmetric and invertible for any choice of $\beta > 0$, the left hand sides of the necessary conditions (3.34) - (3.35) translate into

$$\frac{\partial J_P}{\partial a_i} = -c^T \frac{dV}{da_i} + \frac{1}{2}c^T \frac{dM_{\beta N}}{da_i} c \ ; \quad i = 1, 2 \tag{3.36}$$

$$\frac{\partial J_P}{\partial c} = -V^T + c^T M_{\beta N} \tag{3.37}$$

which can easily be solved for $c$ yielding

$$c = M_{\beta N}^{-1} V$$

which upon substitution into (3.36) would deliver the gradient with respect to $a$ alone:

$$\frac{\partial J_P}{\partial a_i} = -V^T M_{\beta N}^{-1} \frac{dV}{da_i} + \frac{1}{2} V^T M_{\beta N}^{-1} \frac{dM_{\beta N}}{da_i} M_{\beta N}^{-1} V \ ; \quad i = 1, 2 \tag{3.38}$$

The latter “dimensional reduction trick” whereby the variable $c$ is effectively removed from the optimization procedure may not be practical if the size of the gradient (3.37) is several orders larger than that of (3.36) (system insensitivity to variation of the parameter $a$). In such an event, the optimization is better performed jointly with respect to both $a$ and $c$.

Also the penalty approach enables one to compute $\frac{dM_{\beta N}}{da_i}$ , analytically as it is a quadratic form in $a$.

The value of the penalty parameter $\beta > 0$ should be larger at the beginning, say $\beta = 10$, and be decreased as the algorithm progresses in a commensurate way with the size of the gradient (3.38). It should thus reduce to $\beta = 0$, so in the end, the penalized and unconstrained problems become identical. Hopefully, such a gradual decrease can be done slowly enough for the penalized solution to converges to the unconstrained one $(a, c)$ with the true $a$ and $c$.

Finally, since the gradient w.r.t. $a$, (3.36), is easiest to obtain by way of central finite difference approximation, the penalty term can be conveniently substituted by

$$\beta||\sum_{i=1}^{n} c_j K_{DS}(t_j,\cdot) - \sum_{i=1}^{n} c_j K_A^{int}(t_j,\cdot)||^2_{L_2[a,b]} = \beta||\sum_{i=1}^{n} c_j[K_{DS}(t_j,\cdot) - K_A^{int}(t_j,\cdot)]||^2_{L_2[a,b]}$$
$$= \beta \int_a^b [D(\tau)c]^T D(\tau)c d\tau = c^T \int_a^b D(\tau)^T D(\tau) d\tau \ c \tag{3.39}$$
$$K_A^{int}(t_j,\cdot) := \int_a^b K_{DS}(t_j;\zeta)K_{DS}(\cdot,\zeta)d\zeta$$
$$\text{with} \quad D(\tau) := [K_{DS}(t_1,\tau) - K_A^{int}(t_1,\tau),\cdots,K_{DS}(t_n,\tau) - K_A^{int}(t_n,\tau)]^T \ ;$$

which does not complicate much such a gradient estimation. The modification of the matrix $M(a)$ is now of integral matrix form, just like $M(a)$ itself. The additional advantage is that selection of "interpolation knots" (the $\tau_i$, $i = 1,\cdots,m$) as well as their number are not needed in (3.39).
A basic gradient (steepest descent) algorithm utilizing Armijo step size can be used to optimize all the above mentioned constrained optimization methods, as they are explicitly based on calculating the gradient of the cost functions w.r.t. the optimizing parameters. The suggested optimization algorithms are Gradient, Armijo step size and Lagrange multiplier algorithms:

## 3.2 Two-step Non-asymptotic Parameter and State Estimation of LTI SISO Systems on Finite Interval [27] [28]

There is another way to "identify" the true parameter $a$ in terms of a simple unconstrained minimization that aims at making $V(a,z) = y$ (refer section 3.1 and Theorem 1). If exact matching is required only at a finite number $n$ of discrete time points $t_j; j = 1,\cdots,n$ then the problem amounts to finding the optimal solution to:

$$\min\{J(a) := \frac{1}{2n}\sum_{i=1}^{n}(y(t_i) - < y, K_{DS}(t_i,\cdot) >_2)^2 \mid \text{w.r.t. } a \in \mathbb{R}^3\} \qquad (3.40)$$
$$= \min\{\frac{1}{2n}\sum_{i=1}^{n}\left[y(t_i) - \int_a^b K_{DS}(t_i,\tau)y(\tau)d\tau\right]^2 \mid \text{w.r.t. } a \in \mathbb{R}^3\}$$

A continuous time version of the above becomes

$$\min\{\frac{1}{2T}\int_a^b\left[y(t) - \int_a^b K_{DS}(t,\tau)y(\tau)d\tau\right]^2 dt \mid \text{w.r.t. } a \in \mathbb{R}^3\} \qquad (3.41)$$

with $T := b - a$.

The cost function in (3.41) can be calculated as follows. $K_{DS}(t,\tau)$ is expressed as a scalar product of some partial kernels,(here fourth order system is considered)

$$K_{DS}(t,\tau) = K_v(t,\tau)^T a + k_{v5}(t,\tau) \text{ equivalently } K_{DS}(t,\tau) = a^T K_v(t,\tau) + k_{v5}(t,\tau) \qquad (3.42)$$

$$K_v(t,\tau)^T := [k_{v1}(t,\tau), k_{v2}(t,\tau), k_{v3}(t,\tau), k_{v5}(t,\tau)];\ a := [a_0, a_1, a_2, a_3]^T \qquad (3.43)$$

Substituting the above into the cost of (3.41) yields (with $T := b - a$)

$$J(a) := \frac{1}{2T}\int_a^b\left[y(t) - \int_a^b K_{DS}(t,\tau)\ y(\tau)d\tau\right]^2 dt$$
$$= \frac{1}{T}\int_a^b\left[\frac{1}{2}y(t)^2 - y(t)\int_a^b K_{DS}(t,\tau)\ y(\tau)d\tau + \frac{1}{2}\left(\int_a^b K_{DS}(t,\tau)\ y(\tau)d\tau\right)^2\right] dt$$
$$= \frac{1}{T}\int_a^b\left[\frac{1}{2}y(t)^2 - y(t)\int_a^b K_{DS}(t,\tau)\ y(\tau)d\tau + \frac{1}{2}\int_a^b K_{DS}(t,\tau)\ y(\tau)d\tau\int_a^b K_{DS}(t,s)\ y(s)ds\right] dt$$
$$= \frac{1}{2T}\int_a^b y(t)^2 dt - \frac{1}{T}\int_a^b\int_a^b K_{DS}(t,\tau)\ y(\tau)y(t)\ d\tau\ dt$$
$$+ \frac{1}{2T}\int_a^b\int_a^b\int_a^b K_{DS}(t,\tau)K_{DS}(t,s)\ y(\tau)y(s)\ d\tau\ ds\ dt$$

Hence

$$
\begin{aligned}
&J(a) = \\
&= \frac{1}{2T}\int_a^b y(t)^2 dt - \frac{1}{T}\int_a^b \int_a^b [K_v(t,\tau)^T a + k_{v5}(t,\tau)]\ y(\tau)y(t)\ d\tau\ dt \\
&\quad + \frac{1}{2T}\int_a^b \int_a^b \int_a^b [a^T K_v(t,\tau) + k_{v5}(t,\tau)][K_v(t,s)^T a + k_{v5}(t,s)]\ y(\tau)y(s)\ d\tau\ ds\ dt \\
&= \frac{1}{2T}\int_a^b y(t)^2 dt - \frac{1}{T}\int_a^b \int_a^b K_v(t,\tau)^T\ y(\tau)y(t)\ d\tau\ dt\ a - \frac{1}{T}\int_a^b \int_a^b k_{v5}(t,\tau)\ y(\tau)y(t)\ d\tau\ dt \\
&+ \frac{1}{2T}\ a^T \int_a^b \int_a^b \left[\int_a^b K_v(t,\tau)K_v(t,s)^T\ dt\right] y(\tau)y(s)\ d\tau\ ds\ a \\
&+ \frac{1}{2T}\int_a^b \int_a^b \int_a^b k_{v5}(t,\tau)K_v(t,s)^T\ y(\tau)y(s)\ d\tau\ ds\ dt\ a \\
&+ a^T \frac{1}{2T}\int_a^b \int_a^b \int_a^b K_v(t,\tau)k_{v5}(t,s)\ y(\tau)y(s)\ d\tau\ ds\ dt \\
&+ \frac{1}{2T}\int_a^b \int_a^b \int_a^b k_{v5}(t,\tau)k_{v5}(t,s)]\ y(\tau)y(s)\ d\tau\ ds\ dt
\end{aligned}
$$

Assembling terms

$$
\begin{aligned}
&J(a) = d + b^T a + \frac{1}{2} a^T C a \quad \text{with} \\
&d := \left\{ \frac{1}{2T}\int_a^b y(t)^2 dt - \frac{1}{T}\int_a^b \int_a^b k_{v5}(t,\tau)\ y(\tau)y(t)\ d\tau\ dt \right. \\
&\qquad\qquad \left. + \frac{1}{2T}\int_a^b \int_a^b \int_a^b k_{v5}(t,\tau)k_{v5}(t,s)]\ y(\tau)y(s)\ d\tau\ ds\ dt \right\} \\
&b^T := \left\{ -\frac{1}{T}\int_a^b \int_a^b K_v(t,\tau)^T\ y(\tau)y(t)\ d\tau\ dt \right. \\
&\qquad\qquad \left. + \frac{1}{2T}\int_a^b \int_a^b \int_a^b [K_v(t,s)^T k_{v5}(t,\tau) + K_v(t,\tau)^T k_{v5}(t,s)]\ y(\tau)y(s)\ d\tau\ ds\ dt \right\} \\
&C := \frac{1}{T}\left\{ \int_a^b \int_a^b \left[\int_a^b K_v(t,\tau)K_v(t,s)^T\ dt\right] y(\tau)y(s)\ d\tau\ ds \right\}
\end{aligned}
$$

The standard quadratic cost yields a minimization problem w.r.t. parameter $a$ that is solved globally and analytically; [26] :

$$J(a) := d + b^T a + \frac{1}{2} a^T C a$$
$$\min\{J(a) \mid a \in \mathbb{R}^3\} \quad \text{is attained globally and uniquely at}$$
$$\hat{a} = -C^{-1} b; \quad \text{with minimum value} \;\; J(\hat{a}) = d - \frac{1}{2} b^T C^{-1} b \tag{3.44}$$

Also, it should be noted that the triple intergrals can be written as alternative integral products expressions which are easier to handle numerically

$$\int_a^b \int_a^b \int_a^b k_{v5}(t,\tau) k_{v5}(t,s)]\; y(\tau) y(s)\; d\tau\; ds\; dt = \int_a^b \left( \int_a^b k_{v5}(t,\tau)\; y(\tau)\; d\tau \right)^2 dt$$

$$\begin{aligned}
&\int_a^b \int_a^b \int_a^b [K_v(t,s)^T k_{v5}(t,\tau) + K_v(t,\tau)^T k_{v5}(t,s)]\; y(\tau) y(s)\; d\tau\; ds\; dt \\
&\quad = \int_a^b \left( \int_a^b K_v(t,s)^T y(s)\; ds \right) \left( \int_a^b k_{v5}(t,\tau)\; y(\tau)\; d\tau \right)\; dt \\
&\qquad + \int_a^b \left( \int_a^b K_v(t,\tau)^T y(\tau)\; d\tau \right) \left( \int_a^b k_{v5}(t,s)\; y(s)\; ds \right)\; dt
\end{aligned}$$

$$\begin{aligned}
&\int_a^b \int_a^b \left[ \int_a^b K_v(t,\tau) K_v(t,s)^T\; dt \right] y(\tau) y(s)\; d\tau\; ds \\
&\quad = \int_a^b \left[ \int_a^b K_v(t,\tau)\; y(\tau)\; d\tau \right] \left[ \int_a^b K_v(t,s)\; y(s)\; ds \right]^T\; dt
\end{aligned}$$

The discrete cost (3.40) can be computed similarly, as follows:

$$
\begin{aligned}
J(a) &:= \frac{1}{2n}\sum_{i=1}^{n}\left[y(t_i) - \int_a^b K_{DS}(t_i,\tau)\ y(\tau)d\tau\right]^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left[\frac{1}{2}y(t_i)^2 - y(t_i)\int_a^b K_{DS}(t_i,\tau)\ y(\tau)d\tau + \frac{1}{2}\left(\int_a^b K_{DS}(t_i,\tau)\ y(\tau)d\tau\right)^2\right] \\
&= \frac{1}{n}\sum_{i=1}^{n}\left[\frac{1}{2}y(t_i)^2 - y(t_i)\int_a^b K_{DS}(t_i,\tau)\ y(\tau)d\tau + \frac{1}{2}\int_a^b K_{DS}(t_i,\tau)\ y(\tau)d\tau\int_a^b K_{DS}(t_i,s)\ y(s)ds\right] \\
&= \frac{1}{2n}\sum_{i=1}^{n} y(t_i)^2 - \frac{1}{n}\sum_{i=1}^{n} y(t_i)\int_a^b K_{DS}(t_i,\tau)\ y(\tau)\ d\tau \\
&\qquad + \frac{1}{2n}\sum_{i=1}^{n}\int_a^b\int_a^b K_{DS}(t_i,\tau)K_{DS}(t_i,s)\ y(\tau)y(s)\ d\tau\ ds
\end{aligned}
$$

Expanding the kernels yields

$$
\begin{aligned}
&J(a) = \\
&= \frac{1}{2n}\sum_{i=1}^{n} y(t_i)^2 - \frac{1}{n}\sum_{i=1}^{n}\int_a^b [K_v(t_i,\tau)a + k_{v5}(t_i,\tau)]\ y(\tau)y(t_i)\ d\tau \\
&\quad + \frac{1}{2n}\sum_{i=1}^{n}\int_a^b\int_a^b [a^T K_v(t_i,\tau) + k_{v5}(t_i,\tau)][K_v(t_i,s)^T a + k_{v5}(t_i,s)]\ y(\tau)y(s)\ d\tau\ ds \\
&= \frac{1}{2n}\sum_{i=1}^{n} y(t_i)^2 dt - \frac{1}{n}\sum_{i=1}^{n}\int_a^b K_v(t_i,\tau)^T\ y(\tau)y(t_i)\ d\tau\ a - \frac{1}{n}\sum_{i=1}^{n}\int_a^b k_{v5}(t_i,\tau)\ y(\tau)y(t_i)\ d\tau \\
&+ \frac{1}{2n}a^T\sum_{i=1}^{n}\int_a^b\int_a^b K_v(t_i,\tau)K_v(t_i,s)^T y(\tau)y(s)\ d\tau\ ds\ a
\end{aligned}
$$

$$+\frac{1}{2n}\sum_{i=1}^{n}\int_a^b\int_a^b k_{v5}(t_i,\tau)K_v(t_i,s)^T\ y(\tau)y(s)\ d\tau\ ds\ a$$
$$+a^T\frac{1}{2n}\sum_{i=1}^{n}\int_a^b\int_a^b K_v(t_i,\tau)k_{v5}(t_i,s)\ y(\tau)y(s)\ d\tau\ ds$$
$$+\frac{1}{2n}\sum_{i=1}^{n}\int_a^b\int_a^b k_{v5}(t_i,\tau)k_{v5}(t_i,s)]\ y(\tau)y(s)\ d\tau\ ds$$

Assembling terms again delivers a standard quadratic

$$J(a)=d+b^Ta+\frac{1}{2}a^TCa \quad \text{with}$$
$$d:=\left\{\frac{1}{2n}\sum_{i=1}^{n}y(t_i)^2-\frac{1}{n}\sum_{i=1}^{n}\int_a^b k_{v5}(t_i,\tau)\ y(\tau)y(t_i)\ d\tau\right.$$
$$\left.+\frac{1}{2n}\sum_{i=1}^{n}\int_a^b\int_a^b k_{v5}(t_i,\tau)k_{v5}(t_i,s)]\ y(\tau)y(s)\ d\tau\ ds\right\}$$

$$b^T:=\left\{-\frac{1}{n}\sum_{i=1}^{n}\int_a^b K_v(t_i,\tau)^T\ y(\tau)y(t_i)\ d\tau\right.$$
$$\left.+\frac{1}{2n}\sum_{i=1}^{n}\int_a^b\int_a^b[K_v(t_i,s)^Tk_{v5}(t_i,\tau)+K_v(t_i,\tau)^Tk_{v5}(t_i,s)]\ y(\tau)y(s)\ d\tau\ ds\right\}$$

$$C:=\left\{\frac{1}{n}\sum_{i=1}^{n}\int_a^b\int_a^b K_v(t_i,\tau)K_v(t_i,s)^T\ y(\tau)y(s)\ d\tau\ ds\right\}$$

$$\min\{J(a)\mid a\in\mathbb{R}^3\} \quad \text{is attained globally and uniquely at}$$
$$\hat{a}=-C^{-1}b;\quad \text{with minimum value}\ \ J(\hat{a})=d-\frac{1}{2}b^TC^{-1}b \tag{3.45}$$

The double integrals above can again be written in terms of single integrals

$$\int_a^b \int_a^b k_{v5}(t_i,\tau)k_{v5}(t_i,s)]\ y(\tau)y(s)\ d\tau\ ds = \left(\int_a^b k_{v5}(t_i,\tau)\ y(\tau)\ d\tau\right)^2$$

$$\int_a^b \int_a^b [K_v(t_i,s)^T k_{v5}(t_i,\tau) + K_v(t_i,\tau)^T k_{v5}(t_i,s)]\ y(\tau)y(s)\ d\tau\ ds$$
$$= \left(\int_a^b K_v(t_i,s)^T y(s)\ ds\right)\left(\int_a^b k_{v5}(t_i,\tau)\ y(\tau)\ d\tau\right)$$
$$+ \left(\int_a^b K_v(t_i,\tau)^T y(\tau)\ d\tau\right)\left(\int_a^b k_{v5}(t_i,s)\ y(s)\ ds\right)$$

$$\int_a^b \int_a^b K_v(t_i,\tau)K_v(t_i,s)^T y(\tau)y(s)\ d\tau\ ds$$
$$= \left[\int_a^b K_v(t_i,\tau)\ y(\tau)\ d\tau\right]\left[\int_a^b K_v(t_i,s)\ y(s)\ ds\right]^T$$

### 3.2.1 Output estimation by projection [27]

The system output after estimating the parameters can be reconstructed using

$$\hat{y} = \int_a^b K_{DS}(t_i,\tau)z(\tau)d\tau \tag{3.46}$$

where $z(\tau)$ is the measured signal.

However, as a more precise alternative to the above, the system output can be smoothed/reconstructed from a noisy measurement by direct orthogonal projection onto the subspace spanned by the fundamental solutions of the characteristic equation (2.5) of the system. This is because every solution of the characteristic equation with the already identified parameter vector $a$ satisfies the reproducing property of Theorem 1 in chapter 2 and so, the projection onto the space of fundamental solutions will be the noise free trajectory of the system. The fundamental solutions of the characteristic equation are obtained by the direct integration of (2.5). This is performed as follows.

We select n independent vectors as initial conditions for the homogeneous LTI system in (2.5) in chapter 2. These initial conditions can be taken as the vectors of the canonical basis in $\mathbb{R}^n$ i.e,

$$\begin{aligned} e_1 &= [1, 0, ..., 0] \\ e_2 &= [0, 1, ..., 0] \\ &\cdots \quad (3.47) \\ e_n &= [0, 0, ..., 1] \quad (3.48) \end{aligned}$$

The system equation (2.5) is then solved for each individual initial condition yielding solutions

$$\overline{y_i}(a) = e_i \qquad i = 1, \cdots, n.$$

It is an elementary fact from the theory of ordinary differential equations that any solution of the system (2.5) with any initial condition is a linear combination of such fundamental solutions $\overline{y_i}$. Hence we search for the coefficients of this linear combination so that the resulting function is the closest to the output measurement data. Closest solution is found in terms of an orthogonal projection onto span of $S^a$.

$$S^a = \text{span}\left\{\overline{y}_i(\cdot), i = 1, \cdots, n\right\}$$

This is best done by orthonormalizing the set of fundamental solutions.The projection of a measured noisy signal $z(\cdot) \in L^2[a, b]$ into $S^a$ is given as,

$$y_E(\cdot) \triangleq \arg\min\left\{\|z - y\|_2^2 \,|y \in S^a\right\} \tag{3.49}$$

.

We seek,

$$\hat{y} = \sum_{i=1}^{n} \hat{c}_i \overline{y}_i \tag{3.50}$$

,

As $\hat{y}$ is a linear combination, the optimality conditions in (3.49) is achieved if and only if,

$$\left\langle z|\overline{y}_j\right\rangle_2 = \sum_{i=1}^{n} \hat{c}_i \left\langle \overline{y}_i|\overline{y}_j\right\rangle_2 j = 1, \cdots, n \tag{3.51}$$

which can be written in a matrix form as:

$$
\begin{aligned}
&v = G(\overline{y})\hat{c}; \quad G(\overline{y}) \triangleq \text{mat}\left\{\left\langle \overline{y}_i | \overline{y}_j \right\rangle_2\right\}_{i,j=1}^{n} \\
&v \triangleq \text{vec}\left\{\langle z|\overline{y}_i\rangle_2\right\}_{i=1}^{n}; \hat{c} \triangleq \text{vec}\left\{\hat{c}_i\right\}_{i=1}^{n}
\end{aligned}
\tag{3.52}
$$

$G$ is called the Gram matrix for vectors in span $S^a$ and is invertible because it is known that all fundamental solutions are linearly independent, from the theory of differential equations.

$$
\hat{c} = G^{-1}(\overline{y})v \tag{3.53}
$$

$\hat{y}$ , the estimated output is thus obtained from (3.50).

## 3.3 Parameter and State Estimation by Least Squares and Instrument Variable - Generalized Least squares [14] [24]

### 3.3.1 Parameter estimation as least squares problem

Consider the equation (2.6). As the kernels of Theorem 1 are linear in the unknown system coefficients, the reproducing property (for homogenous systems) is first re-written to bring out this fact while omitting the obvious dependence of the kernels on $n$. The term $\alpha_{ab}^{-1}(t)$ is subsumed in $K_{DS}$ for convenience of notation.

$$
y(t) = \int_a^b K_{DS,y}(t,\tau)y(\tau)\ d\tau \tag{3.54}
$$

$$
= \sum_{i=0}^{n} \beta_i \int_a^b K_{DS(i),y}(t,\tau)y(\tau)d\tau \tag{3.55}
$$

where the $K_{DS(i),y}; i = 0, \cdots, n$ are ‘component kernels’ of $K_{DS,y}$ that post-multiply the coefficients $\beta_i = a_i; i = 0, \cdots, n-1$, with $\beta_n = 1$ for convenience of notation. In a noise-free deterministic setting, the output variable $y$ becomes the measured output coinciding with the nominal output trajectory $y_T$, so the regression equation for the constant parameters

$a_i, i = 0, \ldots, n-1$, (3.55), can be written in a partitioned form as

$$\begin{aligned} y_T(t) &= [K^{\bar{a}}, K^1](t; y_T)\beta^T \\ \bar{a} &:= [a_0, \cdots, a_{n-1}]; \quad \beta^T := [\bar{a}; \beta_n] \end{aligned} \tag{3.56}$$

where $K^{\bar{a}}(t; y_T)$ is a row vector with integral components

$$K^{\bar{a}}(t; y_T)_k := \int_a^b K_{DS(k),y}(t,\tau) y_T(\tau) d\tau \; ; \; k = 0, \ldots n-1 \tag{3.57}$$

while $K^1(t; y_T)$ is a scalar

$$K^1(t; y_T) := \int_a^b K_{DS(n),y}(t,\tau) y_T(\tau) d\tau \tag{3.58}$$

corresponding to $\beta_n = 1$. Given distinct time instants $t_1, \cdots, t_N \in (a, b]$, here referred to as *knots*, the regression equation is re-written point-wise in the form of a matrix equation

$$Q(y_T) = P(y_T)\bar{a} \tag{3.59}$$

$$Q := \begin{bmatrix} q(t_1) \\ \vdots \\ q(t_N) \end{bmatrix} \quad ; P := \begin{bmatrix} p_0(t_1) \cdots p_{n-1}(t_1) \\ \ddots \\ p_0(t_N) \cdots p_{n-1}(t_N) \end{bmatrix}$$

$$\begin{aligned} q(t_i) &= y_T(t_i) - K^1(t_i, y_T); \\ p_k(t_i) &= K^{\bar{a}}(t_i; y_T)_k \end{aligned} \tag{3.60}$$

that can be solved using least squares error minimization provided adequate identifiability assumptions are met and the output is measured without error.

### 3.3.2 Parameter Estimation as Instrument Variable - Generalised Least Squares Method [24] [29]

In the presence of measurement noise, here assumed to be AWGN - white Gaussian and additive, the regression equation (3.55) is no longer valid as the reproducing property fails to hold along an inexact output trajectory. It must thus be suitably replaced leading to a stochastic regression problem. $y_M$ is assumed to be adapted to the natural ltration of the standard Wiener process W on [a,b] is

$$y_M(t,\omega) = y_T(t) + \eta^\sigma(t,\omega) \ ; \quad t \in [a,b] \tag{3.61}$$

where $\eta^\sigma$ signifies the AWGN with constant variance $\sigma^2$ and where $y_T$ is the true system output [30]. Without adhering to any particular realization of the measurement process this yields a random kernel expression

$$\int_a^b K_{DS,y}(t,\tau) y_M(\tau)\ d\tau \tag{3.62}$$

$$= \sum_{i=0}^{n} \beta_i \int_a^b K_{DS(i),y}(t,\tau) y_M(\tau) d\tau \tag{3.63}$$

which is better written using the proper stochastic nomenclature as

$$\int_a^b K_{DS,y}(t,\tau) y_T(\tau)\ d\tau + \int_a^b K_{DS,y}(t,\tau)\ dW^\sigma(\tau) \tag{3.64}$$

with its partitioned form (3.55) following similarly. Here, $W^\sigma$ is the Wiener process with intensity $\sigma$ so that, informally, $\eta^\sigma(t)dt = \sigma dW(t)$ with $W$ as the standard Brownian motion. It follows that the following equality is valid

$$y_M(t) = \int_a^b K_{DS,y}(t,\tau) y_M(\tau)\ d\tau + e(t) \tag{3.65}$$

$$\text{with} \quad e(t) := \eta^\sigma(t) - \int_a^b K_{DS,y}(t,\tau)\ dW^\sigma(\tau) \tag{3.66}$$

since $y_T$ satisfies the reproducing property in the deterministic regression equation (3.55). It is noted that the random error variable $e$ is dependent on the unknown system parameters $a_i, i = 0, \cdots, n-1$ while the stochastic regression equation

$$y_M(t) = \sum_{i=0}^{n} \beta_i \int_a^b K_{DS(i),y}(t,\tau) y_M(\tau) d\tau + e(t) \tag{3.67}$$

has the random regressor vector

$$\left[\int_a^b K_{DS(0),y}(t,\tau)y_M(\tau)d\tau, \cdots, \int_a^b K_{DS(n),y}(t,\tau)y_M(\tau)d\tau\right]^T \tag{3.68}$$

It is easily verified (see later developments) that the assumptions of the Gauss-Markov Theorem are violated in the linear regression problem (3.67) because the random regressor is correlated with a regression error, which additionally fails to be homoskedastic. The above regression is thus a typical 'error-in-the-variable' problem with heteroskedastic noise. Heteroskedasticity has serious consequences for the OLS estimator [31]. A standard quite powerful way to deal with *unknown* heteroskedasticity is to resort to generalized least squares (GLS), which can be shown to be BLUE (Best Linear Unbiased Estimator); see [32]. The GLS employs inverse covariance weighting in the regression error minimization problem associated with (3.67). In similarity with the notation used in the deterministic OLS (3.59) let $Q(y_M)$ and $P(y_M)$ be the matrices corresponding to $N$ samples of the measurement process realization $y_M$ at a batch of knots $t_1, \cdots, t_N$. Then the stochastic regression error vector is given by

$$e := [e(t_1), \cdots, e(t_N)]^T = Q(y_M) - P(y_M)\bar{a} \tag{3.69}$$

where $e(t_i)$ are as in (3.66). The standard GLS regression error minimization for estimation of the parameter vector $\bar{a}$ is

$$\min_{\bar{a}} \left( [Q(y_M) - P(y_M)\bar{a}]^T S [Q(y_M) - P(y_M)\bar{a})] \right) \tag{3.70}$$

$$\text{with} \quad S := [\text{Cov}(e)]^{-1} \tag{3.71}$$

Applying the expectation operator to equations (3.61) and (3.66) and using the properties of the Wiener process yields

$$\mathrm{E}[y_M(t)] = \mathrm{E}[y_T(t)] + \mathrm{E}[\eta^\sigma(t)] = y_T(t) \tag{3.72}$$

$$\mathrm{E}[e(t)] = \mathrm{E}[\eta^\sigma(t)] - \mathrm{E}\left[\int_a^b K_{DS}(t,\tau)dW^\sigma(\tau)\right] \tag{3.73}$$

$$= \int_a^b K_{DS}(t,\tau)\mathrm{E}[dW^\sigma(\tau)] = 0 \tag{3.74}$$

thus

$$\mathrm{Cov}(e) = \mathrm{E}\left[ee^T\right] \tag{3.75}$$

The components of the covariance matrix are calculated as $\mathrm{Cov}(e(t_i), e(t_j))$.

Proceeding somewhat informally, the covariance formula for the generalized white noise process is written as

$$\mathrm{E}[\eta^\sigma(t)\eta^\sigma(s)] = \sigma^2\delta(t-s) \tag{3.76}$$

The sifting property (also referred to as sampling property) of the delta Dirac function, which is valid for all tempered distributions $f$ (thus also functions with compact support which are square integrable), is needed here as stated in the form

$$\int_a^b f(t)\delta(t-s)dt = f(s) \tag{3.77}$$

Writing $\eta^\sigma(\tau)d\tau$ in place of $dW^\sigma(\tau)$, for lucidity of derivations, the covariance calculation is then carried out as follows.

$$
\begin{aligned}
&\mathrm{Cov}[e(t_i), e(t_j)] = \mathrm{E}[e(t_i)e(t_j)] \\
&= \mathrm{E}\left[\left[\eta^\sigma(t_i) - \int_a^b K_{DS}(t_i,\tau)\eta^\sigma(\tau)d\tau\right]\left[\eta^\sigma(t_j) - \int_a^b K_{DS}(t_j,s)\eta^\sigma(s)ds\right]\right] \\
&= \mathrm{E}[\eta^\sigma(t_i)\eta^\sigma(t_j)] - \mathrm{E}\left[\int_a^b K_{DS}(t_i,\tau)\eta^\sigma(t_j)\eta^\sigma(\tau)d\tau\right] - \mathrm{E}\left[\int_a^b K_{DS}(t_j,s)\eta^\sigma(t_i)\eta^\sigma(s)ds\right] \\
&\quad + \mathrm{E}\left[\int_a^b\int_a^b K_{DS}(t_i,\tau)K_{DS}(t_j,s)\eta^\sigma(\tau)\eta^\sigma(s)d\tau ds\right] \\
&= \sigma^2\delta(t_i - t_j) - \int_a^b K_{DS}(t_i,\tau)\mathrm{E}[\eta^\sigma(t_j)\eta^\sigma(\tau)]d\tau - \int_a^b K_{DS}(t_j,s)\mathrm{E}[\eta^\sigma(t_i)\eta^\sigma(s)]ds \\
&\quad + \int_a^b\int_a^b K_{DS}(t_i,\tau)K_{DS}(t_j,s)\mathrm{E}[\eta^\sigma(\tau)\eta^\sigma(s)]d\tau ds \\
&= \sigma^2\delta(t_i - t_j) - \sigma^2\int_a^b K_{DS}(t_i,\tau)\delta(\tau - t_j)d\tau - \sigma^2\int_a^b K_{DS}(t_j,\tau)\delta(\tau - t_i)d\tau \\
&\quad + \sigma^2\int_a^b K_{DS}(t_i,\tau)\int_a^b K_{DS}(t_j,s)\delta(s-\tau)ds\ d\tau \\
&= \sigma^2\delta(t_i - t_j) - \sigma^2 K_{DS}(t_i,t_j) - \sigma^2 K_{DS}(t_j,t_i) + \sigma^2\int_a^b K_{DS}(t_i,\tau)K_{DS}(t_j,\tau)d\tau
\end{aligned}
$$

The formula for the variance follows by substituting $t = t_i = t_j$, and recalling that, informally, $\delta(0) = 1$,

$$
\mathrm{Var}[e(t)] = \sigma^2 - 2\sigma^2 K_{DS}(t,t) + \sigma^2\int_a^b \left[K_{DS}(t,\tau)\right]^2 d\tau
$$

At this point, it should be clear that the standard GLS, as in (3.70), cannot be applied directly as the covariance matrix depends on the unknown variance $\sigma^2$ and also on the unknown parameter vector $\bar{a}$ in the $K_{DS}$ kernels. Hence a *feasible* version of the GLS must be employed here in which the covariance matrix is estimated progressively as more information about the regression residuals becomes available. This is typically performed as part of a recursive scheme in which consecutive batches of samples are drawn from the realization of $y_M$. Letting $Q_i - P_i\bar{a}$ denote the regression error $e_i$ in batch $i$, the recursive

GLS algorithm computes

$$\hat{a}_k = \arg\min_{\bar{a}} \left( \sum_{i=1}^{k} (Q_i - P_i\bar{a})^T S_i (Q_i - P_i\bar{a}) \right) \tag{3.78}$$

where $\hat{a}_k$ is the parameter estimate update at iteration $k$ of the algorithm. Each weighting matrix $S_{k+1}$, is calculated as the inverse of the covariance matrix based on the parameter estimate $\hat{a}_k$ and an estimate of the variance $\sigma^2$ obtained from the residual trajectory $y_M(t) - y_E(t)$ in previous iteration $k$, where $y_E$ signifies the estimated/reconstructed output. Detailed steps of a modified recursive scheme is shown in next chapter.

### 3.3.3 Errors-in-variables [24] [29]

It is well known that the presence of errors-in-variables induces an asymptotic bias in OLS regression estimates, which is proportional to the signal-to-noise ratio in the observed regressand. In such situations, the leading way to eliminate estimation bias is to use *Instrumental Variables* (IV); see [33], in the normal equations that deliver the optimal estimates. The IV method has a long history and multiple applications; refer to [34], [35], [36], [37], [38], [39].

To render statistical consistency for the estimation problem at hand, the IV is constructed by way of the backward reproducing kernel, as described below.

Referring to the exposition of the basic regression problem in section 3.3.1 it follows from Theorem 1 that the "double-sided" regression equation (3.56), can be cloned as two statistically independent regression equations corresponding to the forward and backward kernels $K_{F,y}$ and $K_{B,y}$ as follows:

$$\begin{aligned} (t-a)^n y_T(t) &= \int_a^t K_{F,y}(t,\tau)\ y_T(\tau) d\tau \\ &:= [K_F^{\bar{a}}, K_F^1](t; y_T)\ \beta\ ; t \in [a,b] \end{aligned} \tag{3.79}$$

$$\begin{aligned} (b-t)^n y_T(t) &= \int_t^b K_{B,y}(t,\tau)\ y_T(\tau) d\tau \\ &:= [K_B^{\bar{a}}, K_B^1](t; y_T)\ \beta\ ; t \in [a,b] \end{aligned} \tag{3.80}$$

Given a set of knots $[t_1, \cdots, t_N]; N > n; t_1 >> a; t_N << b$, the latter are written in discrete time as $N$ copies of (3.79) and (3.80) in matrix-vector form

$$Y_T = K_F(y_T)\beta \tag{3.81}$$
$$Y_T = K_B(y_T)\beta \tag{3.82}$$
$$Y_T := \begin{bmatrix} y_T(t_1) & \cdots & y_T(t_N) \end{bmatrix}^T$$
$$K_F(y_T) := \begin{bmatrix} \frac{1}{(t_1-a)^n}[K_F^{\bar{a}}, K_F^1](t_1, y_T) \\ \vdots \\ \frac{1}{(t_N-a)^n}[K_F^{\bar{a}}, K_F^1](t_N, y_T) \end{bmatrix};$$
$$K_B(y_T) := \begin{bmatrix} \frac{1}{(b-t_1)^n}[K_B^{\bar{a}}, K_B^1](t_1, y_T) \\ \vdots \\ \frac{1}{(b-t_N)^n}[K_B^{\bar{a}}, K_B^1](t_N, y_T) \end{bmatrix}$$

The equations (3.81) and (3.82) deliver two "independent" OLS estimators for the parameter vector $\beta$ corresponding to two normal equations :

$$K_F(y_T)^T K_F(y_T)\beta = K_F(y_T)^T Y_T$$
$$K_B(y_T)^T K_B(y_T)\beta = K_B(y_T)^T Y_T$$

namely:

$$\hat{\beta}_F := \left[K_F(y_T)^T K_F(y_T)\right]^{-1} K_F(y_T)^T Y_T \tag{3.83}$$
$$\hat{\beta}_B := \left[K_B(y_T)^T K_B(y_T)\right]^{-1} K_B(y_T)^T Y_T \tag{3.84}$$

provided that each of the inverted matrices are nonsingular. The more efficient of them would be the one that corresponds to an inverted matrix having smaller condition number.

Of course, nothing stands in the way of pre-multiplying the forward estimation equation (3.81) by the backward matrix $K_B(y_T)$, or, vice-versa, pre-multiplying the backward estimation equation (3.82) by the forward matrix $K_F(y_T)$. If any of the symmetric matrices satisfy :

$$\det\left[K_B(y_T)^T K_F(y_T)\right] \neq 0 \quad \text{or}$$
$$\det\left[K_F(y_T)^T K_B(y_T)\right] \neq 0 \tag{3.85}$$

the other one will also be nonsingular (because they are transposes of one another). This observation delivers two more estimators which, in the noiseless case, will be competitive with those in (3.83)-(3.84) :

$$K_B(y_T)^T K_F(y_T)\beta = K_B(y_T)^T Y_T$$
$$K_F(y_T)^T K_B(y_T)\beta = K_F(y_T)^T Y_T$$

so that

$$\hat{\beta}_{IVF} := \left[K_B(y_T)^T K_F(y_T)\right]^{-1} K_B(y_T)^T Y_T \tag{3.86}$$
$$\hat{\beta}_{IVB} := \left[K_F(y_T)^T K_B(y_T)\right]^{-1} K_F(y_T)^T Y_T \tag{3.87}$$

These are in fact "instrumental variable estimators" as compared with the OLS estimators (3.83) and (3.84) with $K_B$ as the IV for the forward equation (3.86) and with $K_F$ as the IV for the backward equation (3.87). Clearly, as long as the nonsingularity condition (3.85) is satisfied, then, in the noiseless case, all the estimators are bound to produce the same value of the estimated parameter vector, i.e., $\beta_F = \beta_B = \beta_{IVF} = \beta_{IVB}$.

The use of the IV as defined above is rigorously justified as follows. When a noisy realization of a measurement process $y_M$ replaces the unknown system output $y_T$, the regression equations (3.81), (3.82) re-write as

$$Y_M = K_F(y_M)\beta + \xi \tag{3.88}$$
$$Y_M = K_B(y_M)\beta + \psi \tag{3.89}$$

with the obvious meaning of $K_F(y_M)$ and $K_F(y_M)$ whose arguments were substituted accordingly. As follows from the definition of the stochastic regression error (3.66) the noise processes $\xi$ and $\psi$ are expressed by

$$\xi := \eta - K_F(\eta)\beta; \quad \psi := \eta - K_B(\eta)\beta \tag{3.90}$$

Now, the "instrumental variable" $K_B(y_M)$ is well correlated with $Y_M$ (as in the noise free case $K_B(y_T)$ produces a system output $y_T$ that is computed by the forward kernel as well as by the backward kernel). On the other hand, the backward kernel matrix $K_B(y_M)$ is uncorrelated with the "forward" noise $\xi$ (over the interval $(a,t)$) if the backward kernel $K_B(y_M)$ (involving the noise $\psi$) is calculated over a disjoint interval $(t+\varepsilon, b)$. Implied is the fact $\mathrm{E}[K_B(y_M)\psi] = 0$ while $\mathrm{E}[K_B(y_M)K_F(y_M)] \neq 0$ for a small separation constant $\varepsilon > 0$.
A symmetric statement is that the "instrumental variable" $K_F(y_M)$ is uncorrelated with the "backward" noise $\psi$.
The statistical properties of the IV instrument as discussed above restore consistency of the modified GLS estimator employing such IV instrument; see [33] for a proof in the OLS case. However, the IV-GLS recursions are now modified to

$$\hat{\beta}_{k+1} = \hat{\beta}_k + R_{k+1}\tilde{P}_{k+1}^T S_{k+1}(\tilde{Q}_{k+1} - \tilde{P}_{k+1}\hat{\beta}_k) \tag{3.91}$$

$$R_{k+1} = R_k - R_k\tilde{P}_{k+1}^T(S_{k+1}^{-1} + \tilde{P}_{k+1}R_k\tilde{P}_{k+1}^T)^{-1}\tilde{P}_{k+1}R_k \tag{3.92}$$

where the matrices $P$ and $Q$ are replaced by $\tilde{P}$ and $\tilde{Q}$

$$\tilde{P} = K_B(y_M)^T K_F(y_M) \ ; \ \tilde{Q} = K_B(y_M)^T Y_M \tag{3.93}$$

and $M_{k+1} = \sum_{i=0}^{k+1} \tilde{P}_i^T S_i \tilde{P}_i$ and $R_{k+1} = M_{k+1}^{-1}$ .
It is worth noticing that a stopping criterion of the recursive scheme is elegantly delivered by the fact that the final estimate should satisfy $\beta_n = 1$ where $\beta := [a_0, \cdots, a_{n-1}, \beta_n]$, lending a criterion

$$|\hat{a}_n - 1| < \epsilon \text{ for some } \epsilon > 0 \tag{3.94}$$

A similar recursive scheme for the estimator for $\hat{\beta}_{IVB}$ can also be developed.
When using the recursive IV-GLS algorithm, the covariance of the error terms should be calculated by replacing $K_{DS}$ by $K_F$ or $K_B$ appropriately.

# Chapter 4

# Finite Interval Parameter and State estimation in LTI Systems Using Kernel-Based Multiple Regression

The parameter estimation methods that are discussed in the previous chapter deals with a simple regression equation. In this thesis, the parameter estimation is obtained by recursive generalized least squares involving multiple regression. By using the reproducing property of Theorem 1, the kernel representation in (2.56) is first integrated two times to deliver two additional linearly independent regression equations; this is to match the number of the parameters with the number of regression equations. A third-order system is considered with its unknown parameters $a_0, a_1, a_2$ to be estimated.
The kernel representation (2.56) can be written simply as,

$$\alpha_a(t)y(t) \triangleq \int_a^t K_F(t,s)y(s)ds \tag{4.1}$$

and using the Cauchy formula for repeated integration on (4.1) yields

$$\frac{1}{(n-1)!}\int_a^t \alpha_a(s)(t-s)^{n-1}y(s)ds = \frac{1}{n!}\int_a^t (t-s)^n K_F(t,s)y(s)ds \quad n = 1,2 \tag{4.2}$$

From (2.37), we have $K_F(t,s)$ defined as,

$$\begin{aligned} K_F(t,s) &= \Big[9(s-a)^2 - a_2(s-a)^3\Big] \\ &+ (t-s)\Big[-18(s-a) + 6a_2(s-a)^2 - a_1(s-a)^3\Big] \\ &+ (t-s)^2\Big[6 - 6a_2(s-a) + 3a_1(s-a)^2 - a_0(s-a)^3\Big] \end{aligned} \tag{4.3}$$

Substituting (4.3) in (4.2), we get

$$\begin{aligned} &\frac{1}{(n-1)!}\int_a^t \alpha_a(s)(t-s)^{n-1}y(s)ds \\ &= \int_a^t \frac{(t-s)^n}{n!}\Big[9(s-a)^2 - a_2(s-a)^3\Big]y(s)ds \\ &+ \int_a^t \frac{(t-s)^{n+1}}{(n+1)!}\Big[-18(s-a) + 6a_2(s-a)^2 - a_1(s-a)^3\Big]y(s)ds \\ &+ \int_a^t \frac{(t-s)^{n+2}}{(n+2)!}\Big[6 - 6a_2(s-a) + 3a_1(s-a)^2 - a_0(s-a)^3\Big]y(s)ds \end{aligned} \tag{4.4}$$

The equation (4.2) can be re-written as

$$\frac{1}{(n-1)!}\int_a^t \alpha_a(s)(t-s)^{n-1}y(s)ds = \int_a^t K_F^n(t,s)y(s)ds \quad \text{for } n = 1,2 \tag{4.5}$$

where $K_F^n(t,s)$ is

$$\begin{aligned} K_F^n(t,s) &= \frac{(t-s)^n}{n!}\Big[9(s-a)^2 - a_2(s-a)^3\Big] \\ &+ \frac{(t-s)^{n+1}}{(n+1)!}\Big[-18(s-a) + 6a_2(s-a)^2 - a_1(s-a)^3\Big] \\ &+ \frac{(t-s)^{n+2}}{(n+2)!}\Big[6 - 6a_2(s-a) + 3a_1(s-a)^2 - a_0(s-a)^3\Big] \end{aligned} \tag{4.6}$$

A similar procedure is then used for the backward kernel representation.

$$\alpha_b(t)y(t) = \int_t^b K_B(t,s)y(s)ds = -\int_b^t K_B(t,s)y(s)ds \tag{4.7}$$

using the Cauchy formula for repeated integration on (4.7) yields

$$\frac{1}{(n-1)!}\int_b^t \alpha_b(s)(t-s)^{n-1}y(s)ds = -\frac{1}{n!}\int_b^t (t-s)^n K_B(t,s)y(s)ds \quad \text{for } n = 1,2 \tag{4.8}$$

So,

$$\frac{1}{(n-1)!}\int_t^b \alpha_b(s)(t-s)^{n-1}y(s)ds = -\frac{1}{n!}\int_t^b (t-s)^n K_B(t,s)y(s)ds \quad \text{for } n = 1,2 \tag{4.9}$$

From (2.53), we have $K_B(t,s)$ defined as,

$$\begin{aligned} K_B(t,s) = &\Bigg[9(b-s)^2 + a_2(b-s)^3\Bigg] \\ &+ (t-s)\Bigg[18(b-s) + 6a_2(b-s)^2 + a_1(b-s)^3\Bigg] \\ &+ \frac{1}{2}(t-s)^2\Bigg[6 + 6a_2(b-s) + 3a_1(b-s)^2 + a_0(b-s)^3\Bigg] \end{aligned} \tag{4.10}$$

Substituting (4.10) in (4.9), we get

$$\begin{aligned} &\frac{1}{(n-1)!}\int_t^b \alpha_b(s)(t-s)^{n-1}y(s)ds \\ &= -\Bigg[\Bigg[\int_t^b \frac{(t-s)^n}{n!} 9(b-s)^2 + a_2(b-s)^3\Bigg]y(s)ds \\ &+ \int_t^b \frac{(t-s)^{n+1}}{(n+1)!}\Bigg[18(b-s) + 6a_2(b-s)^2 + a_1(b-s)^3\Bigg]y(s)ds \\ &+ \int_t^b \frac{(t-s)^{n+2}}{(n+2)!}\Bigg[6 + 6a_2(b-s) + 3a_1(b-s)^2 + a_0(b-s)^3\Bigg]y(s)ds\Bigg] \end{aligned} \tag{4.11}$$

The equation (4.9) can be re-written as

$$\frac{1}{(n-1)!}\int_t^b \alpha_b(s)(t-s)^{n-1}y(s)ds = -\int_t^b K_B^n(t,s)y(s)ds \tag{4.12}$$

where $K_B^n(t,s)$ is

$$\begin{aligned} K_B^n(t,s) = \Bigg[&\frac{(t-s)^n}{n!}\Big[9(b-s)^2 + a_2(b-s)^3\Big] \\ &+ \frac{(t-s)^{n+1}}{(n+1)!}\Big[18(b-s) + 6a_2(b-s)^2 + a_1(b-s)^3\Big] \\ &+ \frac{(t-s)^{n+2}}{(n+2)!}\Big[6 + 6a_2(b-s) + 3a_1(b-s)^2 + a_0(b-s)^3\Big]\Bigg] \end{aligned} \tag{4.13}$$

Adding (4.1) and (4.7) yields

$$\alpha_a(t)y(t) + \alpha_b(t)y(t) = \int_a^t K_F(t,s)y(s)ds + \int_t^b K_B(t,s)y(s)ds \tag{4.14}$$

$$\alpha_a(t)y(t) + \alpha_b(t)y(t) = \int_a^b K_{DS}(t,s)y(s)ds \tag{4.15}$$

$$\alpha_{ab}(t)y(t) = \int_a^b K_{DS}(t,s)y(s)ds \tag{4.16}$$

with

$$K_{DS}(t,s) \triangleq \begin{cases} K_F(t,s) : s \leq t \\ K_B(t,s) : s > t \end{cases} \tag{4.17}$$

and

$$\alpha_{ab}(t) = \alpha_a(t) + \alpha_b(t) \tag{4.18}$$

and adding (4.5) and (4.12) yields

$$\frac{1}{(n-1)!}\left[\int_a^t \alpha_a(s)(t-s)^{n-1}y(s)ds + \int_t^b \alpha_b(s)(t-s)^{n-1}y(s)ds\right] = \int_a^t K_F^n(t,s)y(s)ds + \int_t^b -K_B^n(t,s)y(s)ds \tag{4.19}$$

$$\frac{1}{(n-1)!}\left[\int_a^t \alpha_a(s)(t-s)^{n-1}y(s)ds + \int_t^b \alpha_b(s)(t-s)^{n-1}y(s)ds\right] = \int_a^b K_{DS}^n(t,s)y(s)ds \tag{4.20}$$

Similar to (4.16), we can write (4.20) as,

$$\frac{1}{(n-1)!}\left[\int_a^b \alpha_{ab}(s)(t-s)^{n-1}y(s)ds\right] = \int_a^b K_{DS}^n(t,s)y(s)ds \tag{4.21}$$

with

$$K_{DS}^n(t,s) \triangleq \begin{cases} K_F^n(t,s) : s \leq t \\ -K_B^n(t,s) : s > t \end{cases} \tag{4.22}$$

The kernels of Theorem 1 are linear in the unknown system coefficients $\beta_i$. Hence, the reproducing property for homogeneous systems is first re-written to bring out this fact while omitting the dependence of the kernels on $n$ [24]. The additional linearly independent equations formed by repeated integration as in (4.21) with $n = 1, 2$ can also be represented. The equations (4.16) and (4.21) can be re-written as follows

$$\alpha_{ab}(t)y(t) = \sum_{i=0}^{n} \beta_i \int_a^b K_{DS(i)}(t,s)y(s)ds \tag{4.23}$$

$$\int_a^b \alpha_{ab}(s)y(s)ds = \sum_{i=0}^{n} \beta_i \int_a^b K_{DS(i)}^1(t,s)y(s)ds \quad \text{for } n=1 \tag{4.24}$$

$$\int_a^b \alpha_{ab}(s)(t-s)y(s)ds = \sum_{i=0}^{n} \beta_i \int_a^b K_{DS(i)}^2(t,s)y(s)ds \quad \text{for } n=2 \tag{4.25}$$

where the $K_{DS(i)}$ and $K^n_{DS(i)}$, $i = 0, ..., n$ are 'component kernels' of $K_{DS}$ and $K^n_{DS}$ that post-multiply the coefficients $\beta_i = a_i$, $i = 0, ..., n-1$ , with $\beta_n = 1$ for convenience of notation. In a noise-free deterministic setting, the output variable y becomes the measured output coinciding with the nominal output trajectory $y_T$, so the regression equation for the constant parameters $a_i$, $i = 0, ..., n-1$, can be written in a partitioned form as

$$\alpha_{ab}(t)y_T(t) = [K_{DS(i)}, K_{DS(n)}](t; y_T)\beta^T \tag{4.26}$$

$$\int_a^b \alpha_{ab}(s)y(s)ds = [K^1_{DS(i)}, K^1_{DS(n)}](t; y_T)\beta^T \quad \text{for } n = 1 \tag{4.27}$$

$$\int_a^b \alpha_{ab}(s)(t-s)y(s)ds = [K^2_{DS(i)}, K^2_{DS(n)}](t; y_T)\beta^T \quad \text{for } n = 2 \tag{4.28}$$

Let

$$\overline{a} := [a_0, a_1, ..., a_{n-1}]; \beta^T := [\overline{a}, \beta_n], i = 0, 1, ..., n-1. \tag{4.29}$$

In the above equations, $K_{DS(i)}(t; y_T)$ , $K^1_{DS(i)}(t; y_T)$, $K^2_{DS(i)}(t; y_T)$ are row vectors with integral components corresponding to $\overline{a}$ and $K_{DS(n)}(t; y_T)$ , $K^1_{DS(n)}(t; y_T)$, $K^2_{DS(n)}(t; y_T)$ are scalars with respect to $\beta_n = 1$. The equations (4.26),(4.27),(4.28) can be written as follows

$$\alpha_{ab}(t)y_T(t) = \overline{a}\left[\int_a^b K_{DS(i)}(t,s)y(s)ds\right] + \beta_n\left[\int_a^b K_{DS(n)}(t,s)y(s)ds\right] \tag{4.30}$$

$$\int_a^b \alpha_{ab}(s)y(s)ds = \overline{a}\left[\int_a^b K^1_{DS(i)}(t,s)y(s)ds\right] + \beta_n\left[\int_a^b K^1_{DS(n)}(t,s)y(s)ds\right] \tag{4.31}$$

$$\int_a^b \alpha_{ab}(s)(t-s)y(s)ds = \overline{a}\left[\int_a^b K^2_{DS(i)}(t,s)y(s)ds\right] + \beta_n\left[\int_a^b K^2_{DS(n)}(t,s)y(s)ds\right] \tag{4.32}$$

Given distinct time instants $t_1, ..., t_N \in (a, b]$, here referred to as knots, the regression equation is re-written point-wise in the form of a matrix equation

$$Q(y_T) = P(y_T)\overline{a} \tag{4.33}$$

with

$$\overline{a} = [a_1, a_2, a_3] \tag{4.34}$$

for a third order system
In general,

$$Q = \begin{bmatrix} q^1(t_1) \\ \vdots \\ q^1(t_N) \\ \vdots \\ q^n(t_1) \\ \vdots \\ q^n(t_N) \end{bmatrix}; P = \begin{bmatrix} p_0^1(t_1) & \dots & p_{n-1}^1(t_1) \\ \vdots & \ddots & \vdots \\ p_0^1(t_N) & \dots & p_{n-1}^1(t_N) \\ \vdots & \ddots & \vdots \\ p_0^n(t_1) & \dots & p_{n-1}^n(t_1) \\ \vdots & \ddots & \vdots \\ p_0^n(t_N) & \dots & p_{n-1}^n(t_N) \end{bmatrix} \tag{4.35}$$

The equations (4.30),(4.31),(4.32) are re-written in the below way for the ease of applying least squares approach to estimate the parameters,

$$\alpha_{ab}(t)y_T(t) - \beta_n\Big[\int_a^b K_{DS(n)}(t,s)y(s)ds\Big] = \overline{a}\Big[\int_a^b K_{DS(i)}(t,s)y(s)ds\Big] \tag{4.36}$$

$$\int_a^b \alpha_{ab}(s)y(s)ds - \beta_n\Big[\int_a^b K^1_{DS(n)}(t,s)y(s)ds\Big] = \overline{a}\Big[\int_a^b K^1_{DS(i)}(t,s)y(s)ds\Big] \tag{4.37}$$

$$\int_a^b \alpha_{ab}(s)(t-s)y(s)ds - \beta_n\Big[\int_a^b K^2_{DS(n)}(t,s)y(s)ds\Big] = \overline{a}\Big[\int_a^b K^2_{DS(i)}(t,s)y(s)ds\Big] \tag{4.38}$$

Here

$$Q = \begin{bmatrix} q^1(t_i) \\ q^2(t_i) \\ q^3(t_i) \end{bmatrix} = \begin{bmatrix} \alpha_{ab}(t)y_T(t_i) - \beta_n \Big[ \int_a^b K_{DS(n)}(t_i, s)y(s)ds \Big] \\ \int_a^b \alpha_{ab}(s)y(s)ds - \beta_n \Big[ \int_a^b K^1_{DS(n)}(t_i, s)y(s)ds \Big] \\ \int_a^b \alpha_{ab}(s)(t_i - s)y(s)ds - \beta_n \Big[ \int_a^b K^2_{DS(n)}(t, s)y(s)ds \Big] \end{bmatrix} \tag{4.39}$$

$$P = \begin{bmatrix} p^1(t_i) \\ p^2(t_i) \\ p^3(t_i) \end{bmatrix} = \begin{bmatrix} \int_a^b K_{DS(i)}(t_i, s)y(s)ds \\ \int_a^b K^1_{DS(i)}(t_i, s)y(s)ds \\ \int_a^b K^2_{DS(i)}(t_i, s)y(s)ds \end{bmatrix} \tag{4.40}$$

The regression equation in (4.33) can be solved using least-squares error minimization provided adequate identifiability assumptions are met, and the output is measured without error.

## 4.1 Identifiability of homogeneous LTI systems from a single realization of a measured output [24] [29]

Identifiability of a homogeneous LTI system such as

$$\begin{aligned} &\dot{x}(t) = Ax(t); \quad y = Cx; \quad x \in \mathbb{R}^n \\ &x(0) = b \end{aligned} \tag{4.41}$$

from a single noise-free realization of its output trajectory $y$ on the interval $[0, \infty)$ has been studied by Stanhope [40]. The identiability condition stated in equivalent form:
*Definition 2: Model 4.42 is globally identifiable from b if and only if the functional mapping* $A \mapsto y(\cdot; A, b)$ *is injective on* $\mathbb{R}^n$ *where* $y(\cdot; A, b)$ *denotes the output orbit of 4.42 through b.*

**Theorem 3.** *Model 4.42 is globally identifiable from b if and only if the output orbit of 4.42 is not confined to a proper subspace of* $\mathbb{R}^n$.

The above criterion has limited use for reasons of practicality: it is difficult to verify computationally, pertains to infinite time horizons $[0, \infty)$, and most importantly, requires the output trajectory to be known exactly. Hence it suffices to invoke a practical version of identifiability as dened below.

*Definition 3: Practical linear identifiability*
The homogeneous system (2.6) is practically linearly identifiable on $[a, b]$ with respect to a particular noisy discrete realization of the output measurement process, $y(t), t \in [a, b]$, if and only if there exist distinct knots $t_1, \cdots, t_N \in (a, b]$ which render $rankP(y_M) = n$. Any such output realization is then called *persistent*.

In the presence of large measurement noise, here assumed to be AWGN - white gaussian and additive, the regression equation (4.23),(4.24) and (4.25) is no longer valid as the reproducing property fails to hold along an inexact output trajectory. It must thus be suitably replaced leading to a stochastic regression problem. First, the stochastic output measurement process on a probability space $L^2(\Omega, \mathcal{F}, \mathbb{P})$ where $y_M$ is assumed to be adapted to the natural ltration of the standard Wiener process W on $[a, b]$ is

$$y_M(t, \omega) = y_T(t) + \sigma \dot{W}(t, \omega) \ ; \quad t \in [a, b] \tag{4.42}$$

where $y_T$ signifies the true system output and $\sigma \dot{W}$ is the white noise process with constant variance $\sigma^2$.
The following facts are recalled from [30] [41]. For any smooth and compactly supported function $g = g(t)$ the generalized derivative $\dot{}\dot{w}(t)$ of $w(t)$ (not obligatory differentiable function) is defined symbolically as

$$\int_0^\infty g(t)\dot{w}(t)dt = -\int_0^\infty \dot{g}(t)w(t)dt \tag{4.43}$$

For a smooth $w(t)$, when $\dot{}\dot{w}(t)$ exists, the above formula is nothing but "integration by parts" formula.
The generalized derivative (t) of Wiener process, defined similarly to integration by parts

with a smooth g(t)

$$g(t)W_t = \int_0^t g(s)\dot{W}_s ds + \int_0^t \dot{g}(s)W_s ds \tag{4.44}$$

is called “Gaussian white noise”.
Main properties of Gaussian white noise are as follows [30] [41]:

1. Expectation of white noise:
$$E[\dot{W}_t] \equiv 0 \tag{4.45}$$

2. Covariance function of white noise is a dirac function:
$$Cov[\dot{W}_t, \dot{W}_s] = E[\dot{W}_t \dot{W}_s] = \delta(t-s) \tag{4.46}$$

3. Variance:
$$Var[\dot{W}_t] = E[\dot{W}_t^2] = 1 \tag{4.47}$$

4. $\int_0^t g(s)\dot{W}_s ds$ is well defined, if only
$$\int_0^t g(s)^2 ds < \infty \tag{4.48}$$

5. Moreover, for any square integrable function $g$,
$$E\int_0^t g(s)\dot{W}_s ds = 0 \tag{4.49}$$
$$E\Big(\int_0^t g(s)\dot{W}_s ds\Big)^2 = \int_0^t g(s)^2 ds \tag{4.50}$$

Here $\delta$ is the delta Dirac distribution acting on square integrable functions as an evaluation functional:

$$\int_a^b g(s)\delta(t-s)ds = g(t), \quad t \in [a,b] \tag{4.51}$$

The equation (4.42) is rewritten again,

$$y_M(t,\omega) = y_T(t) + \sigma \dot{W}(t,\omega)\ ; \quad t \in [a,b] \tag{4.52}$$

This corresponds to a random kernel expression,

$$\int_a^b K_{DS,y}(t,s) y_M(s)\ d\tau = \sum_{i=0}^{n} \beta_i \int_a^b K_{DS(i),y}(t,s) y_M(s) ds \tag{4.53}$$

The above expression can be better-written w.r.t (4.52) as follows,

$$\int_a^b K_{DS,y}(t,s) y_M(s)\ d\tau = \int_a^b K_{DS,y}(t,s) y_T(s)\ ds + \int_a^b K_{DS,y}(t,s) \sigma \dot{W}(s) \tag{4.54}$$

Here, $\sigma \dot{W}(s)$ is the Wiener process with $\sigma$ as intensity and $W$ as the standard Brownian motion. This applies to all the independent regression equations (4.16) and (4.21). It follows that the following equalities are valid

$$\alpha_{ab}(t) y_M(t) = \int_a^b K_{DS,y}(t,s) y_M(s) ds + e_1(t) \tag{4.55}$$

$$\int_a^b \alpha_{ab}(s) y_M(s) ds = \int_a^b K^1_{DS,y}(t,s) y_M(s) ds + e_2(t) \tag{4.56}$$

$$\int_a^b \alpha_{ab}(s)(t-s) y_M(s) ds = \int_a^b K^2_{DS,y}(t,s) y_M(s) ds + e_3(t) \tag{4.57}$$

The above equations can also be written as

$$\alpha_{ab}(t) y_M(t) = \sum_{i=0}^{n} \beta_i \int_a^b K_{DS(i)}(t,s) y_M(s) ds + e_1(t) \tag{4.58}$$

$$\int_a^b \alpha_{ab}(s) y_M(s) ds = \sum_{i=0}^{n} \beta_i \int_a^b K^1_{DS(i)}(t,s) y_M(s) ds + e_2(t) \tag{4.59}$$

$$\int_a^b \alpha_{ab}(s)(t-s) y_M(s) ds = \sum_{i=0}^{n} \beta_i \int_a^b K^2_{DS(i)}(t,s) y_M(s) ds + e_3(t) \tag{4.60}$$

The above equations can be represented in the matrix form as follows,

$$\begin{bmatrix} \alpha_{ab}(t)y_M(t) \\ \int_a^b \alpha_{ab}(s)y_M(s)ds \\ \int_a^b \alpha_{ab}(s)(t-s)y_M(s)ds \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^{n} \beta_i \int_a^b K_{DS(i)}(t,s)y_M(s)ds \\ \sum_{i=0}^{n} \beta_i \int_a^b K^1_{DS(i)}(t,s)y_M(s)ds \\ \sum_{i=0}^{n} \beta_i \int_a^b K^2_{DS(i)}(t,s)y_M(s)ds \end{bmatrix} + \begin{bmatrix} e_1(t) \\ e_2(t) \\ e_3(t) \end{bmatrix} \tag{4.61}$$

Substituting (4.30), (4.31) , (4.32) into (4.61), we get

$$\begin{bmatrix} \alpha_{ab}(t)y_M(t) \\ \int_a^b \alpha_{ab}(s)y_M(s)ds \\ \int_a^b \alpha_{ab}(s)(t-s)y_M(s)ds \end{bmatrix} = \begin{bmatrix} \overline{a} \int_a^b K_{DS(i)}(t,s)y_M(s)ds + \beta_n \int_a^b K_{DS(n)}(t,s)y_M(s)ds \\ \overline{a} \int_a^b K^1_{DS(i)}(t,s)y_M(s)ds + \beta_n \int_a^b K^1_{DS(n)}(t,s)y_M(s)ds \\ \overline{a} \int_a^b K^2_{DS(i)}(t,s)y_M(s)ds + \beta_n \int_a^b K^2_{DS(n)}(t,s)y_M(s)ds \end{bmatrix} + \begin{bmatrix} e_1(t) \\ e_2(t) \\ e_3(t) \end{bmatrix} \tag{4.62}$$

$$\begin{bmatrix} \alpha_{ab}(t)y_M(t) - \beta_n \int_a^b K_{DS(n)}(t,s)y_M(s)ds \\ \int_a^b \alpha_{ab}(s)y_M(s)ds - \beta_n \int_a^b K^1_{DS(n)}(t,s)y_M(s)ds \\ \int_a^b \alpha_{ab}(s)(t-s)y_M(s)ds - \beta_n \int_a^b K^2_{DS(n)}(t,s)y_M(s)ds \end{bmatrix} = \begin{bmatrix} \int_a^b K_{DS(i)}(t,s)y_M(s)ds \\ \int_a^b K^1_{DS(i)}(t,s)y_M(s)ds \\ \int_a^b K^2_{DS(i)}(t,s)y_M(s)ds \end{bmatrix} \overline{a} + \begin{bmatrix} e_1(t) \\ e_2(t) \\ e_3(t) \end{bmatrix} \tag{4.63}$$

The equations (4.55), (4.56), (4.65) have the random regressor matrix

$$\begin{bmatrix} \int_a^b K_{DS(0),y}(t,s)y_M(s)ds, \cdots, \int_a^b K_{DS(n),y}(t,s)y_M(s)ds \\ \int_a^b K^1_{DS(0),y}(t,s)y_M(s)ds, \cdots, \int_a^b K^1_{DS(n),y}(t,s)y_M(s)ds \\ \int_a^b K^2_{DS(0),y}(t,s)y_M(s)ds, \cdots, \int_a^b K^2_{DS(n),y}(t,s)y_M(s)ds \end{bmatrix}^T \tag{4.64}$$

and the error terms are written as

$$e_1(t) := \alpha_{ab}(t)(\sigma\dot{W}(t)) - \int_a^b K_{DS,y}(t,s)\sigma\dot{W}(s)ds \tag{4.65}$$

$$e_2(t) := \int_a^b \alpha_{ab}(s)(\sigma\dot{W}(s))ds - \int_a^b K^1_{DS,y}(t,s)\sigma\dot{W}(s)ds \tag{4.66}$$

$$e_3(t) := \int_a^b \alpha_{ab}(s)(t-s)(\sigma\dot{W}(s))ds - \int_a^b K^2_{DS,y}(t,s)\sigma\dot{W}(s)ds \tag{4.67}$$

since $y_T$ satisfies the reproducing property in the deterministic regression equation (4.23),(4.24) and (4.25) . It is noted that the random error variable $e$ is dependent on the unknown system parameters $a_i, i = 0, \cdots, n-1$. It can be verified that the assumptions of Gauss-Markov Theorem are violated in the linear regression problem because the random regressor is correlated with a regression error, which additionally fails to be homoskedastic. The problem of heteroskedasticity is dealt next.

## 4.2 Heteroskedicity [24]

As seen in the previous chapter, Heteroskedasticity has serious consequences for the Ordinary Least Squares estimator. Despite the fact that the OLS estimator remains unbiased, the estimated regression error is wrong, while confidence intervals cannot be relied on. The best way to deal with heteroskedasticity is to employ Recursive Generalized Least Squares (GLS) as it employs inverse covariance weighting in the regression error minimization problem associated with (4.58), (4.59) and (4.60). Let $Q(y_M)$ and $P(y_M)$ as defined in the (4.39) be the matrices corresponding to $N$ samples of the measurement process realization $y_M$ at a batch of knots $t_1, \cdots, t_N$. Then the stochastic regression error vector is given by

$$e_n := [e_n(t_i), \cdots, e_n(t_N)]^T = Q(y_M) - P(y_M)\bar{a} \tag{4.68}$$

where $e_n(t_i)$ are as in (4.65), (4.66) and (4.67). The standard GLS regression error minimization for estimation of the parameter vector $\bar{a}$ is

$$\min_{\bar{a}} \left( [Q(y_M) - P(y_M)\bar{a}]^T S [Q(y_M) - P(y_M)\bar{a})] \right) \tag{4.69}$$

$$\text{with} \quad S := [\mathrm{Cov}(e)]^{-1} \tag{4.70}$$

Applying the expectation operator to equations (4.42), (4.65), (4.66) and (4.67) and using the properties of the Gaussian noise yields

$$\mathrm{E}[y_M(t)] = \mathrm{E}[y_T(t)] + \mathrm{E}[\sigma\dot{W}(t)] = y_T(t) \quad t \in [a, b] \tag{4.71}$$

$$\mathrm{E}[e_1(t)] = \mathrm{E}\left[\alpha_{ab}(t)(\sigma\dot{W}(t))\right] - \mathrm{E}\left[\int_a^b K_{DS,y}(t,s)\sigma\dot{W}(s)ds\right] = 0 \tag{4.72}$$

$$\mathrm{E}[e_2(t)] = \mathrm{E}\left[\int_a^b \alpha_{ab}(s)(\sigma\dot{W}(s))ds\right] - \mathrm{E}\left[\int_a^b K^1_{DS,y}(t,s)\sigma\dot{W}(s)ds\right] = 0 \tag{4.73}$$

$$\mathrm{E}[e_3(t)] = \mathrm{E}\left[\int_a^b \alpha_{ab}(s)(t-s)(\sigma\dot{W}(s))ds\right] - \mathrm{E}\left[\int_a^b K^2_{DS,y}(t,s)\sigma\dot{W}(s)ds\right] = 0 \tag{4.74}$$

thus

$$\mathrm{Cov}(e) = \mathrm{E}\left[ee^T\right] \tag{4.75}$$

Following is the calculation of the error covariance matrix, which is used in the GLS algorithm.

## 4.3 Calculation of error covariance matrix needed for recursive GLS method

The error terms $e_1, e_2$, and $e_3$ are specified in the equations (4.65), (4.66), and (4.67). Using the properties (4.45) - (4.51) of AWGN and recalling that the kernel functions are Hilbert-Schmidt, hence are square integrable, the covariance matrix is then calculated as follows.

$$
\begin{aligned}
&Cov[e_1(t_i), e_1(t_j)] = E[e_1(t_i)e_1(t_j)] \\
&= \sigma^2 E\Bigg[\Big[\alpha_{ab}(t_i)\dot{W}(t_i) - \int_a^b K_{DS}(t_i,s)\dot{W}(s)ds\Big]\Big[\alpha_{ab}(t_j)\dot{W}(t_j) - \int_a^b K_{DS}(t_j,\tau)\dot{W}(\tau)d\tau\Big]\Bigg]
\end{aligned} \tag{4.76}
$$

$$
\begin{aligned}
&= \sigma^2 E\Big[\alpha_{ab}(t_i)\alpha_{ab}(t_j)\dot{W}(t_i)\dot{W}(t_j)\Big] - \sigma^2 E\Big[\alpha_{ab}(t_i)\dot{W}(t_i)\int_a^b K_{DS}(t_j,\tau)\dot{W}(\tau)d\tau\Big] \\
&\quad - \sigma^2 E\Big[\alpha_{ab}(t_j)\dot{W}(t_j)\int_a^b K_{DS}(t_i,s)\dot{W}(s)ds\Big] \\
&\quad + \sigma^2 E\Big[\int_a^b\int_a^b K_{DS}(t_i,s)K_{DS}(t_j,\tau)\dot{W}(s)\dot{W}(\tau)dsd\tau\Big]
\end{aligned} \tag{4.77}
$$

$$
\begin{aligned}
&= \sigma^2 E\Big[\alpha_{ab}(t_i)\alpha_{ab}(t_j)\dot{W}(t_i)\dot{W}(t_j)\Big] - \sigma^2 E\Big[\alpha_{ab}(t_i)\int_a^b K_{DS}(t_j,\tau)\dot{W}(t_i)\dot{W}(\tau)d\tau\Big] \\
&\quad - \sigma^2 E\Big[\alpha_{ab}(t_j)\int_a^b K_{DS}(t_i,s)\dot{W}(t_j)\dot{W}(s)ds\Big] \\
&\quad + \sigma^2 E\Big[\int_a^b\int_a^b K_{DS}(t_i,s)K_{DS}(t_j,\tau)\dot{W}(s)\dot{W}(\tau)dsd\tau\Big]
\end{aligned} \tag{4.78}
$$

$$
\begin{aligned}
&= \sigma^2 \alpha_{ab}(t_i)\alpha_{ab}(t_j)E\Big[\dot{W}(t_i)\dot{W}(t_j)\Big] - \sigma^2\alpha_{ab}(t_i)\int_a^b K_{DS}(t_j,\tau)E\Big[\dot{W}(t_i)\dot{W}(\tau)\Big]d\tau \\
&\quad - \sigma^2\alpha_{ab}(t_j)\int_a^b K_{DS}(t_i,s)E\Big[\dot{W}(t_j)\dot{W}(s)\Big]ds \\
&\quad + \sigma^2\int_a^b\int_a^b K_{DS}(t_i,s)K_{DS}(t_j,\tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau
\end{aligned} \tag{4.79}
$$

Applying the property (4.46) in the above equation.

$$
\begin{aligned}
&= \sigma^2\alpha_{ab}(t_i)\alpha_{ab}(t_j)\delta(t_i - t_j) - \sigma^2\alpha_{ab}(t_i)\int_a^b K_{DS}(t_j,\tau)\delta(t_i-\tau)d\tau \\
&\quad - \sigma^2\alpha_{ab}(t_j)\int_a^b K_{DS}(t_i,s)\delta(t_j - s)ds + \sigma^2\int_a^b K_{DS}(t_i,s)\int_a^b K_{DS}(t_j,\tau)\delta(s-\tau)d\tau ds
\end{aligned} \tag{4.80}
$$

Applying the property (4.51) in the above equation.

$$
\begin{aligned}
&= \sigma^2 \alpha_{ab}(t_i)\alpha_{ab}(t_j)\delta(t_i - t_j) - \sigma^2 \alpha_{ab}(t_i) K_{DS}(t_i, t_j) - \sigma^2 \alpha_{ab}(t_j) K_{DS}(t_j, t_i) \\
&\quad + \sigma^2 \int_a^b K_{DS}(t_i, s) K_{DS}(t_j, s) ds
\end{aligned} \tag{4.81}
$$

$$
\begin{aligned}
Cov[e_2(t_i), e_2(t_j)] &= E[e_2(t_i)e_2(t_j)] \\
&= \sigma^2 E\Big[\Big[\int_a^b \alpha_{ab}(s)\dot{W}(s)ds - \int_a^b K^1_{DS}(t_i, s)\dot{W}(s)ds\Big] \\
&\qquad \Big[\int_a^b \alpha_{ab}(\tau)\dot{W}(\tau)d\tau - \int_a^b K^1_{DS}(t_j, \tau)\dot{W}(\tau)d\tau\Big]\Big]
\end{aligned} \tag{4.82}
$$

$$
\begin{aligned}
&= \sigma^2 E\Big[\int_a^b \int_a^b \alpha_{ab}(s)\alpha_{ab}(\tau)\dot{W}(s)\dot{W}(\tau)dsd\tau\Big] \\
&\qquad - \sigma^2 E\Big[\int_a^b \int_a^b \alpha_{ab}(s)\dot{W}(s)K^1_{DS}(t_j, \tau)\dot{W}(\tau)dsd\tau\Big] \\
&\qquad - \sigma^2 E\Big[\int_a^b \int_a^b \alpha_{ab}(\tau)\dot{W}(\tau)K^1_{DS}(t_i, s)\dot{W}(s)dsd\tau\Big] \\
&\qquad + \sigma^2 E\Big[\int_a^b \int_a^b K^1_{DS}(t_i, s)K^1_{DS}(t_j, \tau)\dot{W}(s)\dot{W}(\tau)dsd\tau\Big]
\end{aligned} \tag{4.83}
$$

$$
\begin{aligned}
&= \sigma^2 \int_a^b \int_a^b \alpha_{ab}(s)\alpha_{ab}(\tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau \\
&\qquad - \sigma^2 \int_a^b \int_a^b \alpha_{ab}(s)K^1_{DS}(t_j, \tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau \\
&\qquad - \sigma^2 \int_a^b \int_a^b \alpha_{ab}(\tau)K^1_{DS}(t_i, s)E\Big[\dot{W}(\tau)\dot{W}(s)\Big]dsd\tau \\
&\qquad + \sigma^2 \int_a^b \int_a^b K^1_{DS}(t_i, s)K^1_{DS}(t_j, \tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau
\end{aligned} \tag{4.84}
$$

Applying the property (4.46) in the above equation.

$$
\begin{aligned}
&= \sigma^2 \int_a^b \int_a^b \alpha_{ab}(s)\alpha_{ab}(\tau)\delta(s-\tau)dsd\tau - \sigma^2 \int_a^b \int_a^b \alpha_{ab}(s)K^1_{DS}(t_j,\tau)\delta(s-\tau)dsd\tau \\
&\quad - \sigma^2 \int_a^b \int_a^b \alpha_{ab}(\tau)K^1_{DS}(t_i,s)\delta(s-\tau)dsd\tau \\
&\quad + \sigma^2 \int_a^b \int_a^b K^1_{DS}(t_i,s)K^1_{DS}(t_j,\tau)\delta(s-\tau)dsd\tau
\end{aligned} \tag{4.85}
$$

Applying the property (4.51) in the above equation.

$$
\begin{aligned}
&= \sigma^2 \int_a^b \alpha_{ab}(s)\alpha_{ab}(s)ds - \sigma^2 \int_a^b \alpha_{ab}(s)K^1_{DS}(t_j,s)ds \\
&\quad - \sigma^2 \int_a^b \alpha_{ab}(s)K^1_{DS}(t_i,s)ds + \sigma^2 \int_a^b K^1_{DS}(t_i,s)K^1_{DS}(t_j,s)ds
\end{aligned} \tag{4.86}
$$

$$
\begin{aligned}
Cov[e_3(t_i), e_3(t_j)] &= E[e_3(t_i)e_3(t_j)] \\
&= \sigma^2 E\Big[\Big[\int_a^b \alpha_{ab}(s)(t_i-s)\dot{W}(s)ds - \int_a^b K^2_{DS}(t_i,s)\dot{W}(s)ds\Big] \\
&\qquad \Big[\int_a^b \alpha_{ab}(\tau)(t_j-\tau)\dot{W}(\tau)d\tau - \int_a^b K^2_{DS}(t_j,\tau)\dot{W}(\tau)d\tau\Big]\Big]
\end{aligned} \tag{4.87}
$$

$$
\begin{aligned}
&= \sigma^2 E\Big[\int_a^b \int_a^b \alpha_{ab}(s)\alpha_{ab}(\tau)(t_i-s)(t_j-\tau)\dot{W}(s)\dot{W}(\tau)dsd\tau\Big] \\
&\quad - \sigma^2 E\Big[\int_a^b \int_a^b \alpha_{ab}(s)(t_i-s)\dot{W}(s)K^2_{DS}(t_j,\tau)\dot{W}(\tau)dsd\tau\Big] \\
&\quad - \sigma^2 E\Big[\int_a^b \int_a^b \alpha_{ab}(\tau)(t_j-\tau)\dot{W}(\tau)K^2_{DS}(t_i,s)\dot{W}(s)dsd\tau\Big] \\
&\quad + \sigma^2 E\Big[\int_a^b \int_a^b K^2_{DS}(t_i,s)K^2_{DS}(t_j,\tau)\dot{W}(s)\dot{W}(\tau)dsd\tau\Big]
\end{aligned} \tag{4.88}
$$

$$
\begin{aligned}
&= \sigma^2 \int_a^b \int_a^b \alpha_{ab}(s)\alpha_{ab}(\tau)(t_i - s)(t_j - \tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau \\
&\quad - \sigma^2 \int_a^b \int_a^b \alpha_{ab}(s)(t_i - s)K_{DS}^2(t_j, \tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau \\
&\quad - \sigma^2 \int_a^b \int_a^b \alpha_{ab}(\tau)(t_j - \tau)K_{DS}^2(t_i, s)E\Big[\dot{W}(\tau)\dot{W}(s)\Big]dsd\tau \\
&\quad + \sigma^2 \int_a^b \int_a^b K_{DS}^2(t_i, s)K_{DS}^2(t_j, \tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau
\end{aligned} \tag{4.89}
$$

Applying the property (4.46) in the above equation.

$$
\begin{aligned}
&= \sigma^2 \int_a^b \int_a^b \alpha_{ab}(s)\alpha_{ab}(\tau)(t_i - s)(t_j - \tau)\delta(s - \tau)dsd\tau \\
&\quad - \sigma^2 \int_a^b \int_a^b \alpha_{ab}(s)(t_i - s)K_{DS}^2(t_j, \tau)\delta(s - \tau)dsd\tau \\
&\quad - \sigma^2 \int_a^b \int_a^b \alpha_{ab}(\tau)(t_j - \tau)K_{DS}^2(t_i, s)\delta(s - \tau)dsd\tau \\
&\quad + \sigma^2 \int_a^b \int_a^b K_{DS}^2(t_i, s)K_{DS}^2(t_j, \tau)\delta(s - \tau)dsd\tau
\end{aligned} \tag{4.90}
$$

Applying the property (4.51) in the above equation.

$$
\begin{aligned}
&= \sigma^2 \int_a^b \alpha_{ab}(s)\alpha_{ab}(s)(t_i - s)(t_j - s)ds - \sigma^2 \int_a^b \alpha_{ab}(s)(t_i - s)K_{DS}^2(t_j, s)ds \\
&\quad - \sigma^2 \int_a^b \alpha_{ab}(s)(t_j - s)K_{DS}^2(t_i, s)ds + \sigma^2 \int_a^b K_{DS}^2(t_i, s)K_{DS}^2(t_j, s)ds
\end{aligned} \tag{4.91}
$$

$$Cov[e_1(t_i), e_2(t_j)] = Cov[e_2(t_i), e_1(t_j)] = E[e_1(t_i)e_2(t_j)]$$
$$= \sigma^2 E\Big[\Big[\alpha_{ab}(t_i)\dot{W}(t_i) - \int_a^b K^1_{DS}(t_i,s)\dot{W}(s)ds\Big]\Big[\int_a^b \alpha_{ab}(\tau)\dot{W}(\tau)d\tau - \int_a^b K^2_{DS}(t_j,\tau)\dot{W}(\tau)d\tau\Big]\Big] \tag{4.92}$$

$$\begin{aligned}
&= \sigma^2 E\Big[\int_a^b \alpha_{ab}(t_i)\alpha_{ab}(\tau)\dot{W}(t_i)\dot{W}(\tau)d\tau\Big] - \sigma^2 E\Big[\int_a^b \alpha_{ab}(t_i)\dot{W}(t_i)K^2_{DS}(t_j,\tau)\dot{W}(\tau)d\tau\Big] \\
&\quad - \sigma^2 E\Big[\int_a^b\int_a^b \alpha_{ab}(\tau)\dot{W}(\tau)K^1_{DS}(t_i,s)\dot{W}(s)ds\Big] \\
&\quad + \sigma^2 E\Big[\int_a^b\int_a^b K^1_{DS}(t_i,s)K^2_{DS}(t_j,\tau)\dot{W}(s)\dot{W}(\tau)dsd\tau\Big]
\end{aligned} \tag{4.93}$$

$$\begin{aligned}
&= \sigma^2 \int_a^b \alpha_{ab}(t_i)\alpha_{ab}(\tau)E\Big[\dot{W}(t_i)\dot{W}(\tau)\Big]d\tau - \sigma^2\int_a^b \alpha_{ab}(t_i)K^2_{DS}(t_j,\tau)E\Big[\dot{W}(t_i)\dot{W}(\tau)\Big]d\tau \\
&\quad - \sigma^2\int_a^b\int_a^b \alpha_{ab}(\tau)K^1_{DS}(t_i,s)E\Big[\dot{W}(\tau)\dot{W}(s)\Big]dsd\tau \\
&\quad + \sigma^2\int_a^b\int_a^b K^1_{DS}(t_i,s)K^2_{DS}(t_j,\tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau
\end{aligned} \tag{4.94}$$

Applying the property (4.46) in the above equation.

$$\begin{aligned}
&= \sigma^2 \int_a^b \alpha_{ab}(t_i)\alpha_{ab}(\tau)\delta(t_i-\tau)d\tau - \sigma^2\int_a^b \alpha_{ab}(t_i)K^2_{DS}(t_j,\tau)\delta(t_i-\tau)d\tau \\
&\quad - \sigma^2\int_a^b\int_a^b \alpha_{ab}(\tau)K^1_{DS}(t_i,s)\delta(s-\tau)dsd\tau \\
&\quad + \sigma^2\int_a^b\int_a^b K^1_{DS}(t_i,s)K^2_{DS}(t_j,\tau)\delta(s-\tau)dsd\tau
\end{aligned} \tag{4.95}$$

Applying the property (4.51) in the above equation.

$$\begin{aligned}
&= \sigma^2\alpha_{ab}(t_i)\alpha_{ab}(t_i) - \sigma^2\alpha_{ab}(t_i)K^2_{DS}(t_i,t_j) - \sigma^2\int_a^b \alpha_{ab}(\tau)K^1_{DS}(t_i,\tau)d\tau \\
&\quad + \sigma^2\int_a^b K^1_{DS}(t_i,s)K^2_{DS}(t_j,s)ds
\end{aligned} \tag{4.96}$$

$$Cov[e_1(t_i), e_3(t_j)] = Cov[e_3(t_i), e_1(t_j)] = E[e_1(t_i)e_3(t_j)]$$
$$= \sigma^2 E\Bigg[\Big[\alpha_{ab}(t_i)\dot{W}(t_i) - \int_a^b K_{DS}^1(t_i,s)\dot{W}(s)ds\Big]\Big[\int_a^b \alpha_{ab}(\tau)(t_j-\tau)\dot{W}(\tau)d\tau$$
$$- \int_a^b K_{DS}^3(t_j,\tau)\dot{W}(\tau)d\tau\Big]\Bigg] \tag{4.97}$$
$$= \sigma^2 E\Big[\int_a^b \alpha_{ab}(t_i)\alpha_{ab}(\tau)(t_j-\tau)\dot{W}(t_i)\dot{W}(\tau)d\tau\Big] - \sigma^2 E\Big[\int_a^b \alpha_{ab}(t_i)\dot{W}(t_i)K_{DS}^3(t_j,\tau)\dot{W}(\tau)d\tau\Big]$$
$$- \sigma^2 E\Big[\int_a^b\int_a^b \alpha_{ab}(\tau)(t_j-\tau)\dot{W}(\tau)K_{DS}^1(t_i,s)\dot{W}(s)ds\Big]$$
$$+ \sigma^2 E\Big[\int_a^b\int_a^b K_{DS}^1(t_i,s)K_{DS}^3(t_j,\tau)\dot{W}(s)\dot{W}(\tau)dsd\tau\Big] \tag{4.98}$$
$$= \sigma^2 \int_a^b \alpha_{ab}(t_i)\alpha_{ab}(\tau)(t_j-\tau)E\Big[\dot{W}(t_i)\dot{W}(\tau)\Big]d\tau - \sigma^2\int_a^b \alpha_{ab}(t_i)K_{DS}^3(t_j,\tau)E\Big[\dot{W}(t_i)\dot{W}(\tau)\Big]d\tau$$
$$- \sigma^2\int_a^b\int_a^b \alpha_{ab}(\tau)(t_j-\tau)K_{DS}^1(t_i,s)E\Big[\dot{W}(\tau)\dot{W}(s)\Big]dsd\tau$$
$$+ \sigma^2\int_a^b\int_a^b K_{DS}^1(t_i,s)K_{DS}^3(t_j,\tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau \tag{4.99}$$

Applying the property (4.46) in the above equation.

$$= \sigma^2 \int_a^b \alpha_{ab}(t_i)\alpha_{ab}(\tau)(t_j-\tau)\delta(t_i-\tau)d\tau - \sigma^2\int_a^b \alpha_{ab}(t_i)K_{DS}^3(t_j,\tau)\delta(t_i-\tau)d\tau$$
$$- \sigma^2\int_a^b\int_a^b \alpha_{ab}(\tau)(t_j-\tau)K_{DS}^1(t_i,s)\delta(s-\tau)dsd\tau$$
$$+ \sigma^2\int_a^b\int_a^b K_{DS}^1(t_i,s)K_{DS}^3(t_j,\tau)\delta(s-\tau)dsd\tau \tag{4.100}$$

Applying the property (4.51) in the above equation.

$$= \sigma^2\alpha_{ab}(t_i)\alpha_{ab}(t_i)(t_j-t_i) - \sigma^2\alpha_{ab}(t_i)K_{DS}^3(t_i,t_j)$$
$$- \sigma^2\int_a^b \alpha_{ab}(s)(t_j-s)K_{DS}^1(t_i,s)ds + \sigma^2\int_a^b K_{DS}^1(t_i,s)K_{DS}^3(t_j,s)ds \tag{4.101}$$

$$
\begin{aligned}
Cov[e_2(t_i), e_3(t_j)] &= Cov[e_3(t_i), e_2(t_j)] = E[e_2(t_i)e_3(t_j)] \\
&= \sigma^2 E\Big[\Big[\int_a^b \alpha_{ab}(s)\dot{W}(s)ds - \int_a^b K_{DS}^2(t_i,s)\dot{W}(s)ds\Big] \\
&\qquad \Big[\int_a^b \alpha_{ab}(\tau)(t_j-\tau)\dot{W}(\tau)d\tau - \int_a^b K_{DS}^3(t_j,\tau)\dot{W}(\tau)d\tau\Big]\Big] \qquad (4.102) \\
&= \sigma^2 E\Big[\int_a^b\int_a^b \alpha_{ab}(s)\alpha_{ab}(\tau)(t_j-\tau)\dot{W}(s)\dot{W}(\tau)dsd\tau\Big] \\
&\quad - \sigma^2 E\Big[\int_a^b\int_a^b \alpha_{ab}(s)\dot{W}(s)K_{DS}^3(t_j,\tau)\dot{W}(\tau)dsd\tau\Big] \\
&\quad - \sigma^2 E\Big[\int_a^b\int_a^b \alpha_{ab}(\tau)(t_j-\tau)\dot{W}(\tau)K_{DS}^2(t_i,s)\dot{W}(s)dsd\tau\Big] \\
&\quad + \sigma^2 E\Big[\int_a^b\int_a^b K_{DS}^2(t_i,s)K_{DS}^3(t_j,\tau)\dot{W}(s)\dot{W}(\tau)dsd\tau\Big] \qquad (4.103) \\
&= \sigma^2 \int_a^b\int_a^b \alpha_{ab}(s)\alpha_{ab}(\tau)(t_j-\tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau \\
&\quad - \sigma^2 \int_a^b\int_a^b \alpha_{ab}(s)K_{DS}^3(t_j,\tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau \\
&\quad - \sigma^2 \int_a^b\int_a^b \alpha_{ab}(\tau)(t_j-\tau)K_{DS}^2(t_i,s)E\Big[\dot{W}(\tau)\dot{W}(s)\Big]dsd\tau \\
&\quad + \sigma^2 \int_a^b\int_a^b K_{DS}^2(t_i,s)K_{DS}^3(t_j,\tau)E\Big[\dot{W}(s)\dot{W}(\tau)\Big]dsd\tau \qquad (4.104)
\end{aligned}
$$

Applying the property (4.46) in the above equation.

$$
\begin{aligned}
&= \sigma^2 \int_a^b\int_a^b \alpha_{ab}(s)\alpha_{ab}(\tau)(t_j-\tau)\delta(s-\tau)dsd\tau \\
&\quad - \sigma^2 \int_a^b\int_a^b \alpha_{ab}(s)K_{DS}^3(t_j,\tau)\delta(s-\tau)dsd\tau \\
&\quad - \sigma^2 \int_a^b\int_a^b \alpha_{ab}(\tau)(t_j-\tau)K_{DS}^2(t_i,s)\delta(s-\tau)dsd\tau \\
&\quad + \sigma^2 \int_a^b\int_a^b K_{DS}^2(t_i,s)K_{DS}^3(t_j,\tau)\delta(s-\tau)dsd\tau \qquad (4.105)
\end{aligned}
$$

Applying the property (4.51) in the above equation.

$$
\begin{aligned}
&= \sigma^2 \int_a^b \alpha_{ab}(s)\alpha_{ab}(s)(t_j - s)ds - \sigma^2 \int_a^b \alpha_{ab}(s) K_{DS}^3(t_j, s)ds \\
&\qquad - \sigma^2 \int_a^b \alpha_{ab}(s)(t_j - s) K_{DS}^2(t_i, s)ds + \sigma^2 \int_a^b K_{DS}^2(t_i, s) K_{DS}^3(t_j, s)ds
\end{aligned} \tag{4.106}
$$

Covariance matrix=

$$
\begin{bmatrix}
Cov[e_1(t_i), e_1(t_j)] & Cov[e_1(t_i), e_2(t_j)] & Cov[e_1(t_i), e_3(t_j)] \\
Cov[e_2(t_i), e_1(t_j)] & Cov[e_2(t_i), e_2(t_j)] & Cov[e_2(t_i), e_3(t_j)] \\
Cov[e_3(t_i), e_1(t_j)] & Cov[e_3(t_i), e_2(t_j)] & Cov[e_3(t_i), e_3(t_j)]
\end{bmatrix} \tag{4.107}
$$

The elements of the covariance matrix are not constant; hence the error terms are heteroskedastic. The covariance matrix is updated in every step in the recursive GLS estimator.

## 4.4 Recursive GLS algorithm

The covariance matrix depends on the unknown variance $\sigma^2$, and the unknown parameter vector $\bar{a}$ in the $K_{DS}$ kernels. Hence the standard GLS cannot be applied directly. This paves the way to employ the modified recursive GLS [42] [14] in which the covariance matrix is estimated progressively. Letting $Q_i - P_i\bar{a}$ denote the regression error $e_i$ in batch $i$, the modified recursive GLS algorithm computes

$$
\hat{a}_k = \arg\min_{\bar{a}} \left( \sum_{i=1}^{k} (Q_i - P_i\bar{a})^T S_i (Q_i - P_i\bar{a}) \right) \tag{4.108}
$$

where $\hat{a}_k$ is the parameter estimate update at iteration $k$ of the algorithm. Each weighting matrix $S_{k+1}$, is calculated as the inverse of the covariance matrix based on the parameter and variance estimates i.e. $\hat{a}_k$ and $\sigma^2$, obtained from the residual trajectory $y_M(t) - y_E(t)$ in previous iteration $k$, where $y_E$ signifies the estimated output. A modified recursive GLS algorithm with inverse covariance weighting is explained below

At iteration $k+1$, the algorithm strives to minimize

$$\begin{aligned} &\min(\bar{e}_{k+1}^T \bar{S}_{k+1} \bar{e}_{k+1}) \\ \text{subject to:} \quad &\bar{Q}_{k+1} = \bar{P}_{k+1}\bar{a}_{k+1} + \bar{e}_{k+1} \end{aligned} \tag{4.109}$$

where

$$\bar{Q}_{k+1} = \begin{bmatrix} Q_0 \\ Q_1 \\ \vdots \\ Q_{k+1} \end{bmatrix}; \quad \bar{P}_{k+1} = \begin{bmatrix} P_0 \\ P_1 \\ \vdots \\ P_{k+1} \end{bmatrix}; \quad \bar{e} = \begin{bmatrix} e_0 \\ e_1 \\ \vdots \\ e_{k+1} \end{bmatrix}$$

and

$$\bar{S}_{k+1} = \mathrm{diag}(S_0, S_1, \ldots, S_{k+1}) \tag{4.110}$$

The solution of the above is written as

$$(\bar{P}_{k+1}^T \bar{S}_{k+1} \bar{P}_{k+1})\hat{a}_{k+1} = \bar{P}_{k+1}^T \bar{S}_{k+1} \bar{Q}_{k+1} \tag{4.111}$$

or in summation form as

$$\left( \sum_{i=0}^{k+1} P_i^T S_i P_i \right) \hat{a}_{k+1} = \sum_{i=0}^{k+1} P_i^T S_i Q_i \tag{4.112}$$

Defining

$$M_{k+1} = \sum_{i=0}^{k+1} P_i^T S_i P_i \tag{4.113}$$

the recursion for $M_{k+1}$ is:

$$M_{k+1} = M_k + P_{k+1}^T S_{k+1} P_{k+1} \tag{4.114}$$

Rearranging (4.112) gives

$$\begin{aligned}\hat{a}_{k+1} &= M_{k+1}^{-1}\left[\left(\sum_{i=0}^{k} P_i^T S_i P_i\right)\hat{a}_k + P_{k+1}^T S_{k+1} Q_{k+1}\right] \\ &= M_{k+1}^{-1}\left[M_k \hat{a}_k + P_{k+1}^T S_{k+1} Q_{k+1}\right] \end{aligned} \tag{4.115}$$

Another form of (4.115) is delivered by the recursion (4.114) and reads

$$\begin{aligned}\hat{a}_{k+1} &= \hat{a}_k - M_{k+1}^{-1}(P_{k+1}^T S_{k+1} P_{k+1}\hat{a}_k - P_{k+1}^T S_{k+1} Q_{k+1}) \\ &= \hat{a}_k + M_{k+1}^{-1} P_{k+1}^T S_{k+1}(Q_{k+1} - P_{k+1}\hat{a}_k) \end{aligned} \tag{4.116}$$

A recursion for $M_{k+1}^{-1}$ is obtained by applying the following identity to the recursion in (4.114)

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1} \tag{4.117}$$

which yields

$$M_{k+1}^{-1} = M_k^{-1} - M_k^{-1} P_{k+1}^T (P_{k+1} M_k^{-1} P_{k+1}^T + S_{k+1}^{-1})^{-1} P_{k+1} M_k^{-1}$$

Defining $R_{k+1} = M_{k+1}^{-1}$ the latter becomes

$$R_{k+1} = R_k - R_k P_{k+1}^T (S_{k+1}^{-1} + P_{k+1} R_k P_{k+1}^T)^{-1} P_{k+1} R_k \tag{4.118}$$

Equations (4.117) and (4.118) constitute the recursive GLS least squares algorithm.

## 4.5 Reconstruction of the output derivatives

Given a measurement process realization $\overline{y_M}$ on $[a, b]$, the derivatives can be reconstructed using,

$$y^{(i)}(t) = \int_a^b K_{DS}^i(t, \tau)\hat{y}(\tau)\mathrm{d}\tau \tag{4.119}$$

where, $K_{DS}^i$ are the kernel representation for output derivatives. In this thesis we consider a third-order system and hence $i = 1$ and 2. The formulae for kernel representation of output derivatives are developed in Chapter 2.

## 4.6 Parameter estimation using kernels with different dimensions

Selection of a model structure is the first step towards the estimation problem. If we lack the prior knowledge of our system including its noise characteristics, the choice of a reasonable model structure may not be obvious. Also, for the selected choice of model structure, the order needs to be specified before its parameters are estimated. In general, our aim is to not use a model order which is higher than what is required [43]. Parameter estimation that is carried out till now involves the assumption that the order of the system is known. But in reality, one of the challenges we can encounter is finding the order of the differential equations. The estimation of the parameters without knowing the proper knowledge of the order of the system will be problematic. The parameter estimation problem is always split into two steps. First, the problem of the model order selection is solved, and then the parameter vector is solved, assuming that the selected order is the same as the true order of the system. Good models can be obtained by comparing different techniques like Akaike Information Criterion (AIC) [44], Corrected AIC (AICc) [45], and Bayesian Information Criterion (BIC) [46].

In this section, we will use kernels with different dimensions to reconstruct and estimate the parameters of the same dimension homogenous LTI system. First, we will show this by reconstructing a third-order homogenous LTI system using fourth and fifth-order kernels.

### 4.6.1 Kernels for a 4th order homogeneous LTI system

The kernels for the fourth-order system are developed by substituting $n = 4$ in the equations (2.11) and (2.12) by using Theorem 1.

$$
\begin{aligned}
K_{F,y}(4,t,\tau) = {} & 16(\tau-a)^3 - 72(t-\tau)(\tau-a)^2 \\
& + 48(t-\tau)^2(\tau-a) - 4(t-\tau)^3 + a_0[-\frac{1}{6}(t-\tau)^3(\tau-a)^4] \\
& + a_1[-\frac{1}{2}(t-\tau)^2(\tau-a)^4 + \frac{2}{3}(t-\tau)^3(\tau-a)^3] \\
& + a_2[-(t-\tau)(\tau-a)^4 + 4(t-\tau)^2(\tau-a)^3 - 2(t-\tau)^3(\tau-a)^2] \\
& + a_3[-(\tau-a)^4 + 12(t-\tau)(\tau-a)^3 - 18(t-\tau)^2(\tau-a)^2 \\
& + 4(t-\tau)^3(\tau-a)]
\end{aligned}
\tag{4.120}
$$

$$
\begin{aligned}
K_{B,y}(4,t,\tau) = {} & 16(b-\tau)^3 + 72(t-\tau)(b-\tau)^2 \\
& + 48(t-\tau)^2(b-\tau) + 4(t-\tau)^3 + a_0[\frac{1}{6}(t-\tau)^3(b-\tau)^4] \\
& + a_1[\frac{1}{2}(t-\tau)^2(b-\tau)^4 + \frac{2}{3}(t-\tau)^3(b-\tau)^3] \\
& + a_2[(t-\tau)(b-\tau)^4 + 4(t-\tau)^2(b-\tau)^3 + 2(t-\tau)^3(b-\tau)^2] \\
& + a_3[(b-\tau)^4 + 12(t-\tau)(b-\tau)^3 + 18(t-\tau)^2(b-\tau)^2 \\
& + 4(t-\tau)^3(b-\tau)]
\end{aligned}
\tag{4.121}
$$

with

$$
K_{DS,y}(4,t,\tau) \triangleq \begin{cases} K_{F,y}(4,t,\tau), & \text{for } \tau \leq t \\ K_{B,y}(4,t,\tau), & \text{for } \tau > t \end{cases}
\tag{4.122}
$$

And thus we have $y(t)$ in terms of kernel representation as:

$$
y(t) = \int_a^b K_{DS,y}(t,\tau)y(\tau)\ d\tau
\tag{4.123}
$$

Consider a general third order system as below:

$$y^{(3)}(t) + a_2 y^{(2)}(t) + a_1 y^{(1)}(t) + a_0 y(t) = 0 \tag{4.124}$$

If the equation (4.124) is differentiated, we will have:

$$y^{(4)}(t) + a_2 y^{(3)}(t) + a_1 y^{(2)}(t) + a_0 y^{(1)}(t) = 0 \tag{4.125}$$

Now, we write the two systems in this format (both as $4th$ order systems with different coefficients) :

$$b_4 y^{(4)}(t) + b_3 y^{(3)}(t) + b_2 y^{(2)}(t) + b_1 y^{(1)}(t) + b_0 y(t) = 0 \tag{4.126}$$

If (4.126) represents (4.124) then we would have to identify:

$$b_4 = 0, b_3 = 1, b_2 = a_2, b_1 = a_1, b_0 = a_0 \tag{4.127}$$

If (4.126) represents (4.125) then we would have to identify:

$$b_4 = 1, b_3 = a_3, b_2 = a_2, b_1 = a_1, b_0 = 0 \tag{4.128}$$

if a curve $y^*(t)$ fits (4.124) for $t \in [a, b]$, then it also fits (4.125) in the format of (4.126) with coefficients respectively in (4.127) and (4.128). This would indicate that $y^*$ will be reproduced by a $3rd$ order kernel for (4.124) but also by a $4th$ order kernel using respective coefficients.

### 4.6.2 Kernels for a 5th order homogeneous LTI system

The kernels for the fifth-order system are developed by substituting $n = 5$ in the equations (2.11) and (2.12) by using Theorem 1.

$$\begin{aligned}K_{F,y}(5,t,\tau) &= 25(\tau-a)^4 - 200(t-\tau)(\tau-a)^3 + 300(t-\tau)^2(\tau-a)^2 \\ &-100(t-\tau)(t-\tau)^3 + 5(t-\tau)^4 + a_0[-\frac{1}{24}(t-\tau)^4(\tau-a)^5] \\ &+a_1[-\frac{1}{6}(t-\tau)^3(\tau-a)^5 + \frac{5}{24}(t-\tau)^4(\tau-a)^4] + a_2[-\frac{1}{2}(t-\tau)^2 \\ &(\tau-a)^5 + \frac{10}{6}(t-\tau)^3(\tau-a)^4 - \frac{20}{24}(t-\tau)^4(\tau-a)^3] \\ &+a_3[-(\tau-a)^5(t-\tau) + \frac{15}{2}(t-\tau)^2(\tau-a)^4 - 10(t-\tau)^3(\tau-a)^3 \\ &+\frac{60}{24}(t-\tau)^4(\tau-a)^2] + a_4[-\frac{120}{24}(\tau-a)(t-\tau)^4 + \frac{240}{6}(t-\tau)^3 \\ &(\tau-a)^2 - 60(t-\tau)^2(\tau-a)^3 + 20(t-\tau)(\tau-a)^4 - (\tau-a)^5]\end{aligned} \tag{4.129}$$

$$\begin{aligned}K_{B,y}(5,t,\tau) &= 25(b-\tau)^4 + 200(t-\tau)(b-\tau)^3 + 300(t-\tau)^2(b-\tau)^2 \\ &+100(t-\tau)(t-\tau)^3 + 5(t-\tau)^4 + a_0[\frac{1}{24}(t-\tau)^4(b-\tau)^5] \\ &+a_1[\frac{1}{6}(t-\tau)^3(b-\tau)^5 + \frac{5}{24}(t-\tau)^4(b-\tau)^4] + a_2[\frac{1}{2}(t-\tau)^2 \\ &(b-\tau)^5 + \frac{10}{6}(t-\tau)^3(b-\tau)^4 + \frac{20}{24}(t-\tau)^4(b-\tau)^3] \\ &+a_3[(b-\tau)^5(t-\tau) + \frac{15}{2}(t-\tau)^2(b-\tau)^4 + 10(t-\tau)^3(b-\tau)^3 \\ &+\frac{60}{24}(t-\tau)^4(b-\tau)^2] + a_4[-\frac{120}{24}(b-\tau)(t-\tau)^4 + \frac{240}{6}(t-\tau)^3 \\ &(b-\tau)^2 + 60(t-\tau)^2(b-\tau)^3 + 20(t-\tau)(b-\tau)^4 + (b-\tau)^5]\end{aligned} \tag{4.130}$$

with

$$K_{DS,y}(5,t,\tau) \triangleq \begin{cases} K_{F,y}(5,t,\tau), & \text{for } \tau \leq t \\ K_{B,y}(5,t,\tau), & \text{for } \tau > t \end{cases} \tag{4.131}$$

(4.131) corresponds to the fifth-order kernels. To reproduce a third-order system from the fifth order kernels, we assume that $a_0$ and $a_1$ terms are zero, while the constant term, $a_2$,

$a_3$ and $a_4$ constitute the third-order system.

But this assumption does not work in the presence of noise. The parameter estimation using the higher-order kernels is experimented by the recursive GLS algorithm, and the results are tabulated in the next chapter.

# Chapter 5

# Results

In this chapter, we present the results of our proposed method. To begin with, we consider the unstable system with different noise levels. Then we compare the proposed method to the results of [27] and [24]. Finally, we use kernels with different dimensions to estimate the parameters of a third-order system.

## 5.1 Case study: Unstable system with different noise levels

**Example** 1: Let us consider a third order LTI system as mentioned below.

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -26 & -13 & 0 \end{bmatrix} x \ ; y = x_1 \ ; \ x(0) = [0, 0, 1] \tag{5.1}$$

with its corresponding characteristic equation

$$y^{(3)}(t) + a_2 y^{(2)}(t) + a_1 y^{(1)} + a_0 y(t) = 0 \tag{5.2}$$

The parameters $a_0$, $a_1$, $a_2$ are assumed to be unknown.
The measured realization of the output $y_M$ is obtained by adding Gaussian white noise to the nominal trajectory.

$$y_M = y_T + \epsilon \tag{5.3}$$

where $y_T$ is the true trajectory and $\epsilon$ is a white noise, with

$$\epsilon \sim \mathcal{N}(\mu, \sigma^2) \tag{5.4}$$

We apply Gaussian white noise of zero mean ($\mu = 0$) and variance ranging from $\sigma = 0.5$ to $\sigma = 6$ to the solution of the above system.

| $\sigma$ | signal-to-noise ratio(dB) |
|---|---|
| 1 | 4.2794 |
| 1.5 | 0.8162 |
| 2 | -1.6735 |
| 3 | -5.1032 |
| 4 | -7.6043 |
| 6 | -11.1499 |

**Table 5.1** Noise levels and the signal-to-noise ratio in decibel scale

Employing the proposed method as described in Chapter 4, the estimated system parameters using third-order kernels were found as they are presented in Table (5.2) with the number of sample points $N = 3000$ which were selected randomly from the uniform distribution in $[a, b] = [0, 8]$.

| | $a_0$ | $a_1$ | $a_2$ |
|---|---|---|---|
| True values | 26 | 13 | 0 |
| Estimated Values | 25.9701 | 12.9874 | -0.0012 |

**Table 5.2** True and estimated parameter values from a true output with AWGN $\mu = 0$ and $\sigma = 1.5$, N=3000 using third order kernels

Once the parameter estimates are obtained, we reconstruct the output and its derivatives as mentioned in section 4.5 of chapter 4.
To quantify accuracy, we use Root Mean Square Distance (RMSD), which provides a measure of the distance between true and estimated solutions.

$$RMSD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}[y_T(t_i) - y_M(t_i)]^2} \tag{5.5}$$

Table (5.3) shows how the parameter estimation accuracy varies with increasing noise levels, and the number of sample points N. The samples for regression are taken in batches of N = 1000 and N = 3000 knots taken randomly from a uniform distribution over [a,b] = [0,8].

| Noise | $N$ | Estimted $a$ | $RMSD$ |
|---|---|---|---|
| $\sigma = 1$ | 1000 | 26.0035, 12.9961, 0.0006 | 2.84e-4 |
| | 3000 | 26.0029, 13.0049, -0.0001 | 2.14e-4 |
| $\sigma = 1.5$ | 1000 | 26.0167, 12.9624 , -0.0002 | 3.16e-4 |
| | 3000 | 26.0091, 12.9822 , -0.0001 | 2.91e-4 |
| $\sigma = 2$ | 1000 | 26.0344, 12.5089, -0.0023 | 7.25e-4 |
| | 3000 | 26.0242, 12.4009, -0.0021 | 6.71e-4 |
| $\sigma = 3$ | 1000 | 25.5697, 12.6212, -0.0002 | 2.07e-3 |
| | 3000 | 25.6726, 12.8201, 0.0001 | 1.57e-3 |
| $\sigma = 4$ | 1000 | 23.6565, 13.9919, -0.0631 | 3.16e-3 |
| | 3000 | 27.0470, 12.0477, -0.0711 | 2.89e-3 |
| $\sigma = 6$ | 1000 | 16.7977, 14.7731, 0.1528 | 2.27e-2 |
| | 3000 | 17.1820, 14.7037, 0.1589 | 2.15e-2 |

**Table 5.3** Estimates of parameter values and $RMSD$ for various noise levels and sample size N

The true system parameters are $a_0 = 26, a_1 = 13, a_2 = 0$.
On the pages that follow, figures are provided for $y_M$ vs. $y_T$ (estimated/ reconstructed vs. nominal trajectories) with noise levels specified in Table (5.3), the reconstruction of the system output and its derivatives for the parameter estimates computed.

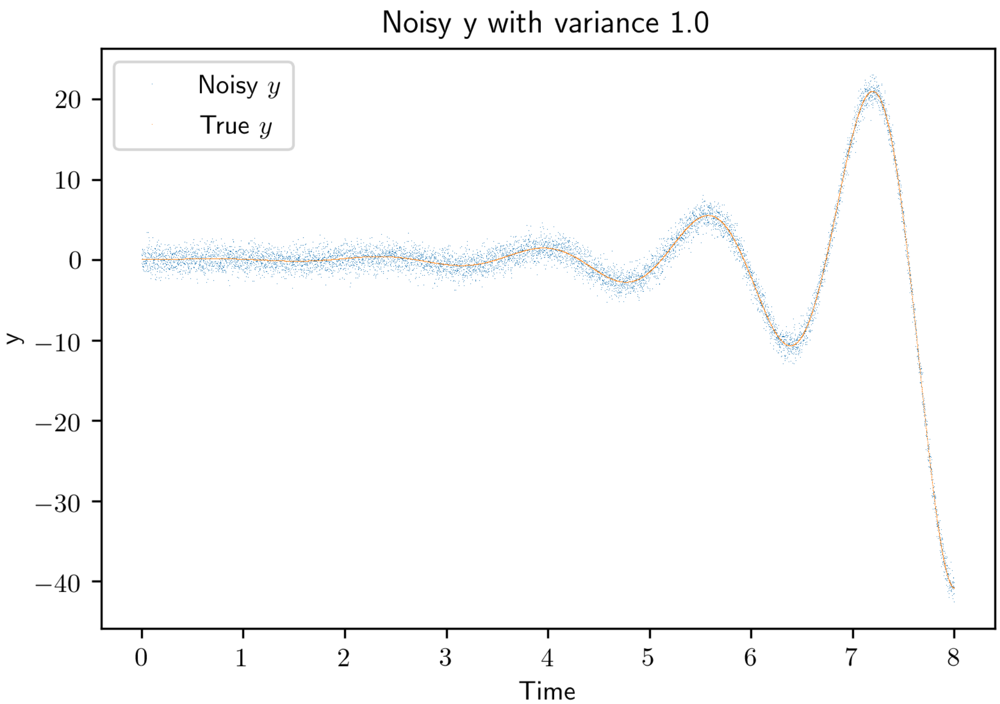


**Figure 5.1** True and noisy system output with AWGN of $\mu = 0$ and $\sigma = 1$ and N=3000

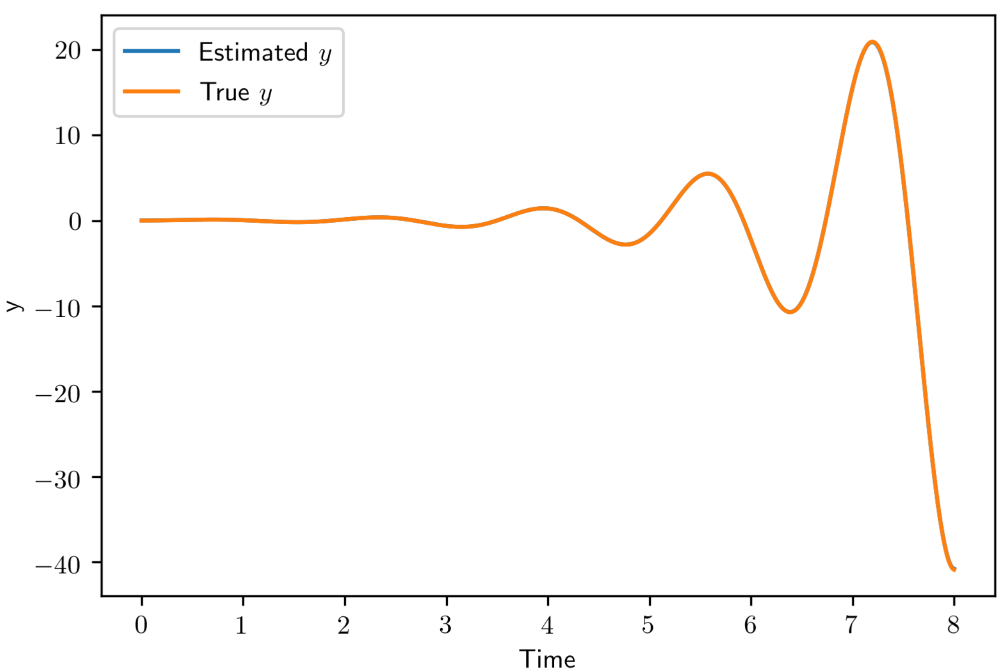


**Figure 5.2** True and reconstructed output trajectories of the system with AWGN of $\mu = 0$ and $\sigma = 1$ and N=3000

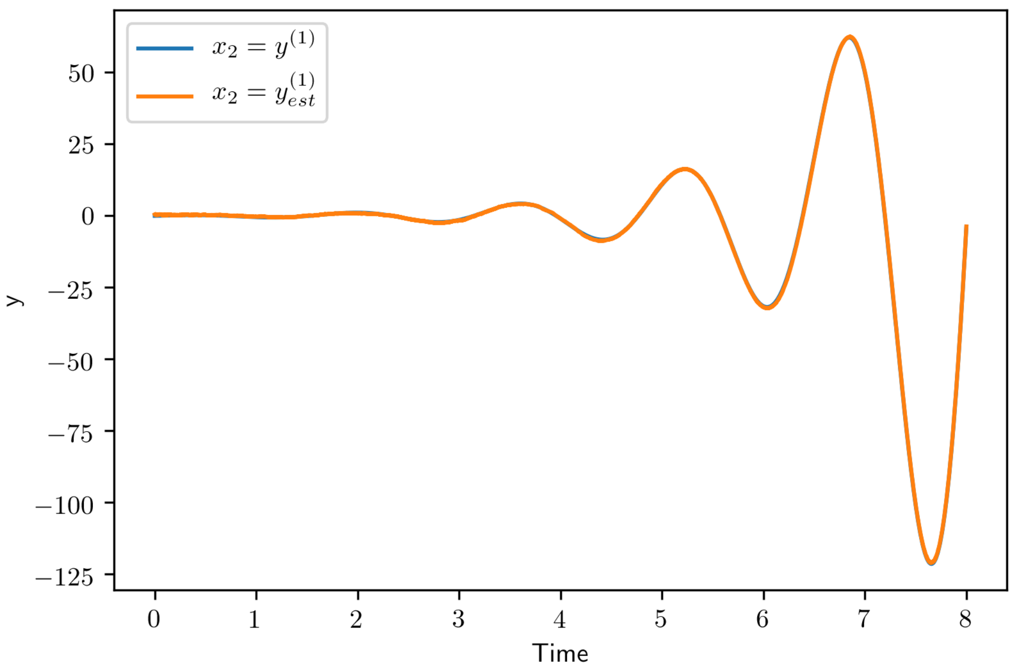


**Figure 5.3** True and reconstructed first derivative of the system with AWGN of $\mu = 0$ and $\sigma = 1$ and N=3000

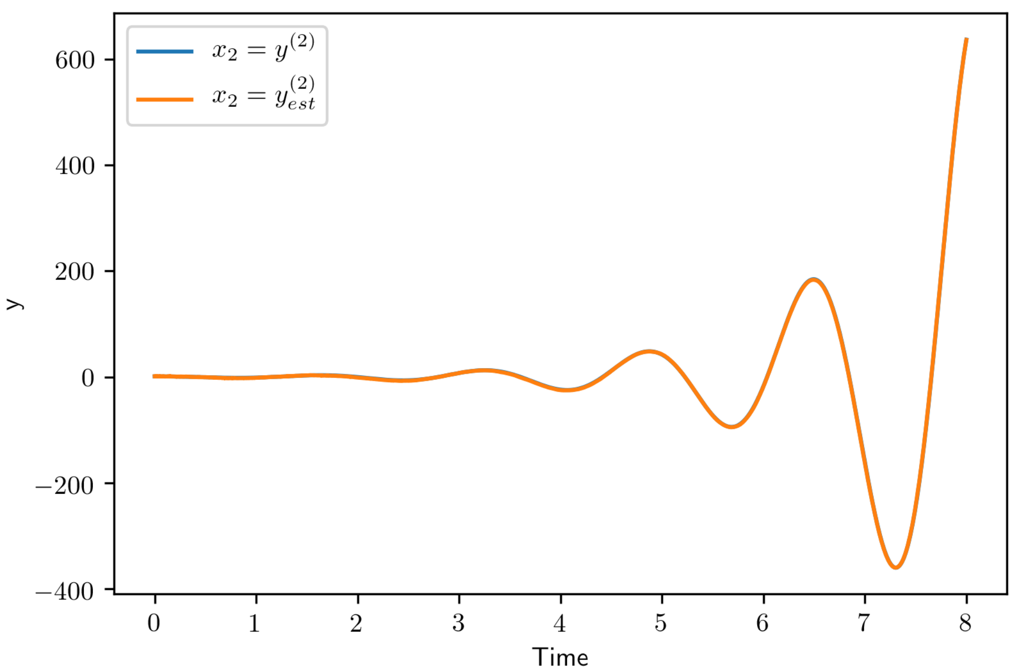


**Figure 5.4** True and reconstructed second derivative of the system with AWGN of $\mu = 0$ and $\sigma = 1$ and N=3000

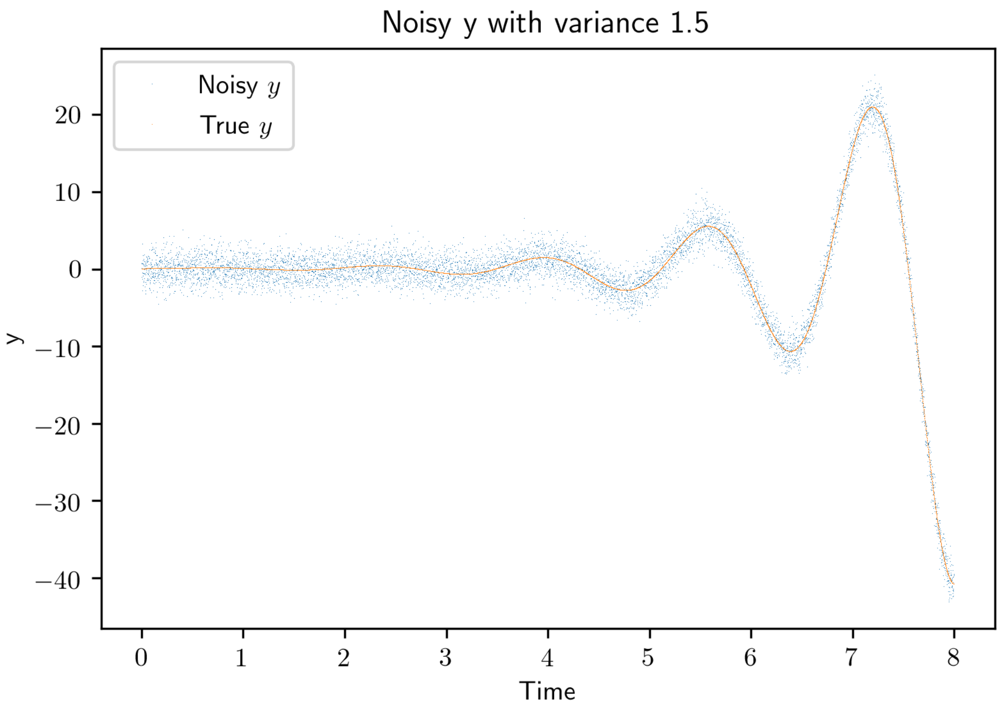


**Figure 5.5** True and noisy system output with AWGN of $\mu = 0$ and $\sigma = 1.5$ and N=3000

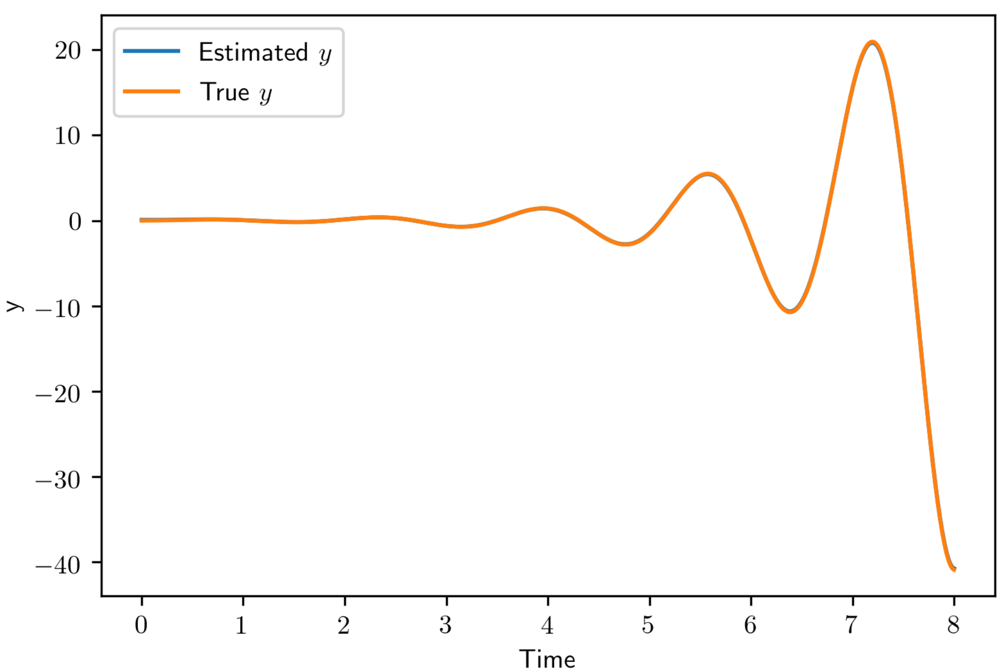


**Figure 5.6** True and reconstructed output trajectories of the system with AWGN of $\mu = 0$ and $\sigma = 1.5$ and N=3000

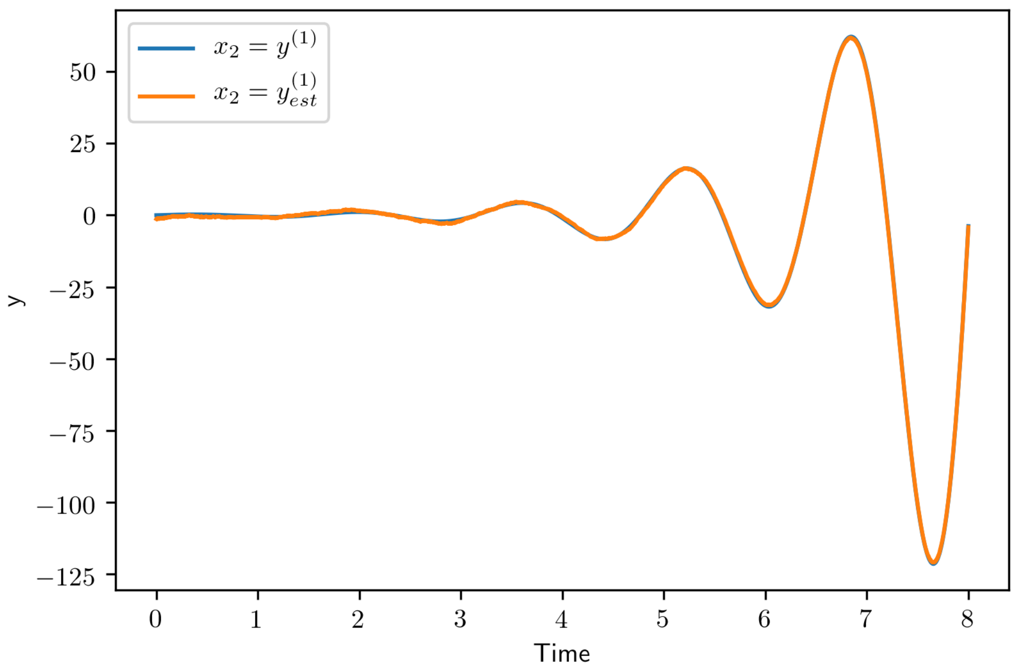


**Figure 5.7** True and reconstructed first derivative of the system with AWGN of $\mu = 0$ and $\sigma = 1.5$ and N=3000

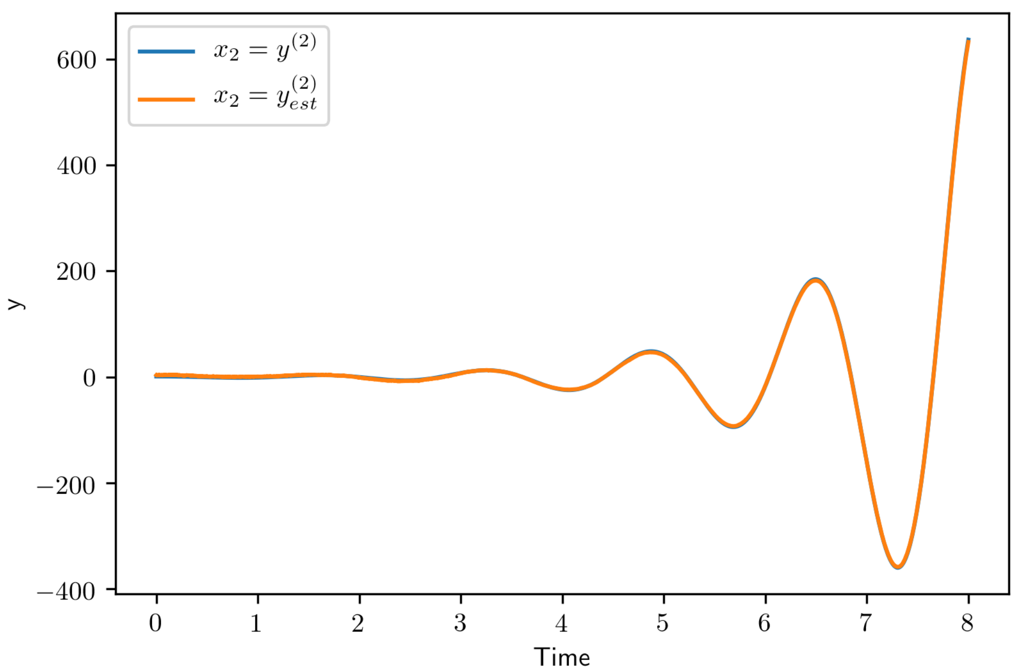


**Figure 5.8** True and reconstructed second derivative of the system with AWGN of $\mu = 0$ and $\sigma = 1.5$ and N=3000

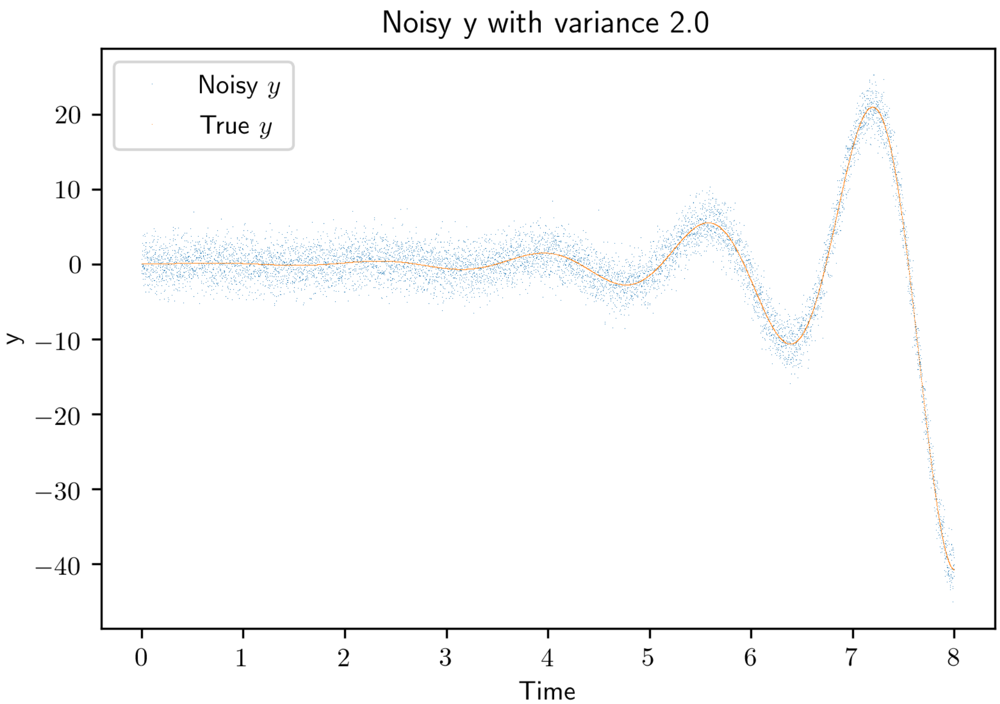


**Figure 5.9** True and noisy system output with AWGN of $\mu = 0$ and $\sigma = 2$ and N=3000

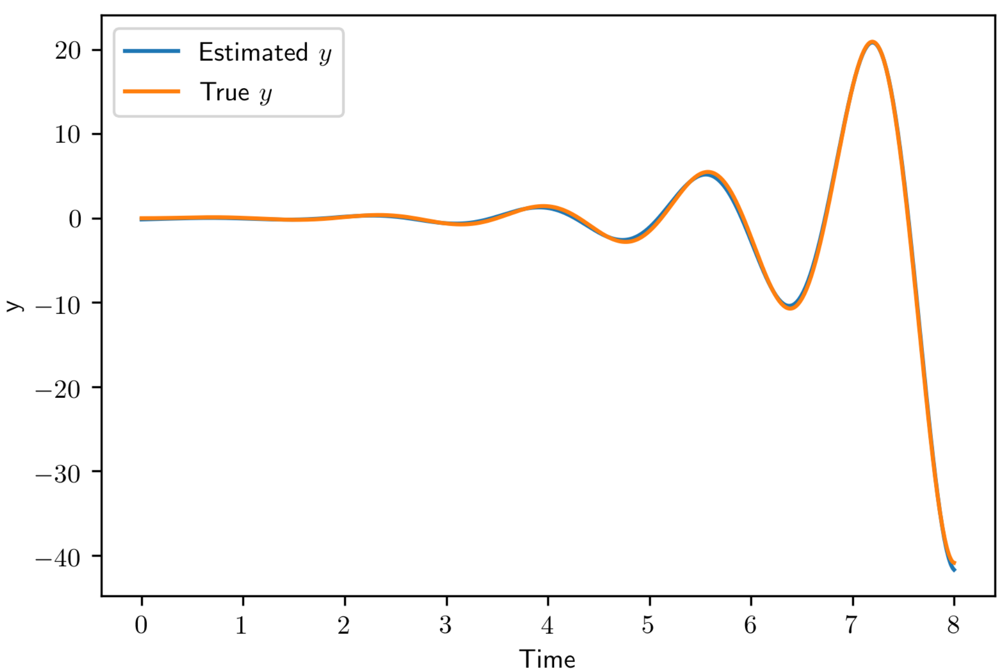


**Figure 5.10** True and reconstructed output trajectories of the system with AWGN of $\mu = 0$ and $\sigma = 2$ and N=3000

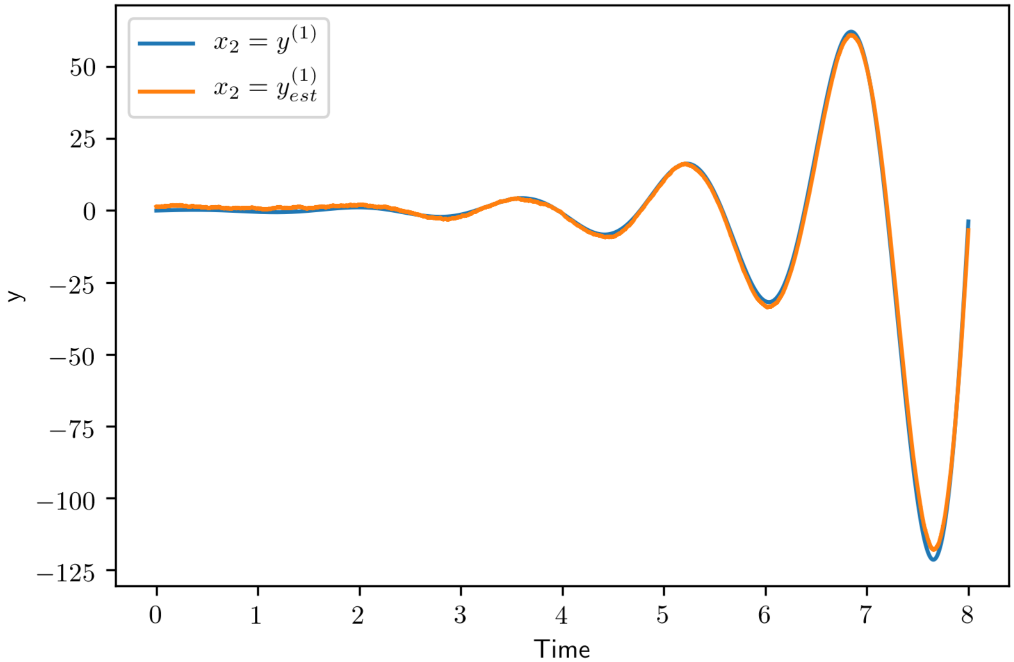


**Figure 5.11** True and reconstructed first derivative of the system with AWGN of $\mu = 0$ and $\sigma = 2$ and N=3000

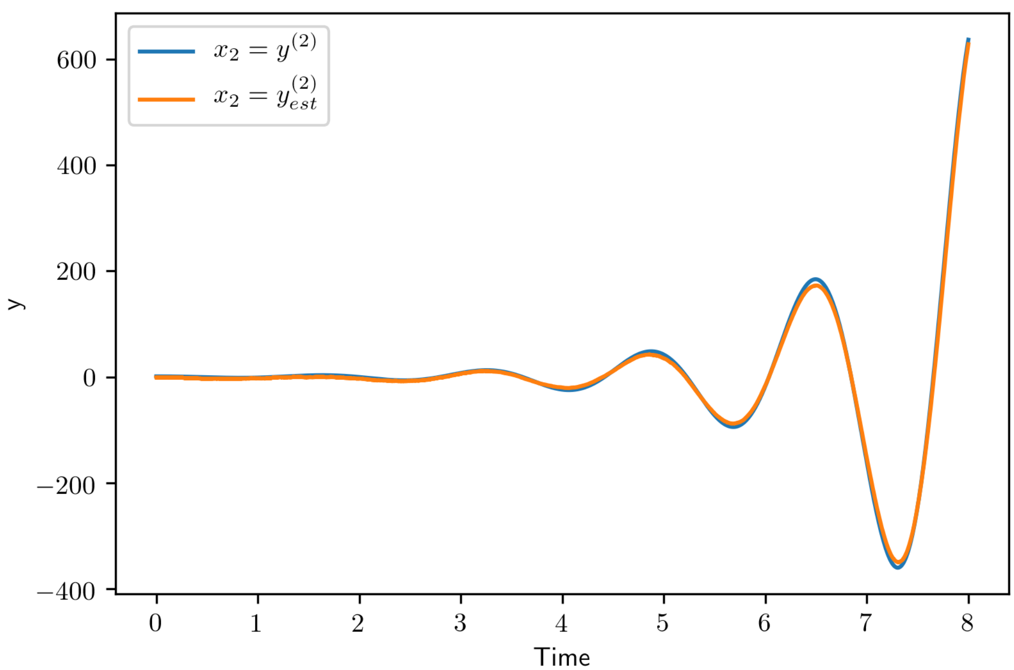


**Figure 5.12** True and reconstructed second derivative of the system with AWGN of $\mu = 0$ and $\sigma = 2$ and N=3000

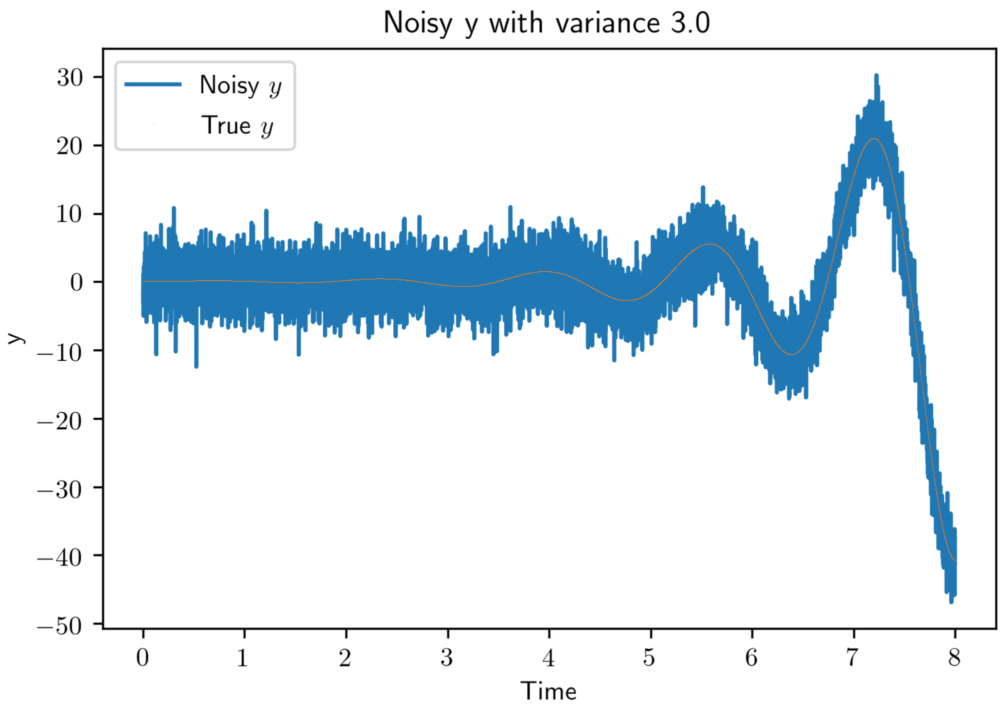


**Figure 5.13** True and noisy system output with AWGN of $\mu = 0$ and $\sigma = 3$ and N=3000

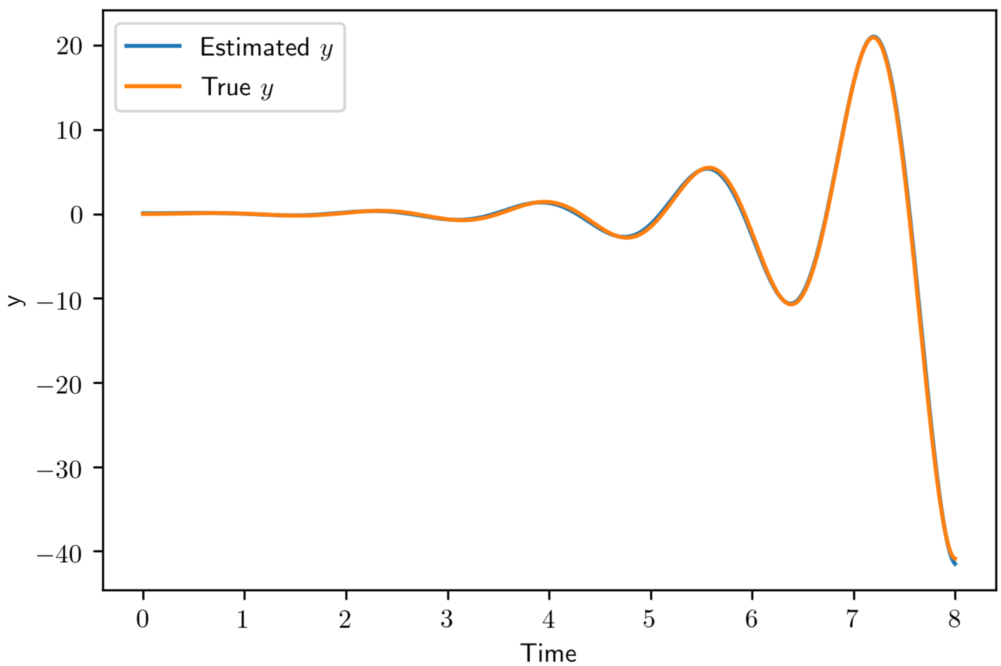


**Figure 5.14** True and reconstructed output trajectories of the system with AWGN of $\mu = 0$ and $\sigma = 3$ and N=3000

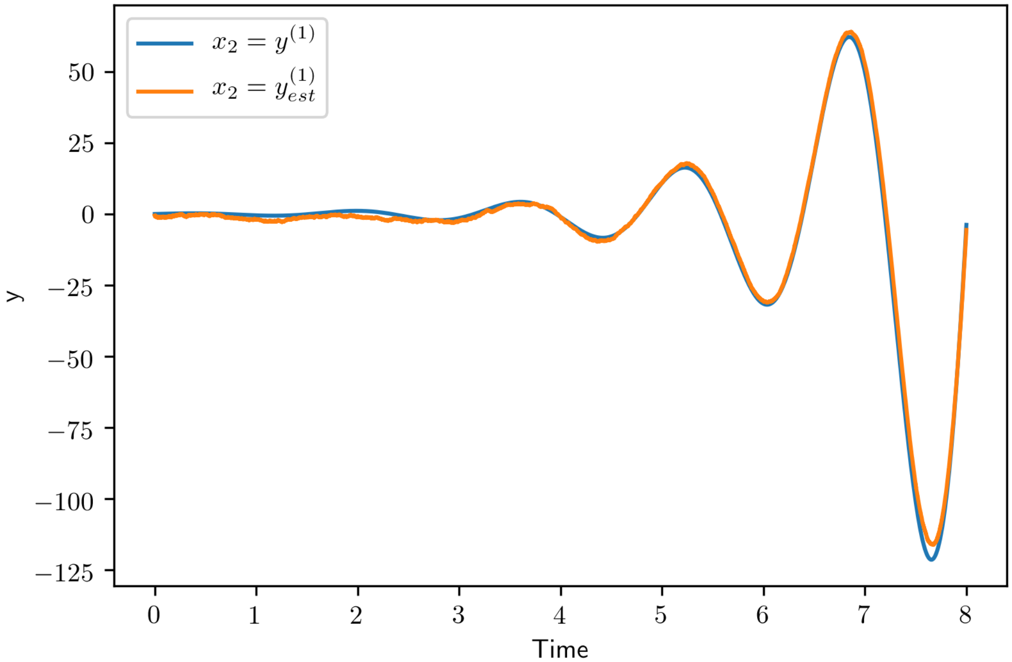


**Figure 5.15** True and reconstructed first derivative of the system with AWGN of $\mu = 0$ and $\sigma = 3$ and N=3000

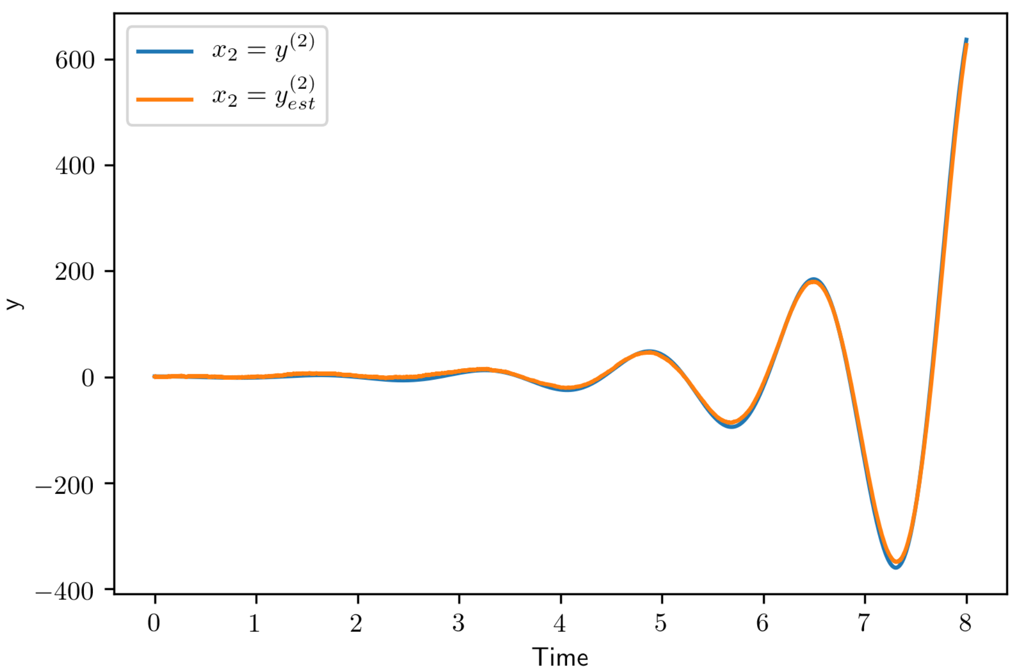


**Figure 5.16** True and reconstructed second derivative of the system with AWGN of $\mu = 0$ and $\sigma = 3$ and N=3000

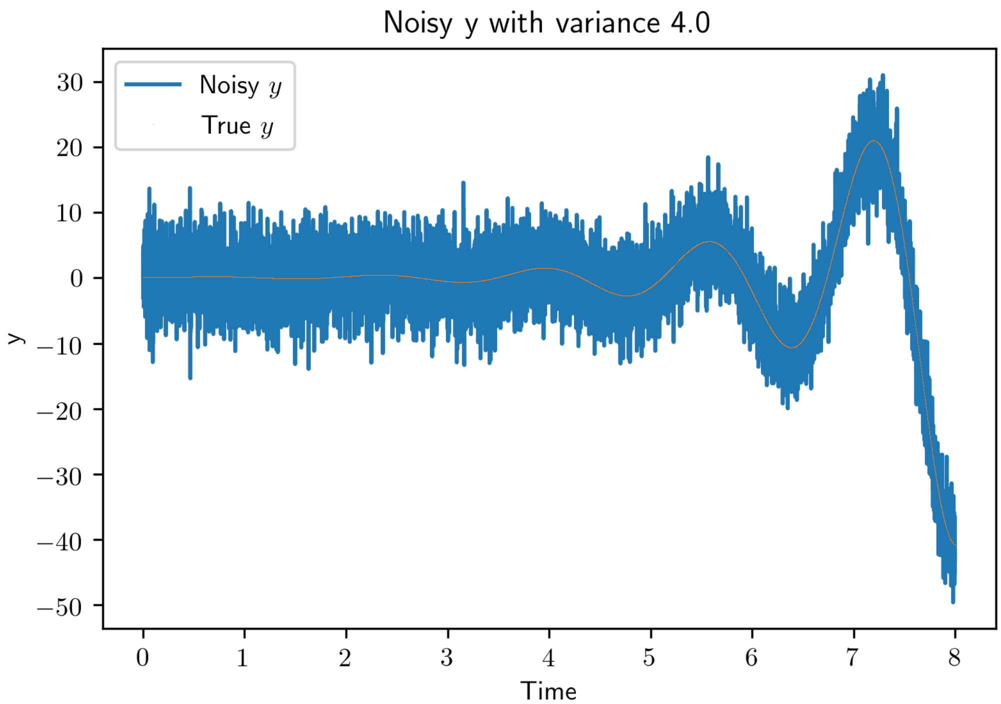


**Figure 5.17** True and noisy system output with AWGN of $\mu = 0$ and $\sigma = 4$ and N=3000

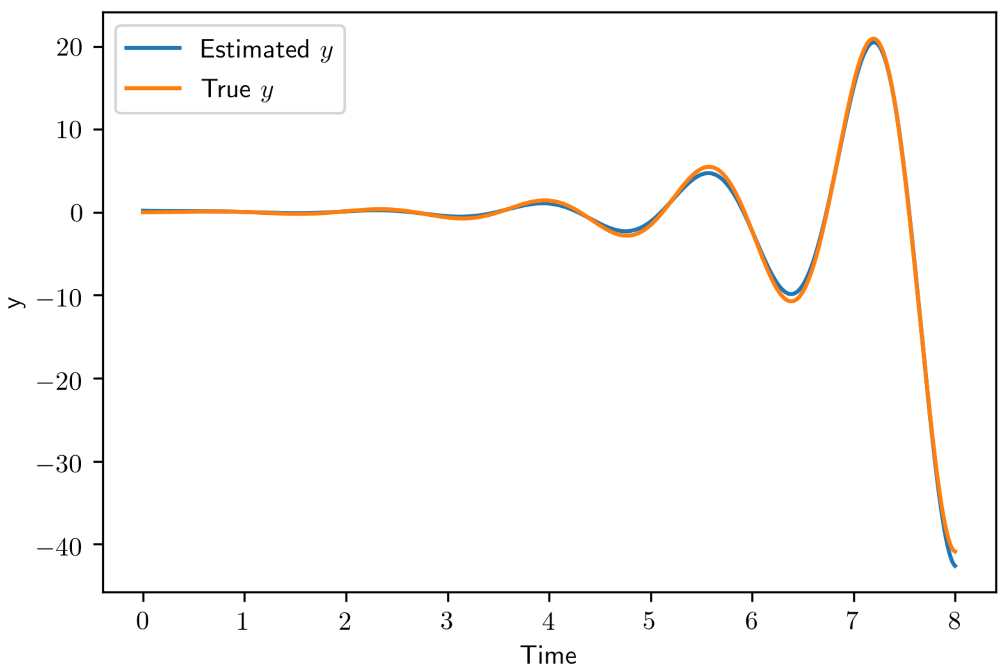


**Figure 5.18** True and reconstructed output trajectories of the system with AWGN of $\mu = 0$ and $\sigma = 4$ and N=3000

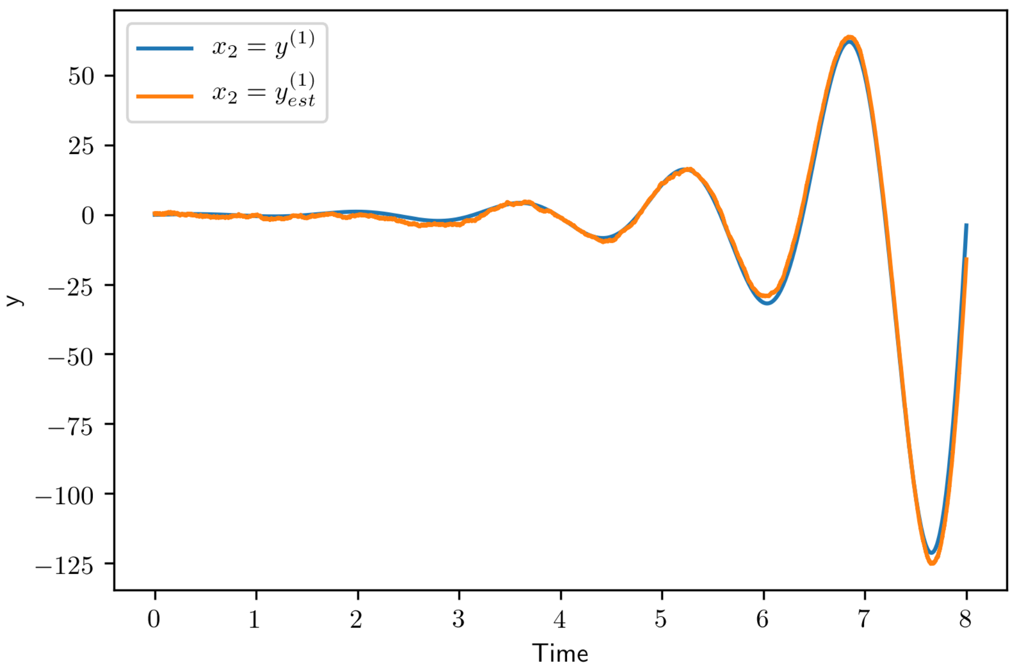


**Figure 5.19** True and reconstructed first derivative of the system with AWGN of $\mu = 0$ and $\sigma = 4$ and N=3000

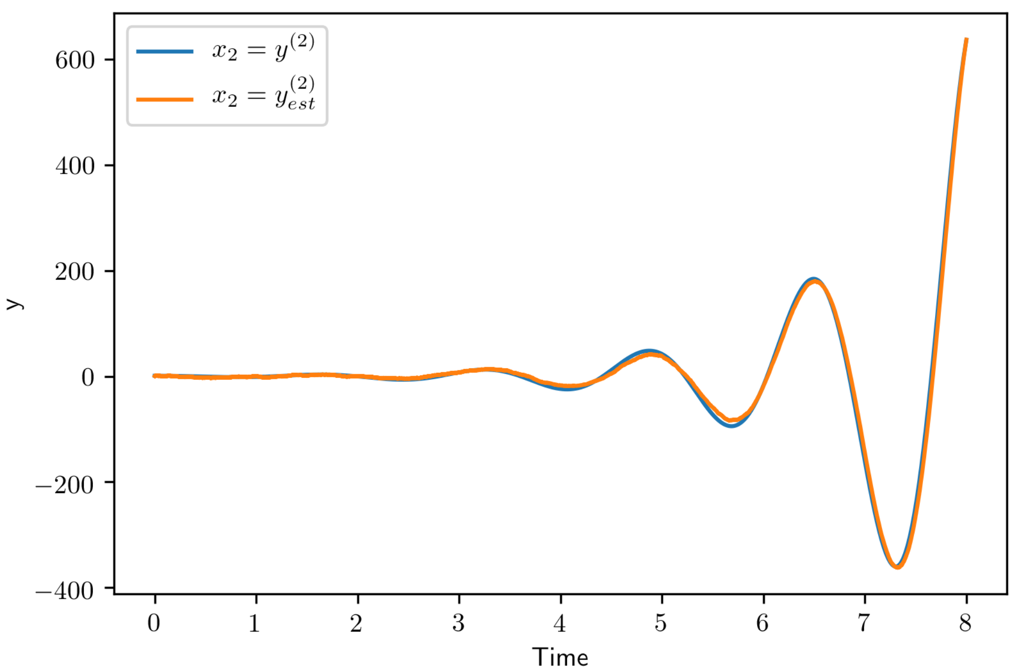


**Figure 5.20** True and reconstructed second derivative of the system with AWGN of $\mu = 0$ and $\sigma = 4$ and N=3000

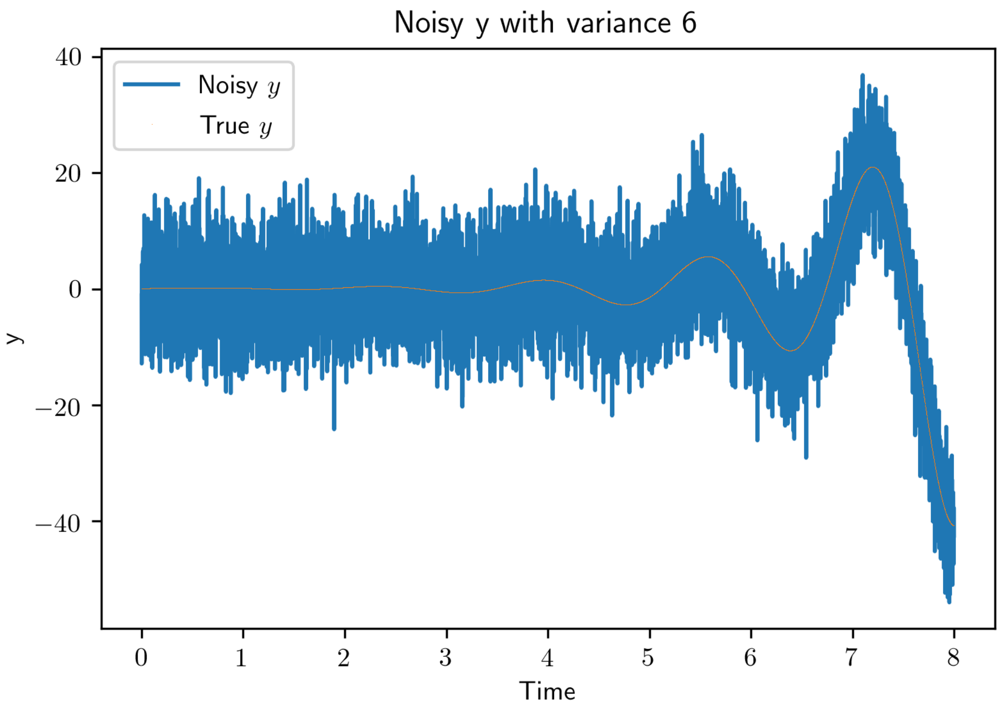


**Figure 5.21** True and noisy system output with AWGN of $\mu = 0$ and $\sigma = 6$ and N=3000

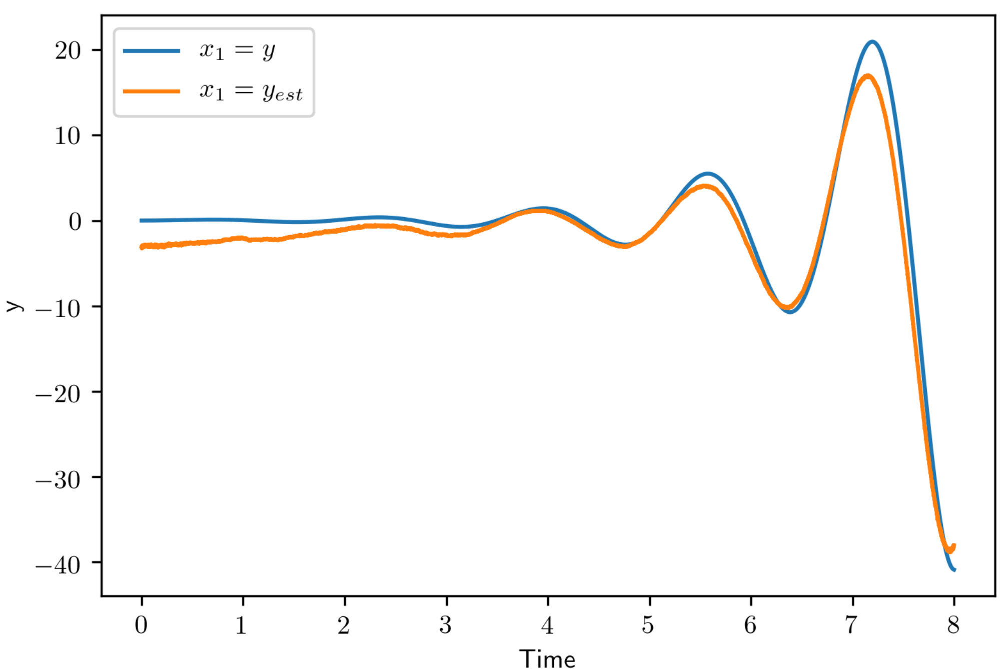


**Figure 5.22** True and reconstructed output trajectories of the system with AWGN of $\mu = 0$ and $\sigma = 6$ and N=3000

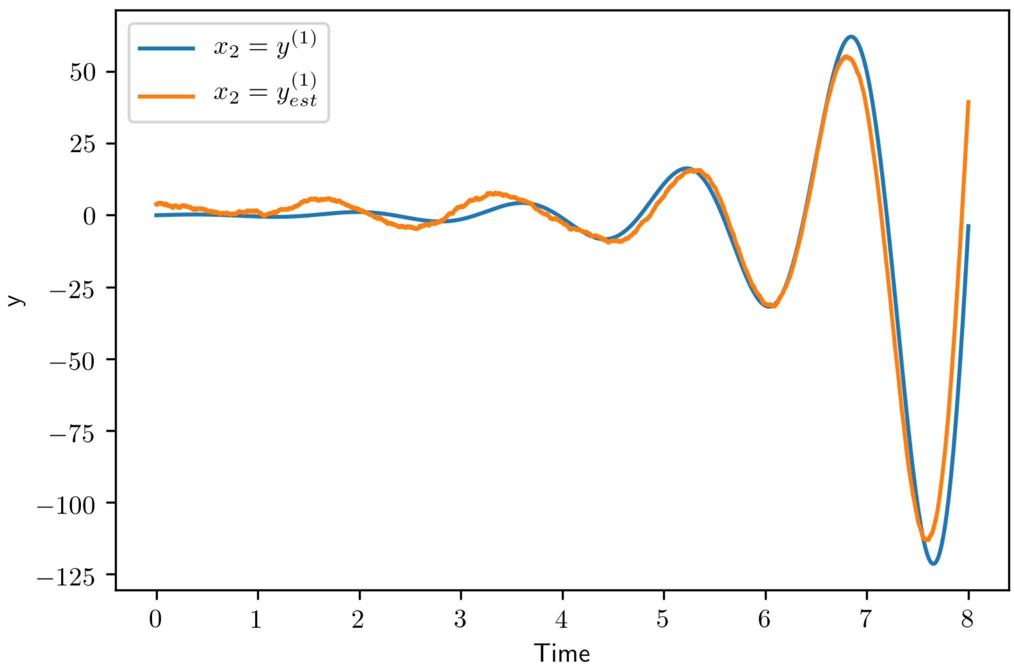


**Figure 5.23** True and reconstructed first derivative of the system with AWGN of $\mu = 0$ and $\sigma = 6$ and N=3000

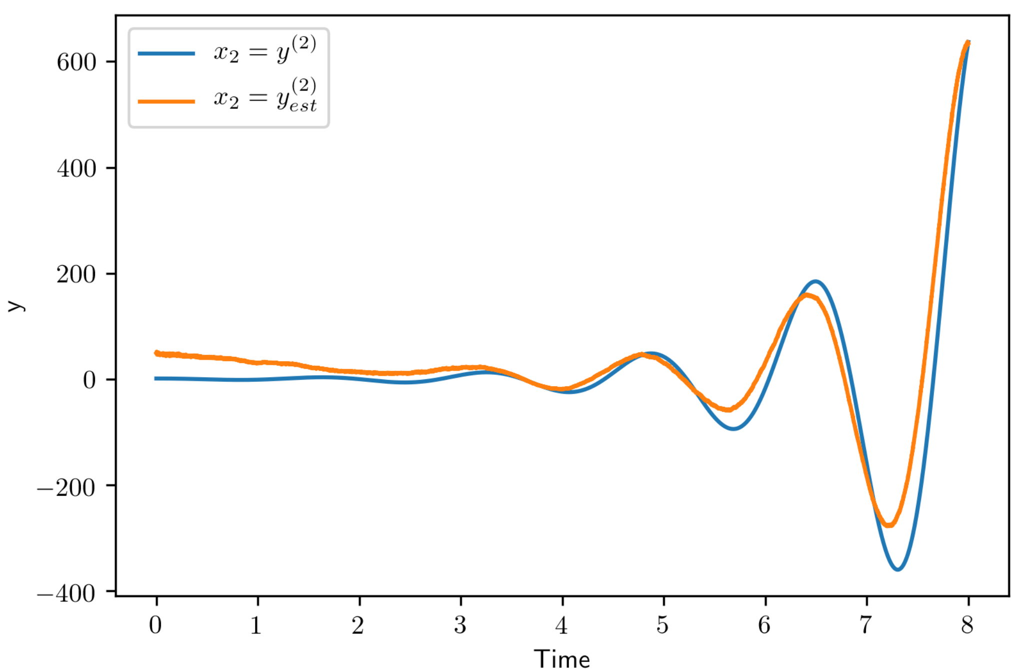


**Figure 5.24** True and reconstructed second derivative of the system with AWGN of $\mu = 0$ and $\sigma = 6$ and N=3000

**Example 2:**

Let us consider another third order LTI system;

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 5 & -20 & 1 \end{bmatrix} x \;; y = x_1 \;;\; x(0) = [1,1,0] \tag{5.6}$$

with its corresponding characteristic equation

$$y^{(3)}(t) + a_2 y^{(2)}(t) + a_1 y^{(1)} + a_0 y(t) = 0 \tag{5.7}$$

The parameters $a_0$, $a_1$, $a_2$ are assumed to be unknown.
Now we apply Gaussian white noise of zero mean and variance of $\sigma = 1.5$ to the solution of the above system. The samples for regression are taken in batches of N = 3000 knots taken randomly from a uniform distribution over [a,b] = [0,6].
Employing the proposed method as described in Chapter 4, the estimated system parameters using third-order kernels were found as they are presented in Table (5.4) with the number of sample points $N = 3000$ which were selected randomly from the uniform distribution in $[a, b]$.

| | $a_0$ | $a_1$ | $a_2$ |
|---|---|---|---|
| True values | -5 | 20 | -1 |
| Estimated Values | -5.2611 | 20.8678 | -0.9787 |

**Table 5.4** True and estimated parameter values from a true output with AWGN $\mu = 0$ and $\sigma = 1.5$, N=3000 using third order kernels

On the pages that follow, figures are provided for $y_m$ vs. $y_T$, the reconstruction of the system output, and its derivatives for the estimated parameter values in Table (5.4)

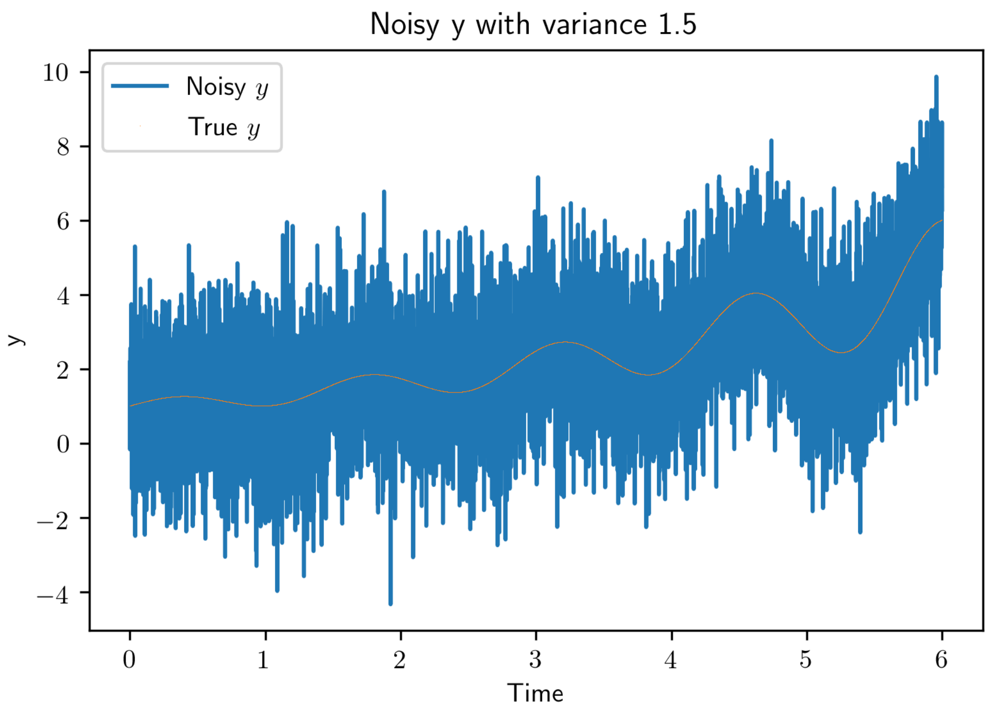


**Figure 5.25** True and noisy system output with AWGN of $\mu = 0$ and $\sigma = 1.5$ and N=3000

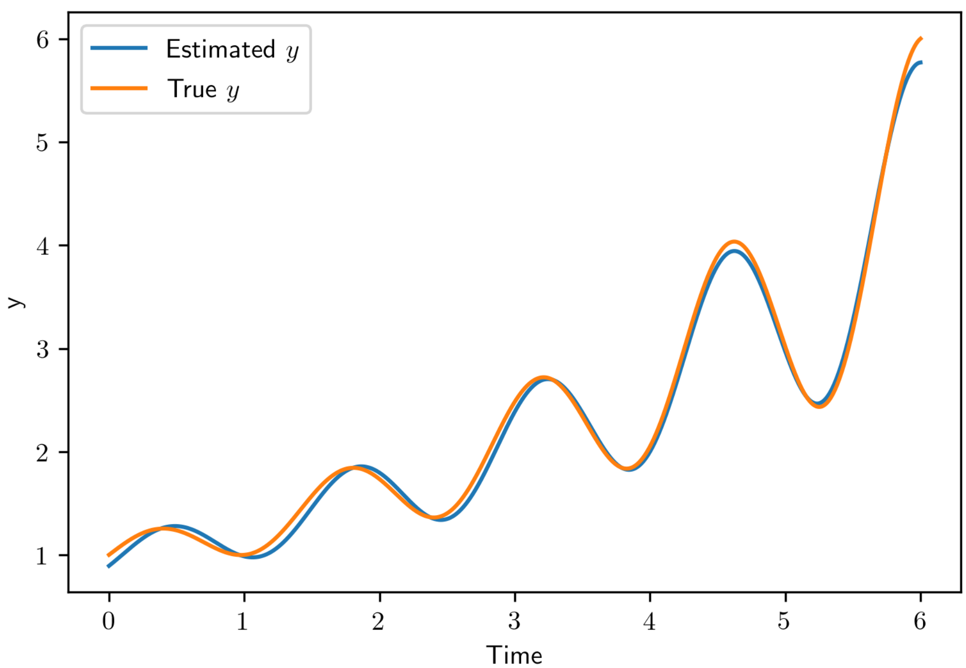


**Figure 5.26** True and reconstructed output trajectories of the system with AWGN of $\mu = 0$ and $\sigma = 1.5$ and N=3000

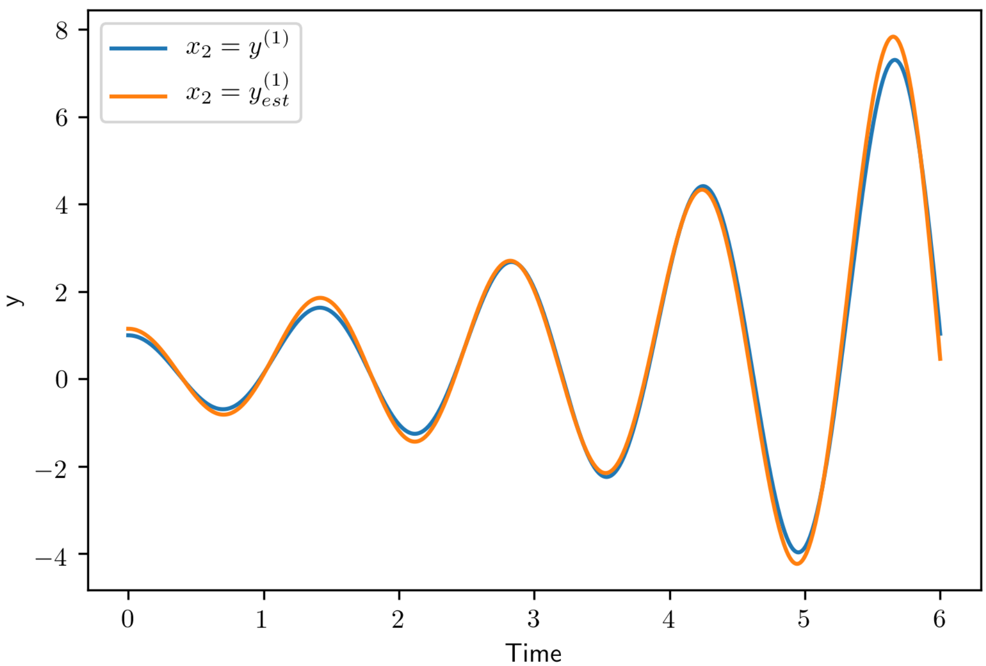


**Figure 5.27** True and reconstructed first derivative of the system with AWGN of $\mu = 0$ and $\sigma = 1.5$ and N=3000

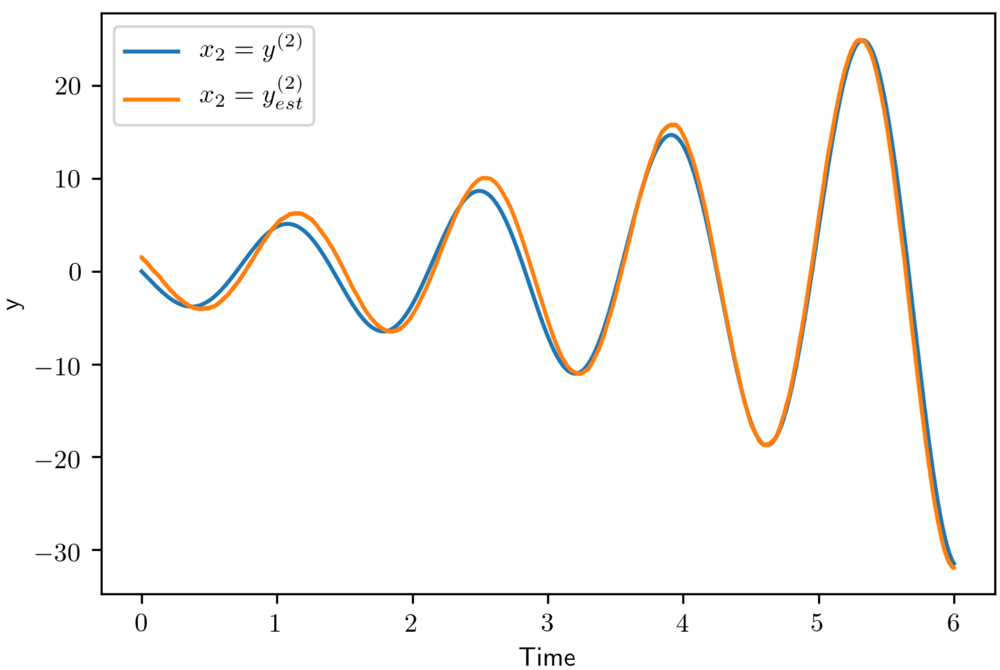


**Figure 5.28** True and reconstructed second derivative of the system with AWGN of $\mu = 0$ and $\sigma = 1.5$ and N=3000

## 5.2 Comparison with previous works:

Table (5.5) shows the comparison of the proposed method with Ghoshal et al. [24] and John et al. [27] results. The number samples taken is 1000 from the uniform distribution in [0,8]. The example 1 in Section 5.1 is considered here.

| Noise | *Method* | Estimted $a_0$ | Estimted $a_1$ | Estimted $a_2$ | RMSD |
|---|---|---|---|---|---|
| $\sigma = 1$ | Proposed | 26.0035 | 12.9961 | 0.0006 | 0.0002 |
| | Ghoshal et al. [24] | 25.6977 | 13.2109 | 0.0020 | 0.0006 |
| | John et al. [27] | 34.2321 | 6.3355 | -0.7380 | 0.0419 |
| $\sigma = 1.5$ | Proposed | 26.0167 | 12.9624 | -0.0002 | 0.0002 |
| | Ghoshal et al. [24] | 23.4432 | 13.6524 | 0.1397 | 0.0095 |
| | John et al. [27] | 7.3532 | 4.1899 | -1.0202 | 0.1432 |
| $\sigma = 2$ | Proposed | 26.0344 | 12.5089 | -0.0023 | 0.0007 |
| | Ghoshal et al. [24] | 16.3091 | 8.5396 | 1.4127 | 0.0152 |
| | John et al. [27] | 1.0298 | -8.0940 | 5.1837 | 0.3024 |

**Table 5.5** Comparitive Study with Ghoshal et al. and Anju et al.

On the pages that follow, figures are provided for $y_m$ vs. $y_T$ (estimated/ reconstructed vs. nominal trajectories), for the parameter values estimated by the methods specified in Table (5.5)

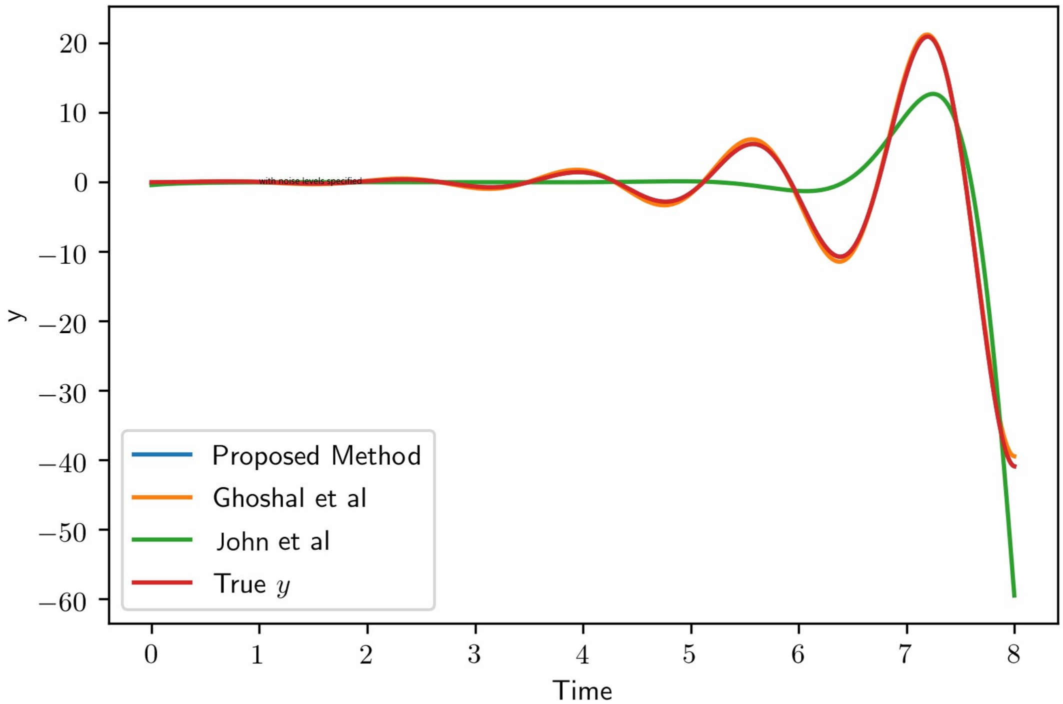


**Figure 5.29** Comparison of different methods with AWGN of $\mu = 0$ and $\sigma = 1.0$ and N=1000

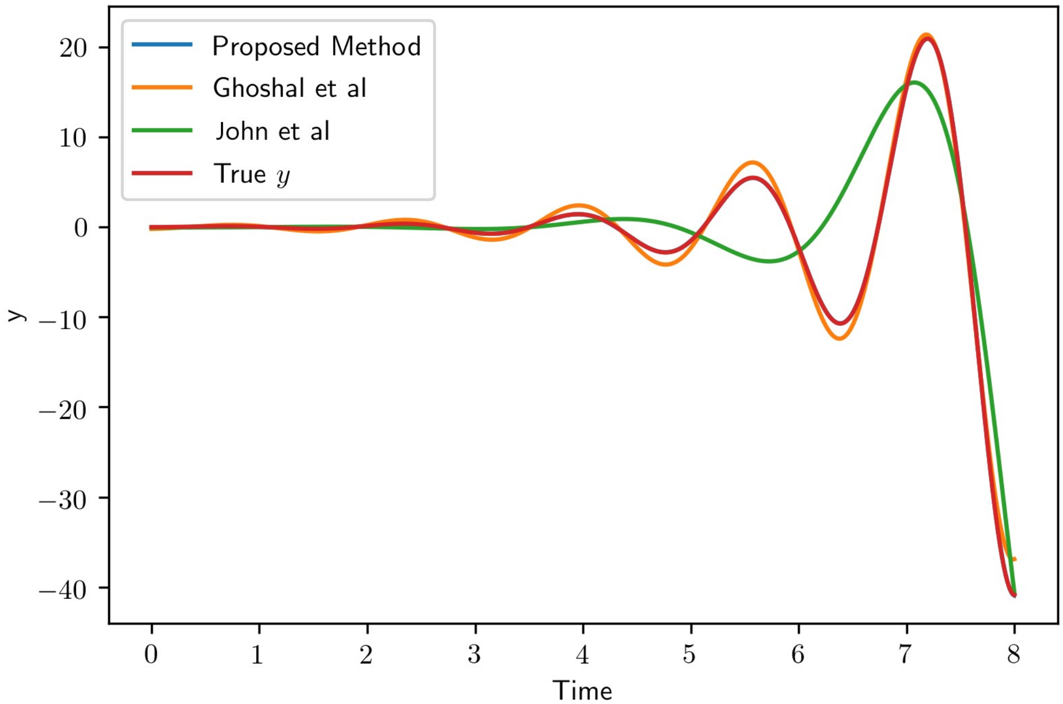


**Figure 5.30** Comparison of different methods with AWGN of $\mu = 0$ and $\sigma = 1.5$ and N=1000

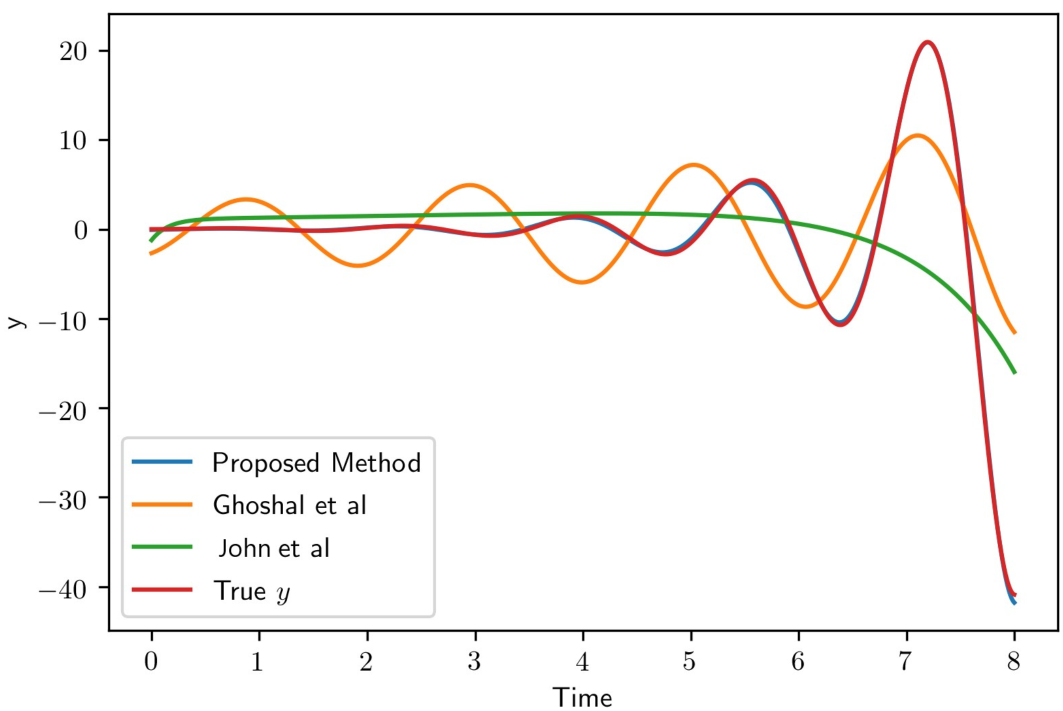


**Figure 5.31** Comparison of different methods with AWGN of $\mu = 0$ and $\sigma = 2.0$ and N=1000

## 5.3 Parameter estimation using kernels with different dimensions:

In the previous sections, we have estimated the parameters of a third-order system using third-order kernels. In this section, we experiment with kernels of different dimensions to estimate the parameters of a third-order system. The example 1 in Section 5.1 is considered here.

Table (5.6) shows the estimated parameter values obtained by using third, fourth and fifth-order kernels. The number samples taken is 3000 from the uniform distribution in $[0, 8]$

| Noise | Order | $a_0$ | $a_1$ | $a_2$ | $a_3$ | $a_4$ |
|---|---|---|---|---|---|---|
| $\sigma = 0.5$ | 3 | 26.0021 | 12.9989 | 0.0002 | - | - |
| | 4 | 15.3306 | 9.6999 | -0.9426 | 11.1827 | - |
| | 5 | 21.7556 | 17.2065 | -3.2358 | -1.7101 | -1.1800 |
| $\sigma = 1.0$ | 3 | 26.0029 | 13.0049 | -0.0001 | - | - |
| | 4 | 4.4694 | 15.7712 | -2.2912 | 6.6738 | - |
| | 5 | 20.8494 | 17.0661 | -1.7050 | -1.6151 | 1.5668 |
| $\sigma = 1.5$ | 3 | 26.0091 | 12.9822 | -0.0010 | - | - |
| | 4 | 4.3390 | 15.9094 | -1.7993 | 8.5469 | - |
| | 5 | -2.3282 | 16.7998 | -1.7097 | 2.6028 | 19.9333 |

**Table 5.6** Estimated parameter values using different kernels

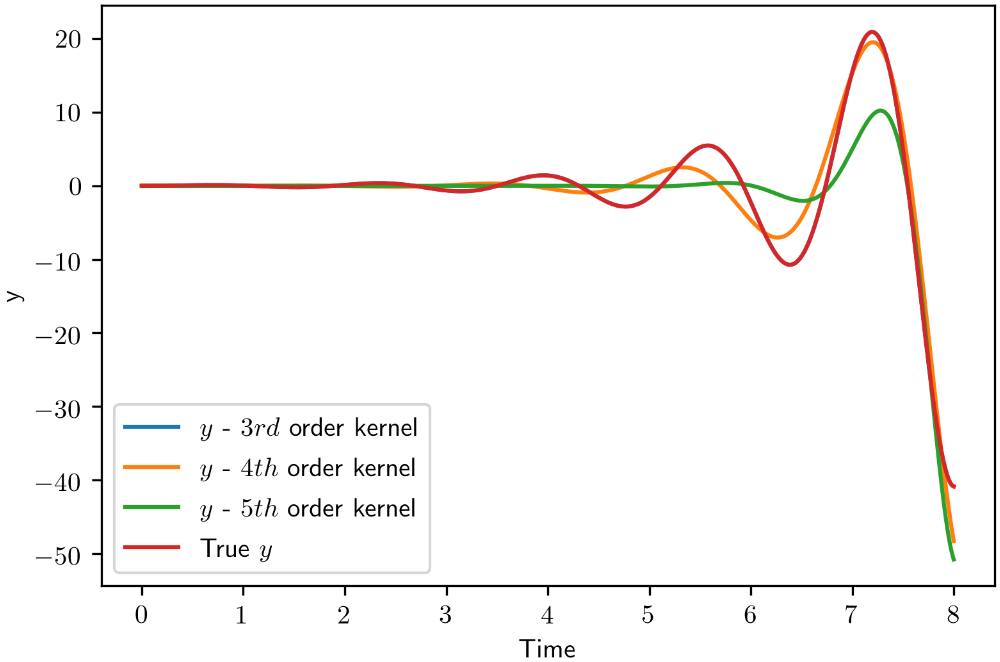


**Figure 5.32** Estimate of $y$ using different kernels vs nominal $y$ for noise of 0.5 SD

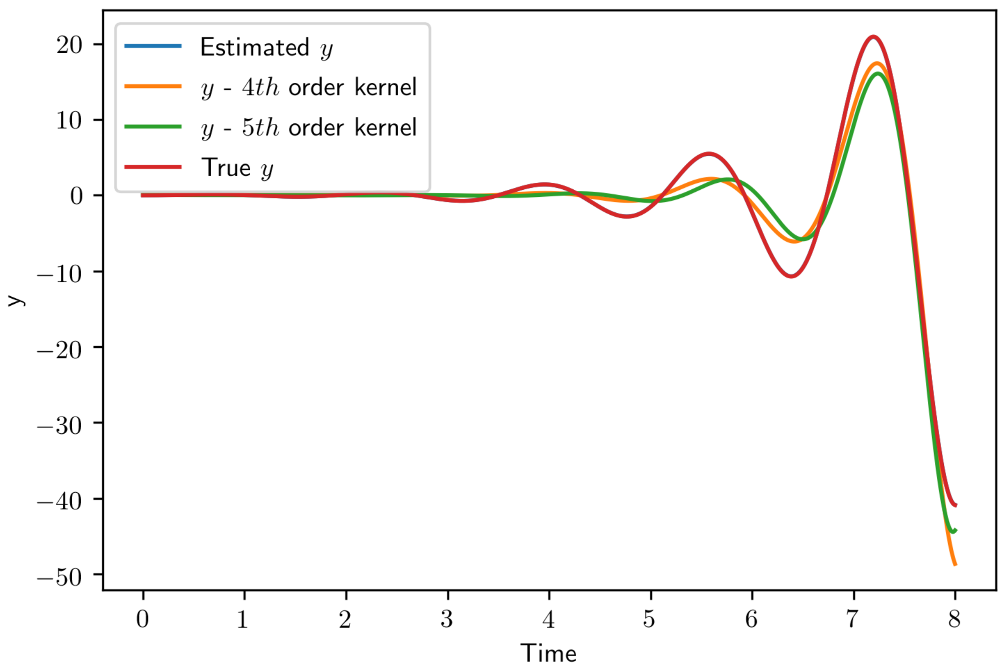


**Figure 5.33** Estimate of $y$ using different kernels vs nominal $y$ for noise of 1 SD

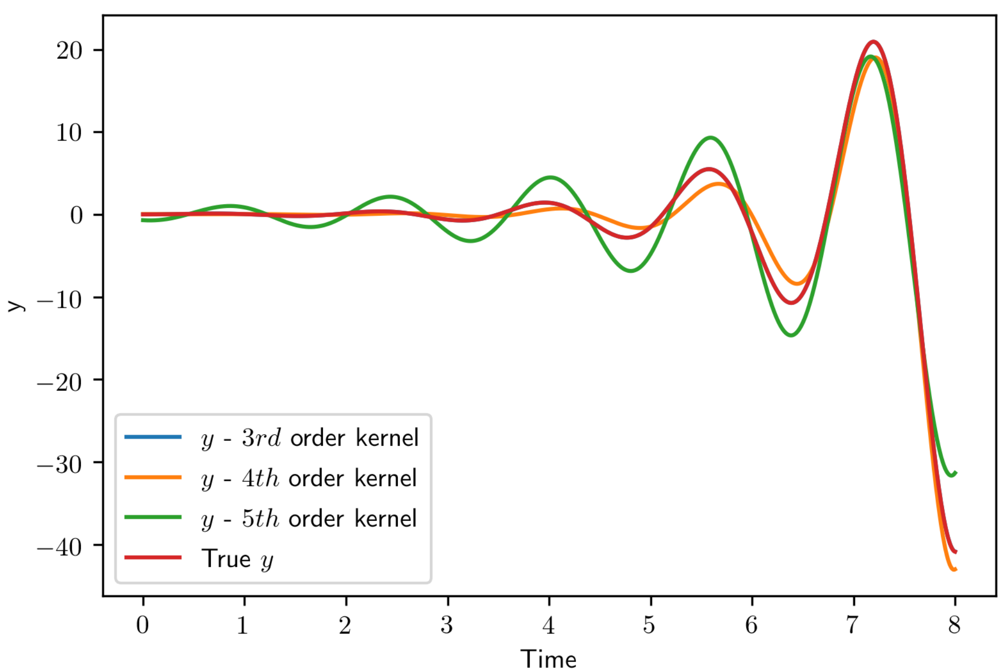


**Figure 5.34** Estimate of $y$ using different kernels vs nominal $y$ for noise of 1.5 SD

## 5.4 Discussion of the results

It is shown that the reconstructed output and its derivative trajectories are identical to their true counterparts, provided that there is no output measurement noise. The accuracy of estimation is very stable for higher noise levels because of the use of statistically independent regression equations. This can be viewed from the comparative study of the proposed method with Ghoshal et al. [14]. We have also seen the method proposed in [27] needs a huge number of samples and fails to work with high noise. Parameter estimation using the kernels with different dimensions is also experimented with. With proper model order selection techniques, an accurate estimation can be made possible for the system of any order.

# Chapter 6

# Conclusion

State and Parameter Estimation is of utmost importance to applications in almost all disciplines of science. The work done in this field is vast and diverse ranging from various classical methods [47], [48], [49] of parameter estimation, methods involving data assimilation [50], recursive stochastic filtering [7], [51] and many more. Recursive stochastic algorithms stand out due to their simplicity and efficient noise attenuating capability. The parameter estimation techniques by those methods would assume some information about the initial state and measurement noise characteristics. In Chapter 1, we had discussed the algebraic methods, which partially overcomes the shortcoming of the classical methods. The algebraic methods include algebraic dead-beat observers based on the concept of differential flatness, algebraic parameter estimation and invariant observers [5] [10] [11] [12]. Chapter 2 discusses the double-sided kernel approach. It solves many drawbacks of algebraic methods such as singularity at t=0 and reinitialization of the estimator at regular intervals. We have derived a kernel approach for a third-order system. The kernel representation for the derivatives of the output of the third-order system is also derived. Chapter 3 discusses the previous works done on state and parameter estimation using the double-sided kernel approach. Chapter 4 builds on the Parameter estimation using the stochastic regression discussed in Chapter 3. A recursive generalized least squares with inverse covariance weighting is used to deal with heteroskedasticity. With the usage of statistically independent regression equations, the estimation accuracy can be maintained for high noise in a system. The simulations and comparative study are provided in Chapter 5.

## Future work

The following are few possible directions for future developments:

- This proposed method of Recursive least squares can be used in a Kalman filter. The high noise attenuating kernel approach methods finds a very special role here. Lets us consider the equation (4.116)

$$\hat{a}_{k+1} = \hat{a}_k + M_{k+1}^{-1} P_{k+1}^T S_{k+1} (Q_{k+1} - P_{k+1} \hat{a}_k)$$

  Here the term $M_{k+1}^{-1} P_{k+1}^T S_{k+1}$ serves as a Kalman gain and the term $(Q_{k+1} - P_{k+1} \hat{a}_k)$ is called the innovations, as it compares the difference between a data update and the prediction, given the last estimate.

- We have used a linear system for our study. This method can be extended to investigate nonlinear systems.

- Another possible direction is to investigate the color noise.

## Programming summary:

The software used in this thesis is written in Python. We used Jupyter Notebook on the Google Colab platform during the initial development. With the increase in complexity and accuracy, we started using the cloud services of Compute Canada. Graham and Beluga are the two clusters that are accessed to run the program. These general-purpose clusters are composed of large memory nodes where we can specify the number of cores and memory needed for the program. To access the remote computing in a secured way, Mobexterm, an SSH client is used. It is the most common way to store the files, run the jobs, and transferring files. Detailed descriptions can be found here [52]. With the availability of more number of cores and memory, the program can be made to run faster with multiprocessing. We can submit multiple processes to separate memory locations. Every process will run independently from each other with the help of distributed memory. More details on multiprocessing can be found in [53].

# Bibliography

[1] F. Ding. System identification: Performances analysis for identification methods. *Science Press, Beijing, China*, 2014.

[2] Y. B. Hu. Iterative and recursive least squares estimation algorithms for moving average systems. *Simulation Modelling Practice and Theory*, 34:12 – 19, 2013.

[3] W. Fan X. L. Xiong and R. Ding. Least-squares parameter estimation algorithm for a class of input nonlinear systems. *Journal of Applied Mathematics*, 2012(684074), 2012.

[4] Y. Shi and H. Fang. Kalman filter-based identification for systems with randomly missing measurements in a network environment. In *International Journal of Control 2010*, 2010.

[5] Hebertt Sira-Ramírez, Carlos García Rodríguez, John Cortés Romero, and Alberto Luviano Juárez. *Algebraic identification and estimation methods in feedback control systems*. John Wiley & Sons, 2014.

[6] Rudolph E Kalman and Richard S Bucy. New results in linear filtering and prediction theory. *Journal of basic engineering*, 83(1):95–108, 1961.

[7] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

[8] D Luenberger. Observers for multivariable systems. *IEEE Transactions on Automatic Control*, 11(2):190–197, 1966.

[9] David G Luenberger. Observing the state of a linear system. *IEEE transactions on military electronics*, 8(2):74–80, 1964.

[10] Michel Fliess, Cédric Join, and Hebertt Sira-Ramírez. Non-linear estimation is easy. *International Journal of Modelling, Identification and Control*, 4(1):12–27, 2008.

[11] Hebertt Sira-Ramírez and Michel Fliess. An algebraic state estimation approach for the recovery of chaotically encrypted messages. *International Journal of Bifurcation and Chaos*, 16(02):295–309, 2006.

[12] Mamadou Mboup, Cédric Join, and Michel Fliess. A revised look at numerical differentiation with an application to nonlinear feedback control. In *The 15th Mediterrean Conference on Control and Automation-MED'2007*, 2007.

[13] Michel Fliess. Analyse non standard du bruit. *Comptes Rendus Mathematique*, 342(10):797–802, 2006.

[14] DP Ghoshal and Hannah Michalska. Finite interval kernel-based identification and state estimation for lti systems with noisy output data. In *American Control Conference, 2019. ACC 2019*, 2019 forthcoming.

[15] Kumar Vishwanath Gopalakrishnan. A new method of non-asymptotic estimation for linear systems. Master's thesis, McGill University, 2016.

[16] Hannah Michalska. Instructor Notes: Reproducing Kernels for Modelling of Systems, 2016.

[17] Lennart Ljung. *System identification: theory for the user*. Prentice-hall, 1987.

[18] DP Ghoshal, Kumar Gopalakrishnan, and Hannah Michalska. Using invariance to extract signal from noise. In *American Control Conference, 2017. ACC 2017*, 2017.

[19] DP Ghoshal, Kumar Gopalakrishnan, and Hannah Michalska. Algebraic parameter estimation using kernel representation of linear systems - submitted. In *IFAC World Congress, 2017. 20th World Congress*, 2017.

[20] DP Ghoshal, Kumar Gopalakrishnan, and Hannah Michalska. Parameter and state estimation in linear systems under arbitrary noise - submitted. In *IFAC World Congress, 2017. 20th World Congress*, 2017 forthcoming.

[21] Debarshi Patanjali Ghoshal, Kumar Gopalakrishnan, and Hannah Michalska. Kernel-based adaptive multiple model target tracking. In *IEEE Conference on Control Technology and Applications*, 2017.

[22] Badi Baltagi. *Econometric analysis of panel data*. John Wiley & Sons, 2008.

[23] William H Greene. *Econometric analysis*. Pearson Education India, 2003.

[24] DP Ghoshal and Hannah Michalska. Forward-backward kernel-based state and parameter estimation for linear systems of arbitrary order. In *International Journal of Control*, (submitted).

[25] Aritra Chatterjee. *Two Stage Kernel Based Optimization of LTI Systems*. Thesis, McGill University, Montreal, August 2019.

[26] Hannah Michalska. Instructor Notes: Parameter Estimation using Lsplines in RKHS, 2019.

[27] Anju John. *Estimation for SISO LTI Systems Using Dierential Invariance*. Thesis, McGill University, Montreal, August 2019.

[28] Hannah Michalska. Instructor Notes: Kernel Based Dynamical System Identication, 2019.

[29] DP Ghoshal and Hannah Michalska. Finite interval estimation of lti systems using differential invariance,instrumental variables, and covariance weighting. In *American Control Conference, 2020. ACC 2020*, 2020 forthcoming.

[30] Stefan Schaffler. *Generalized stochastic processes: modelling and applications of noise processes*. Springer Berlin Heidelberg, New York, NY, 2018.

[31] Humberto Barreto and Frank Howland. *Introductory Econometrics*, (accessed August 23, 2020). http://www3.wabash.edu/econometrics/econometricsbook/chap19.htm.

[32] Alexander C Aitken. Iv.—on least squares and linear combination of observations. *Proceedings of the Royal Society of Edinburgh*, 55:42–48, 1936.

[33] Peter C Young. *Recursive estimation and time-series analysis: An introduction for the student and practitioner*. Springer Science & Business Media, 2011.

[34] DQ Mayne. A method for estimating discrete time transfer functions. In *Advances in Computer Control, Second UKAC Control Convention*. The University of Bristol, 1967.

[35] Kwan Wong and Elijah Polak. Identification of linear discrete time systems using the instrumental variable method. *IEEE Transactions on Automatic Control*, 12(6):707–718, 1967.

[36] Peter C Young. The use of linear regression and related procedures for the identification of dynamic processes. In *Seventh Symposium on Adaptive Processes*, pages 53–53. IEEE, 1968.

[37] P Caines. On the asymptotic normality of instrumental variable and least squares estimators. *IEEE Transactions on Automatic Control*, 21(4):598–600, 1976.

[38] B Friedlander. The overdetermined recursive instrumental variable method. *IEEE Transactions on Automatic Control*, 29(4):353–356, 1984.

[39] Peter E Caines. *Linear stochastic systems*, volume 77. SIAM, 2018.

[40] S Stanhope, JE Rubin, and David Swigon. Identifiability of linear and linear-in-parameters dynamical systems from a single trajectory. *SIAM Journal on Applied Dynamical Systems*, 13(4):1792–1815, 2014.

[41] Wiener process. https://www.eng.tau.ac.il/$\sim$ *liptser/lectures/lect_new10.pdf*, 2019.

[42] Mohammed Dahleh, Munther Dahleh, and George Verghese. *Lectures on Dynamic Systems and Control*. PhD thesis, Massachuasetts Institute of Technology, 2011.

[43] Model structure selection: Determining model order and input delay - matlab simulink process. https://www.mathworks.com/help/ident/ug/model-structure-selection-determining-model-order-and-input-delay.htmld122e16072, 2020.

[44] Hirotugu Akaike. Fitting autoregressive models for prediction. *Annals of the institute of Statistical Mathematics*, 21(1):243–247, 1969.

[45] Hirofumi Wakaki Shinpei Imori, Hirokazu Yanagihara. Simple formula for calculating bias-corrected aic in generalized linear models. *Scandinavian Journal of Statistics*, 41(2):535–555, 2014.

[46] Anderson David R Burnham, Kenneth P. *Model Selection and Multimodel Inference.* Springer Science & Business Media, 2002.

[47] Lennart Ljung. System identification: Theory for the user, ptr prentice hall information and system sciences series. *ed: Prentice Hall, New Jersey*, 1999.

[48] Karl Johan Åström and Peter Eykhoff. System identification—a survey. *Automatica*, 7(2):123–162, 1971.

[49] GC Goodwin and KS Sin. Adaptive filtering prediction and control. 1984. *Englewood Clifs: Prentice Ha lI.*

[50] Eugenia Kalnay. *Atmospheric modeling, data assimilation and predictability*. Cambridge university press, 2003.

[51] Thomas Kailath, Ali H Sayed, and Babak Hassibi. *Linear estimation*, volume 1. Prentice Hall Upper Saddle River, NJ, 2000.

[52] Jupyter - cc doc. https://docs.computecanada.ca/wiki/Jupyter, 2020.

[53] Multiprocessing. https://docs.python.org/2/library/multiprocessing.html, 2020.