# Estimation of Infinite-Dimensional Systems

Amir Issaei

Spring 2014

**Abstract**

The purpose of this research paper is to discuss the observation systems in an infinite-dimensional space. Since finite-dimensional approximations are used to design an estimator for infinite-dimensional systems, it is necessary to discuss the conditions under which the finite-dimensional estimator converges to the one that estimates the infinite-dimensional system. On the other hand, the measurements of the states of an infinite-dimensional system are often taken discretely in time. The convergence of discrete-time observers is presented for an infinite-dimensional system. The results are applied to a one-dimensional parabolic partial differential equation.

# Acknowledgement

I would like to thank my supervisor, Prof. Kirsten Morris, who led me to become more independent, and through which I found my passion. I must express my gratitude to Isabel, my fiancee, for her continued support and encouragement. I was amazed by the patience of my father, mother, brother and sister who experienced all of the ups and downs of my post-graduate studies.

# Contents

# List of Figures

# 1 Introduction

Some of the most important practical problems in control theory have statistical and probabilistic nature. Such problems are: 1) prediction of random function (signals); 2) separation of noise and signal; 3) detection of signals of known form in the presence of random noise. In order to find solutions to such problems, the knowledge of statistics, mathematics and probability is needed.

Consider a dynamical system, where the measurements and the states of the system are contaminated with noise. Different methods have been proposed to estimate the states of such a system (see [1], [3], [4] and [15]). Kalman approached the state estimation from the point of view of conditional distributions and expectations. He treated linear systems as a system of coupled first-order difference (or differential) equations. Since Kalman considered the state transition formulation for a dynamical system, a single derivation covered a large variety of problems including non-stationary statistics, non-linear problems and non-Gaussian processes. The new formulation of the problem brought it into contact with the growing new theory of control systems based on the state point of view. The implementation of Kalman filter is easy in practice. Kalman filter is a recursive algorithm, which does not require a large amount of memory on digital machines [14].

The structure of this research paper is as follows: first, an optimal control problem with linear quadratic cost is discussed. Then the results from that problem are extended for designing Kalman filters for infinite-dimensional systems. The conditions under which the sampled-data finite-dimensional estimator converges is discussed and the results are applied to a one-dimensional heat equation.

# 2 Optimal control problem with linear quadratic cost

The linear quadratic control criterion for placement of actuators is a popular choice among researchers. In many applications, such as vibration control and diffusion modeling, the mathematical formulation of the control system is given by a partial differential equation. The state space for such systems is infinite-dimensional. In this section the control system is formulated, then the existence and uniqueness of the optimal input function, $u_{optimal}(.)$, on an infinite-dimensional space and infinite-time interval, is discussed. The material in this section can be found in [2].

**Definition 1.** *A strongly continuous* $(C_0)$*-semigroup S(t) on Hilbert space Z is a family* $S(t) \in \mathcal{L}(Z;Z)$*, where* $\mathcal{L}(Z;Z)$ *indicates bounded linear operators from* $Z$ *to* $Z$*, such that*
*(i)* $S(0) = I$*,*
*(ii)* $S(t+s) = S(t)S(s)$*,* $\forall t, s \geq 0$*,*
*(ii)* $\lim_{t \to 0^+} S(t)z = z$*, for all* $z \in Z$*.*

**Definition 2.** *The infinitesimal generator A of a* $C_0$*-semigroup on Z is defined by*

$$Az = \lim_{t \to 0^+} \frac{1}{t}(S(t)z - z). \tag{1}$$

*The domain of A,* $D(A)$*, is defined as the set of elements* $z \in Z$ *for which the limit exists.*

Consider a dynamical system described by

$$\dot{z}(t) = Az(t) + Bu(t), \;\; t \geq 0, \;\; z(0) = z_0, \tag{2}$$

where $A$, with domain $D(A)$, generates a $C_0$-semigroup $S(t)$ on a Hilbert space $Z$, $u(t) \in \mathbb{L}_2([0, \infty]; U)$ is the input function, $U$ is a finite-dimensional space (for instance, $\mathbb{R}^p$, where $p$ is a positive integer) and $B \in \mathcal{L}(U; Z)$ where $\mathcal{L}(U; Z)$ indicates the set of bounded linear operators from $U$ to $Z$. The trajectories of (2) can be written as

$$z(t) = S(t)z_0 + \int_0^t S(t-s)Bu(s)ds, \tag{3}$$

using semigroup theory [2]. In the following example, a partial differential equation (PDE) is formulated as an abstract differential equation (2). This example can be found in [2].

Consider a heat transfer problem

$$\begin{aligned} &\frac{\partial z}{\partial t} = \frac{\partial^2 z}{\partial x^2} + u(x,t), \qquad z(x,0) = z_0(x), \\ &\frac{\partial z}{\partial x}(0,t) = \frac{\partial z}{\partial x}(1,t) = 0, \end{aligned} \tag{4}$$

where $z(x,t)$ represents the temperature of a bar at time $t$ and position $x$, and $u(x,t)$ is the control function. Choose $Z = \mathbb{L}_2(0,1)$ as the state space and

$z(.,t) = \{z(x,t), 0 \leq x \leq 1\}$ as the state of the system. Define

$$
\begin{aligned}
&Ah = \frac{\mathrm{d}^2 h}{\mathrm{d}x^2} \\
&D(A) = \{h \in \mathbb{L}_2(0,1) | h, \frac{\mathrm{d}h}{\mathrm{d}x} \text{ are absolutely continious}, \\
&\qquad\qquad \frac{\mathrm{d}h^2}{\mathrm{d}x^2} \in \mathbb{L}_2(0,1) \text{ and } \frac{\mathrm{d}h}{\mathrm{d}x}(0) = \frac{\mathrm{d}h}{\mathrm{d}x}(1) = 0\}, \\
&B = I,
\end{aligned}
\tag{5}
$$

and regard the input trajectory, $u(.,t)$, as the input and the function $z_0(.) \in \mathbb{L}_2(0,1)$ as the initial state. The PDE (4) can be written as

$$
\dot{z}(t) = Az(t) + Bu(t), t \geq 0, \;\; z(0) = z_0. \tag{6}
$$

The trajectories of (4) can also be written as in (3). It is shown in [2, page 107] that the solution to (2) is

$$
z(x,t) = \int_0^1 g(t,x,y) z_0(y) dy + \int_0^t \int_0^1 g(t-s,x,y) u(y,s) dy ds, \tag{7}
$$

where $z_0$ satisfies the boundary conditions and $u(x,t) \in \mathbb{L}_2([0,\tau]; \mathbb{L}_2(0,1))$ is the input function. The integrand, $g(t,x,y)$, represents the Green's function

$$
g(t,x,y) = 1 + \sum_{n=1}^{\infty} 2e^{-n^2\pi^2 t} \cos(n\pi x) \cos(n\pi y). \tag{8}
$$

To interpret (7) abstractly on $\mathbb{L}_2(0,1)$, consider the following bounded operator on $\mathbb{L}_2(0,1)$

$$
z(t) = S(t) z_0, t \geq 0, \tag{9}
$$

where for each $t \geq 0$, $S(t) \in \mathcal{L}(\mathbb{L}_2(0,1))$ is defined by

$$S(t)z_0(x) = \int_0^1 g(t,x,y)z_0(y)dy. \tag{10}$$

Then, the abstract formulation of the solution (7) on $\mathbb{L}_2(0,1)$ is

$$z(t) = S(t)z_0 + \int_0^t S(t-s)u(s)ds. \tag{11}$$

The partial differential equation (4) was formulated as an abstract differential equation (2) on an infinite-dimensional state space $Z = \mathbb{L}_2(0,1)$, where A is the unbounded operator on Z defined by (5), B is the identity on Z, $z_0$ and $u(.,t)$ are functions on Z, and the solution is given by (11). The operator $S(t)$ plays the role of $e^{At}$ in finite dimensions.

**Theorem 2.1.** *[12, page 104] Every bounded, self-adjoint, non-negative operator T has a unique bounded, self-adjoint, non-negative square root G such that $G^2 = T$. Furthermore, G commutes with any linear operator that commutes with T.*

Consider the dynamical system (2) with output

$$y(t) = Q^{\frac{1}{2}}z(t) \tag{12}$$

where $Q \in \mathcal{L}(Z;Z)$ is a self-adjoint, non-negative operator, $Q^{\frac{1}{2}} \in \mathcal{L}(Z;Y)$, and $Y$ is a Hilbert space. In linear quadratic optimal control, a cost function $J(z_0;u)$ is associated with the trajectories in (3)

$$J(z_0;u) = \int_0^\infty \langle Q^{\frac{1}{2}}z(s), Q^{\frac{1}{2}}z(s)\rangle + \langle u(s), Ru(s)\rangle ds, \tag{13}$$

where $z(s)$ is given by (3), $u \in \mathbb{L}_2([0,\infty];U)$ and $R \in \mathcal{L}(U;U)$ is a self-adjoint, coercive operator, i.e $R \geq \epsilon I$ for some $\epsilon > 0$.

The control problem is, given the initial condition, $z_0$, to find an optimal control $u_{optimal} \in \mathbb{L}_2([0,\infty];U)$ that minimizes the cost function $J$ over all trajectories of the control system (2). Since given any initial condition, $z_0$, the trajectories of the system are completely determined by input, it follows that the minimization problem is to find $u$ that minimizes the cost function, $J$.

**Definition 3.** *The linear system* (2) *with the cost function* (13) *is* optimizable *if, for every $z_0 \in Z$, there exists an input function $u \in \mathbb{L}_2([0,\infty);U)$ such that the cost function* (13) *is finite.*

**Definition 4.** *A $C_0-$semigroup, $S(t)$, on $Z$ is* exponentially stable*, if there exists positive constants, $M$ and $\alpha$ such that*

$$||S(t)|| \leq Me^{-\alpha t}$$

*where $||.||$ is the operator norm on $\mathcal{L}(Z;Z)$.*

**Definition 5.** *The system $(A,B)$ is stabilizable if there exists $K \in \mathcal{L}(Z;U)$ such that $A - BK$ generates an exponentially stable semigroup.*

**Definition 6.** *The pair $(A, Q^{\frac{1}{2}})$ is detectable if there exists $F \in \mathcal{L}(Y;Z)$ such that $A - FQ^{\frac{1}{2}}$ generates an exponentially stable semigroup.*

**Theorem 2.2.** *[2, Thm 6.2.4, 6.2.7] If the system* (2) *is optimizable and $(A, Q^{\frac{1}{2}})$ is detectable, then the cost function* (13) *has a minimum for every $z_0 \in Z$. Furthermore,* *there exists a self-adjoint non-negative operator* $\Pi \in \mathcal{L}(Z)$ *such that*

$$\min_{u \in \mathbb{L}_2([0,\infty],U)} J(z_0; u) = \langle z_0, \Pi z_0 \rangle. \tag{14}$$

*The operator* $\Pi$ *is the unique non-negative solution to the Riccati operator equation*

$$\langle Az_1, \Pi z_2 \rangle + \langle \Pi z_1, Az_2 \rangle + \langle Q^{\frac{1}{2}} z_1, Q^{\frac{1}{2}} z_2 \rangle - \langle B^* \Pi z_1, R^{-1} B^* \Pi z_2 \rangle = 0, \quad z_1, z_2 \in D(A). \tag{15}$$

*Let* $K = R^{-1} B^* \Pi$. *The optimal control is* $u_{optimal}(t) = -Kz(t)$ *and* $A - BK$ *generates an exponentially stable semigroup.*

# 3 Kalman filter

Consider the infinite-dimensional dynamical system

$$\begin{aligned} \dot{z}(t) &= Az(t) + Bu(t) + w(t), \quad z(0) = z_0, \\ y(t) &= Cz(t) + v(t), \end{aligned} \tag{16}$$

where A, B and $u(.)$ are defined in (2). Since it is not possible to measure all the states of the system, the output space $Y$, is a finite- dimensional space (for instance, $\mathbb{R}^q$, where q is a positive integer). The operator $C$ is a bounded linear operator that maps $Z$ to $Y$, i.e. $C \in \mathcal{L}(Z, Y)$. It is assumed that $w(t)$ and $v(t)$ are uncorrelated white Gaussian noise with mean zero and covariance $M = M^* \geq 0$ and $V = V^* > 0$ respectively. . The goal is to estimate the states of the system (16) in an optimal sense.

**Definition 7.** *Let X be a continuous random variable. The expectation of X is given by*

$$\mathbb{E}[X] = \int_{-\infty}^{\infty} x f(x) dx, \tag{17}$$

*where f is the probability density function of X.*

**Definition 8.** *Consider the linear system* (16) *with state space Z, input space U, and output space* $\mathbb{R}^q$*. The observer*

$$\begin{aligned} \dot{\hat{z}}(t) &= A\hat{z}(t) + Bu(t) + L[y(t) - C\hat{z}(t)], \hat{z}(0) = \hat{z}_0 \\ \hat{y}(t) &= C\hat{z}(t), \end{aligned} \tag{18}$$

*that minimizes the estimation cost*

$$J_e = \lim_{t\to\infty} \mathbb{E}\big[||z(t) - \hat{z}(t)||^2\big], \tag{19}$$

*is called "minimum-variance estimator" or "Kalman filter" and* $L \in \mathcal{L}(\mathbb{R}^q; Z)$ *is called "observer gain" or "Kalman gain".*

**Problem 1. Minimum-Variance estimator design (Kalman filter design)**

*Given the system* (16)*, find an* $L$ *such that the observer* (18) *minimizes* (19)*.*

It is shown in [5] and [9] that the Riccati equation associated with estimation problem above is

$$A\hat{\Pi} + \hat{\Pi}A^* - \hat{\Pi}C^*V^{-1}C\hat{\Pi} + M = 0, \tag{20}$$

and the observer gain is

$$L = \hat{\Pi}C^*V^{-1}. \tag{21}$$

The similarity between the linear stochastic filtering and the deterministic linear regulator problems was mentioned by Kalman and Bucy in [7] and [8]. The duality is usually described as a correspondence between the Kalman filter gain, the regulator feedback gain and the associated Riccati equations. The differences between the results of the two problems are characterized by a reversal of time and transposition (adjoint) of system matrices (operators) as explained in [13]. The duality between the two problems can be summarized by noting that the gain and the Riccati equation of the filter for the system

$$\begin{aligned} \dot{z}(t) &= A^* z(t) + B^* u(t) + w(t), \\ y(t) &= C^* z(t) + v(t), \end{aligned} \tag{22}$$

are the same as those for the linear regulator for the system

$$\dot{z}(t) = Az(t) + Cu(t), \quad z(0) = z_0, \tag{23}$$

$$J_r = \int_0^\infty \Big( z^*(t) M z(t) + u^*(t) V u(t) \Big) dt. \tag{24}$$

Note that A, B and C are matrices with appropriate dimensions, $*$ denotes the conjugate transposition, $u(t)$ is an input, $w(t)$ and $v(t)$ are uncorrelated white Gaussian noise with mean zero and covariance

$$\begin{aligned} M &= M^* \geq 0, \\ V &= V^* > 0, \end{aligned} \tag{25}$$

respectively [13]. Substitutions of $A$, $B$, $R$ and $Q^{\frac{1}{2}}$ in (15) by $A^*$, $C^*$, $V$ and $M^{\frac{1}{2}}$ result in (20).

Due to duality, the theorems regarding the existence and uniqueness of the optimal estimator in an infinite-dimensional space follow from the similar results for the optimal control problem. To design an estimator numerically, the infinite-dimensional space is usually projected into a finite-dimensional space. The convergence criteria of finite-dimensional approximation of controllers are summarized in [11]. The results can be extended to derive conditions under which the finite-dimensional approximations of the estimator also converge.

# 4 Approximation theory for infinite-dimensional estimator

For most practical examples, a closed form solution of the partial differential equation or of the transfer function is not available and an approximation must be used. The approximation is generally calculated using one of the many standard approaches, such as the finite difference, the finite element and Fourier methods, which are developed for solving partial differential equations. The resulting system of ordinary differential equations is used in the controller and observer design. The advantage of this approach is that the wide body of synthesis methods available for finite-dimensional systems can be used. Three main concerns arise. First, the controlled (observer) system may not perform as predicted because a finite-dimensional design is applied to an infinite-dimensional system. Second, the sequence of controller gains (observer gains) may not converge. Third, the original infinite-dimensional system may not be stabilizable even if the approximated finite-dimensional systems are stabilizable (the finite-dimensional approximation of the estimator may not converge to the infinite-dimensional estimator). The conditions under which the sequence of controller gains (observer gains) converge should be discussed. It is also necessary to investigate the conditions under which the finite-dimensional approximation stabilizes the original infinite-dimensional system and the finite-dimensional estimator converges to the one that estimates the infinite-dimensional system.

Suppose the approximation lies in some finite-dimensional sub-space $Z_n$ of $Z$ with an orthogonal projection $P_n : Z \to Z_n$, where for each $z \in Z$, $\lim_{n\to\infty} ||P_n z - z|| = 0$. It is assumed that the subspace, $Z_n$, is equipped with the norm of original space $Z$. Define $B_n = P_n B$, $C_n = C|_{Z_n}$ (the restriction of $C_n$ to $Z_n$), and define $A_n \in \mathcal{L}(Z_n, Z_n)$, $Q_n \in \mathcal{L}(Z_n, Z_n)$ and $M_n \in \mathcal{L}(Z_n, Z_n)$ using finite difference, finite element or Fourier methods. A sequence of finite-dimensional approximations for the linear quadratic optimal control problem are

$$\dot{z}_n(t) = A_n z(t) + B_n u(t), \;\; z(0) = P_n z_0, \;\; n \geq 1. \tag{26}$$

The linear quadratic cost associated with (26) is

$$J(z_0; u) = \int_0^\infty \langle Q_n^{\frac{1}{2}} z(s), Q_n^{\frac{1}{2}} z(s)\rangle + \langle u(s), Ru(s)\rangle ds, \tag{27}$$

and the Riccati equation associated with (26) and (27) is

$$A_n^* \Pi_n + \Pi_n A_n - \Pi_n B_n R^{-1} B_n^* \Pi_n + Q_n = 0. \tag{28}$$

A sequence of finite-dimensional approximations can also be written for the observer system (18)

$$\begin{aligned} \dot{\hat{z}}_n(t) &= A_n\hat{z}_n(t) + B_n u(t) + L_n[y(t) - C_n\hat{z}_n(t)], \quad \hat{z}(0) = P_n\hat{z}_0, \quad n \geq 1. \\ \hat{y}(t) &= C_n\hat{z}_n(t), \end{aligned} \tag{29}$$

where $L_n = \hat{\Pi}_n C_n^* V^{-1}$ and $\hat{\Pi}_n$ satisfies

$$A_n\hat{\Pi}_n + \hat{\Pi}_n A_n^* - \hat{\Pi}_n C_n^* V^{-1} C_n \hat{\Pi}_n + M_n = 0. \tag{30}$$

The conditions under which the state-feedback approximation of an infinite-dimensional system converges is discussed in [2] and [11]. The following results are from [11].

**Definition 9.** *The control systems $(A_n, B_n)$ are uniformly stabilizable if there exists a sequence of feedback operators $K_n \in \mathcal{L}(Z_n; U)$ with $||K_n|| \leq M_1$ for some constant $M_1$ such that $A_n - B_n K_n$ generates $S_{Kn}(t)$ and $||S_{Kn}|| \leq M_2 e^{-\alpha_1 t}$, $M_2 \geq 1, \alpha_1 > 0$.*

**Definition 10.** *The pairs $(A_n, Q_n^{\frac{1}{2}})$ are uniformly detectable if there exists a sequence of feedback operators $F_n \in \mathcal{L}(Y; Z_n)$ with $||F_n|| \leq M_3$ for some constant $M_3$ such that $A_n - F_n Q_n^{\frac{1}{2}}$ generates $S_{Fn}(t)$ and $||S_{Fn}|| \leq M_4 e^{-\alpha_2 t}$, $M_4 \geq 1, \alpha_2 > 0$.*

**Theorem 4.1.** *[11, Thm. 4.3] Let $S_n(t)$ indicate the semigroup generated by $A_n$ and $S(t)$ be the semigroup generated by $A$. Assume $\lim_{n\to\infty} sup_{t\in[t_1,t_2]}||S_n(t)P_n(t)z - S(t)|| = 0$ and $\lim_{n\to\infty} sup_{t\in[t_1,t_2]}||S_n^*(t)P_n(t)z - S^*(t)|| = 0$ for all $z \in Z$ and all* *intervals of time* $[t_1, t_2]$*. In addition, assume*

$$\begin{aligned} &||B_n u - Bu|| \to 0 \text{ for all } u \in U, \\ &||B_n^* P_n z - B^* z|| \to 0 \text{ for all } z \in Z, \\ &||Q_n^{\frac{1}{2}} P_n z - Q^{\frac{1}{2}} z|| \to 0 \text{ for all } z \in Z. \end{aligned} \tag{31}$$

*If* $(A_n, B_n)$ *is uniformly stabilizable and* $(A_n, Q_n^{\frac{1}{2}})$*is uniformly detectable, for each* $n$*, the Riccati equation* (28) *has a unique nonnegative solution,* $\Pi_n$*, with sup*$||\Pi_n|| < \infty$*. There exists constants* $M_5 \geq 1$ *and* $\alpha_5 > 0$*, independent of* $n$*, such that the semigroups* $S_{nK}(t)$ *generated by* $A_n - B_n K_n$ *satisfy*

$$||S_{nK}(t)|| \leq M_5 e^{-\alpha_5 t}, \tag{32}$$

*where* $K_n = R^{-1} B_n^* \Pi_n$*. For sufficiently large* $n$*, the semigroups* $S_{Kn}(t)$ *generated by* $A - BK_n$ *are uniformly exponentially stable or in other words there exists constants* $M_6 \geq 1$ *and* $\alpha_6 > 0$ *such that*

$$||S_{Kn}(t)|| \leq M_6 e^{-\alpha_6 t}. \tag{33}$$

*Furthermore,*

$$\lim_{n \to +\infty} ||\Pi_n P_n z - \Pi z|| = 0 \tag{34}$$

*where* $\Pi$ *is the solution to Riccati equation*

$$A^* \Pi + \Pi A - \Pi B R^{-1} B^* \Pi + Q = 0. \tag{35}$$

*Moreover,*

$$\lim_{n\to+\infty} ||K_n P_n z - Kz|| = 0, \tag{36}$$

*where* $K = R^{-1}B^*\Pi$. *The cost associated with feedback* $K_n z(t)$ *converges to optimal cost*

$$\lim_{n\to\infty} J(-K_n z(t), z_0) = \langle \Pi z_0, z_0 \rangle. \tag{37}$$

The following corollary gives a result for the minimum-variance estimation problem that uses the duality between the linear regulator and the minimum-variance estimation problems.

**Corollary 4.2.** *Let* $S_n(t)$ *and* $A_n$ *be defined as in theorem (4.1) and satisfy the assumptions mentioned in that theorem. Assume*

$$\begin{aligned} &||C_n^* y - C^* y|| \to 0 \quad \textit{for all } y \in Y, \\ &||C_n P_n z - Cz|| \to 0 \quad \textit{for all } z \in Z, \\ &||M_n^{\frac{1}{2}} P_n z - M^{\frac{1}{2}} z|| \to 0 \quad \textit{for all } z \in Z. \end{aligned} \tag{38}$$

*If* $(A_n^*, C_n^*)$ *is uniformly stabilizable and* $(A_n^*, M_n^{\frac{1}{2}})$ *is uniformly detectable, then for each* $n$, *the Riccati equation* (29) *has a unique non-negative solution,* $\hat{\Pi}_n$, *and*

$$\lim_{n\to+\infty} ||\hat{\Pi}_n P_n z - \hat{\Pi} z|| = 0, \tag{39}$$

*where* $\hat{\Pi}$ *is the solution to Riccati equation* (20). *Define* $L_n = \hat{\Pi}_n C_n^* V^{-1}$ *and* $L = \hat{\Pi} C^* V^{-1}$. *Then, the semigroups generated by* $A_n - L_n C_n$ *and* $A - L_n C$ *are uniformly exponentially stable and*

$$\lim_{n\to+\infty} ||L_n y - Ly|| = 0. \tag{40}$$

*Proof.* Consider the dynamical system

$$\dot{z}(t) = A^* z(t) + C^* u(t), t \geq 0, \;\; z(0) = z_0 \tag{41}$$

with the quadratic cost

$$J(z_0; u) = \int_0^\infty \langle M^{\frac{1}{2}} z(s), M^{\frac{1}{2}} z(s) \rangle + \langle u(s), V u(s) \rangle ds. \tag{42}$$

A sequence of finite-dimensional approximations of (41) and (42) are

$$\dot{z}_n(t) = A_n^* z(t) + C_n^* u(t), \;\; z(0) = P_n z_0, \;\; n \geq 1, \tag{43}$$

and

$$J(z_0; u) = \int_0^\infty \langle M_n^{\frac{1}{2}} z(s), M_n^{\frac{1}{2}} z(s) \rangle + \langle u(s), V u(s) \rangle ds \tag{44}$$

respectively. The Riccati equation associated with (41) and (42) is (20) and a sequence of finite-dimensional approximations of (20) is (29). The results of Theorem (4.1) apply to this corollary. Thus, $\lim_{n \to +\infty} ||\hat{\Pi}_n P_n z - \hat{\Pi} z|| = 0$. The semigroups generated by $A_n^* - C_n^* L_n^*$ and $A^* - C^* L_n^*$ are uniformly exponentially stable. It is known that $(A^*, C^*)$ is uniformly stabilizable if and only if $(A, C)$ is uniformly detectable. Similarly, $(A_n^*, C_n^*)$ is uniformly stabilizable if and only if $(A_n, C_n)$ is uniformly detectable [11]. Hence, the semigroups generated by $A_n - L_n C_n$ and $A - L_n C$ are uniformly exponentially stable. In addition, (38) and (39) imply

$$\lim_{n \to +\infty} ||L_n y - L y|| = 0. \tag{45}$$

■

# 5 Infinite-dimensional sampled-data observer

Since measurements are taken discretely in time, the criteria under which the sampled-data observer estimates the state of the original infinite-dimensional system should be investigated. The stability of the infinite-dimensional sampled-data feedback control systems is investigated in [10]. The results can be extended to design sampled-data infinite-dimensional observers that estimate the state of the original infinite-dimensional system. The material of this section can be found in [10].

Consider the dynamical system

$$\dot{z}(t) = Az(t) + Bu(t), \quad z(0) = z_0, \tag{46}$$

where $A$ and $B$ are defined in (2). Suppose that the feedback control $u(t) = Kz(t)$, where $K \in \mathcal{L}(Z; U)$, is an exponentially stabilizing state feedback control for (46) in the sense that $A + BK$ generates an exponentially stable, strongly continuous semigroup on Z. A natural implementation of this continuous-time control $u(t) = Kz(t)$ is to use the sample and hold method, i.e sample the continuously varying states, $z(t)$, and holds its value at a constant level for a specified minimum period of time, i.e.

$$u(t) = Kz(k\tau) \qquad t \in [k\tau, (k+1)\tau)\,, \tag{47}$$

where $k$ is an integer, $\tau > 0$ is the sampling period and integer multiple of $\tau$ are sampling times. The control (47) is called the sampled-data feedback control and the overall system, (46) and (47), is called the sampled-data feedback system. It is expected that for all sufficiently small sampling periods, (47) is a stabilizing control for (46) in the sense that there exists $N_1 \geq 1$ and $v_1 > 0$ such that

$$||z(t)|| \leq N_1 e^{-v_1 t}||z_0||. \tag{48}$$

The following theorem states the necessary conditions for which the sampled-data feedback controller stabilizes the system (46).

**Theorem 5.1.** *[10, Thm. 3.1] Assume that $A$ generates a strongly continuous semigroup $S(t)$ on $Z$, and $B$ is a bounded operator which maps $U$ to $Z$. In addition, assume $K$ is a compact operator and the semigroup generated by $A + BK$ is exponentially stable. Then, there exists $\tau^* > 0$ such that for every $\tau \in (0, \tau^*)$ there exists $N_2 \geq 1$ and $v_2 > 0$ such that all the solutions of the sampled-data feedback* (46) *and* (47) *satisfy*

$$||z(t)|| \leq N_2 e^{-v_2 t}||z_0|| \tag{49}$$

*for all $z_0 \in Z$, and for all $t \geq 0$.*

Consider the dynamical system (46) with the discrete measurement,

$$\begin{aligned} \dot{z}(t) &= Az(t) + Bu(t), z(0) = z_0, \\ y(t) &= Cz(kT), \quad t \in [kT, (k+1)T), \end{aligned} \tag{50}$$

and an observer

$$\begin{aligned} \dot{\hat{z}}(t) &= A\hat{z}(t) + Bu(t) + L[y(t) - \hat{y}(t)], \quad \hat{z}(0) = \hat{z}_0, \\ \hat{y}(t) &= C\hat{z}(kT), \quad t \in [kT, (k+1)T), \end{aligned} \tag{51}$$

where $k$ is an integer, $T > 0$ is the sampling period and integer multiples of $T$ are the sampling times. The observer (51) is called the sampled-data observer.

**Definition 11.** *The estimation error is defined as*

$$e(t) = z(t) - \hat{z}(t), \tag{52}$$

*where* $z(t)$ *and* $\hat{z}(t)$ *are defined in* (50) *and* (51)*.*

The goal is to find conditions under which the sampled-data estimator estimates the state of the original infinite-dimensional system (50). The following corollary gives a result for the sampled-data observer.

**Corollary 5.2.** *Let* $A$ *and* $S$ *be defined as in theorem (5.1) and* $C$ *be a bounded operator. In addition, assume* $L$ *is a compact operator and the semigroup generated by* $A - LC$ *is exponentially stable. Then, there exists* $\tau^* > 0$ *such that for every* $T \in (0, \tau^*)$ *there exists* $N_3 \geq 1$ *and* $v_3 > 0$ *such that all the solutions to*

$$\begin{aligned} \dot{e}(t) &= Ae(t) - L(y(t) - \hat{y}(t)), \\ y(t) &= Cz(kT), \;\; t \in [kT, (k+1)T), \\ \hat{y}(t) &= C\hat{z}(kT), \;\; t \in [kT, (k+1)T), \end{aligned} \tag{53}$$

*satisfy*

$$||e(t)|| \leq N_3 e^{-v_3 t} ||e(0)||, \tag{54}$$

*for all* $e(0) = z_0 - \hat{z}_0$*, and for all* $t \geq 0$*.*

*Proof.* Using (50), (51) and (52),

$$\begin{aligned}
\dot{e}(t) &= \dot{z}(t) - \dot{\hat{z}}(t) \\
&= Az(t) + Bu(t) - \big(A\hat{z}(t) + Bu(t) + L(y(t) - \hat{y}(t)\big) \\
&= A\big(z(t) - \hat{z}(t)\big) - L\big(y(t) - \hat{y}(t)\big) \\
&= Ae(t) - LC\big(z(kT) - \hat{z}(kT)\big) \\
&= Ae(t) - LCe(kT).
\end{aligned} \tag{55}$$

By the assumptions in the corollary statement $A - LC$ generates an exponentially stable semigroup, $L$ is a compact operator and $C$ is bounded. Therefore, Theorem 5.1 applies to (55). There exists $\tau^* > 0$ such that for every $T \in (0, \tau^*)$ there exists $N_3 \geq 1$ and $v_3 > 0$ such that

$$||e(t)|| \leq N_3 e^{-v_3 t} ||e_0||, \tag{56}$$

for all $e(0) = z_0 - \hat{z}_0$, and for all $t \geq 0$. ■

# 6 Numerical application

In this section a Kalman filter is designed for the one-dimensional heat equation with Dirichlet boundary conditions. Continuous-time finite difference, continuous-time Fourier, discrete-time finite difference and discrete-time Fourier methods are used to investigate the convergence of the Kalman filter and its gain.

Finite difference and Fourier methods are used to solve partial differential equations. Finite difference methods are easy to code, cheap to implement on digital machines and easy to formulate for simple geometries. It is not easy to generalize finite difference methods for complex geometries. In addition, the solution is only defined pointwise, so reconstruction at arbitrary locations is not uniquely defined. Finite difference methods are not Galerkin methods, so the convergence may be more difficult to prove. Fourier method satisfies the orthogonality conditions. Hence, the convergence is easy to prove. The method can be generalized for complex geometry and it usually results in high order accuracy. Moreover, it is easy to implement boundary conditions using Fourier method. Fourier method implementation on computers is usually slower than finite difference method. To choose between finite difference and Fourier methods one should consider different factors such as required accuracy and geometry of the domain.

## 6.1 One-dimensional heat equation

Consider a one-dimensional rod of length $L$ that can be heated along its length according to

$$\begin{aligned} \frac{\partial z(x,t)}{\partial t} &= \alpha \frac{\partial^2 z(x,t)}{\partial x^2} + B_1 u(t) + B_2 w_2(t), \\ z(0,t) &= 0, \\ z(L,t) &= 0, \\ z(x,0) &= z_0(x), \end{aligned} \tag{57}$$

where $z(x,t)$ represents the temperature of the rod at time $t$ and position $x$, $z_0(x)$ is the initial temperature profile, and $u(t) \in \mathbb{L}_2([0,\infty];\mathbb{R})$ is the addition of heat along the bar. Choose $Z = \mathbb{L}_2(0,L)$ as the state space and $z(.,t) = \{z(x,t), 0 \le x \le L\}$ as the state of the system. The operator $B_1 \in \mathcal{L}(\mathbb{R};Z)$ describes the input of the system, $B_2 \in \mathcal{L}(\mathbb{R};Z)$ describes the noise on the state of the system, $w(t) : [0,\infty) \to \mathbb{R}$ is a scalar white Gaussian noise with mean zero and variance Q, and $\alpha$ is the thermal diffusivity.

Define $Az = \alpha \frac{\partial^2 z}{\partial x^2}$. Since not all elements of $\mathbb{L}_2(0, L)$ are differentiable, and the boundary conditions need to be considered, define the domain of the operator $A$ as

$$D(A) = \{z \in \mathcal{H}^2(0, L); z(0) = z(L) = 0\} \tag{58}$$

where $\mathcal{H}^2(0, L)$ indicates the Sobolev space of functions with weak second derivatives. The operator $B_1 \in \mathcal{L}(\mathbb{R}; Z)$ is defined by $B_1 u(t) = b_1(x)u(t)$ for some $b_1(x) \in \mathbb{L}_2(0, L)$ [11]. Similarly, the operator $B_2 \in \mathcal{L}(\mathbb{R}; Z)$ can is defined by $B_2 w_2(t) = b_2(x)w_2(t)$ for some $b_2(x) \in \mathbb{L}_2(0, L)$. The dynamical system (57) can be written as

$$\begin{aligned}
&\dot{z}(t) = Az(t) + B_1 u(t) + B_2 w_2(t), \\
&z(0, t) = 0, \\
&z(L, t) = 0, \\
&z(x, 0) = z_0(x).
\end{aligned} \tag{59}$$

The operator A generates a strongly continuous semigroup $S(t)$ on $Z = \mathbb{L}_2(0, L)$. The state z, the temperature of the rod, evolves on the infinite-dimensional space $\mathbb{L}_2(0, L)$, [2].

Define the output of the dynamical system (59) as

$$y(t) = \int_0^L c(x)z(x, t)\mathrm{d}x + v(t) \tag{60}$$

where $c(x) \in \mathbb{L}_2(0, L)$ and $v(t) : [0, \infty) \to \mathbb{R}$ is a scalar white Gaussian noise with mean zero and variance V. The dynamical system (59) with output (60) can be written as

$$\begin{aligned}
&\dot{z}(t) = Az(t) + B_1 u(t) + B_2 w_2(t),\\
&y(t) = Cz(t) + v(t),\\
&Cz = \int_0^L c(x) z(x,t) \mathrm{d}x,\\
&z(0,t) = 0,\\
&z(L,t) = 0,\\
&z(x,0) = z_0(x),
\end{aligned} \tag{61}$$

## 6.2 Problem statement

Consider a rod with length $L$ and thermal diffusivity $\alpha$. Assume

$$b_1(x) = \begin{cases} 1 & : \quad x \in [x_1, x_2] \\ 0 & : \quad otherwise, \end{cases} \tag{62}$$

and

$$b_2(x) = \begin{cases} 1 & : \quad x \in [x_3, x_4] \\ 0 & : \quad otherwise. \end{cases} \tag{63}$$

where $[x_1, x_2]$ and $[x_3, x_4]$ indicate the intervals where the input and noise are present. Suppose one sensor with half-width $d_s$ is placed at $x = x_s$ to measure the temperature. The kernel of the integral in (60) is

$$c(x) = \begin{cases} \frac{1}{2d_s} & : \quad |x - x_s| < d_s \\ 0 & : \quad |x - x_s| > d_s. \end{cases} \tag{64}$$

Even for finite-dimensional systems, the entire state cannot generally be measured. Measurement of the entire state is never possible for systems described by partial differential equations. Hence, designing an observer is necessary for estimating the state of the dynamical system (59). Consider the observer

$$\begin{aligned}
&\dot{\hat{z}}(t) = A\hat{z}(t) + Bu(t) + L[y(t) - \hat{y}(t)], \;\; \hat{z}(0) = \hat{z}_0, \\
&y(t) = Cz(kT) + v(kT), \;\; t \in [kT, (k+1)T), \\
&\hat{y}(t) = C\hat{z}(kT), \;\; t \in [kT, (k+1)T),
\end{aligned} \tag{65}$$

where $k$ is an integer, $T > 0$ is the sampling period and integer multiples of $T$ are the sampling times. The output space $Y$, is finite-dimensional, $C \in \mathcal{L}(Z, Y)$ is defined in (61), $A$, $u(t)$, and $B_1$ are defined in (59), and $L$ is the estimator gain. The goal is to find the $L$ that minimizes (19). To design the estimator (65), for the system (57), given (62), (63) and (64), a sequence of finite-dimensional approximations of the infinite-dimensional system are required. In the following sections, the finite difference and Fourier methods are used to design an estimator for the infinite-dimensional system (57).

## 6.3 Continuous-time finite difference method

A finite difference method can be used to find the finite-dimensional approximations of operators $A$, $B_1$, $B_2$ and $C$ in (61). The state of the system is discretized so that it becomes the temperature of the rod at distinct points. Define

$$\begin{aligned}
&x_i = x_0 + i\Delta x, \;\; i = 1, 2, 3, 4, ..., N, \\
&\Delta x = \frac{L}{N}, \\
&x_0 = 0, x_N = L,
\end{aligned} \tag{66}$$

where $N+1$ is the number of mesh points in $[0, L]$ and $\Delta x$ indicates the distance between adjacent mesh points. A sequence of finite-dimensional approximations of (61) can be written as

$$\dot{z}_n(t) = A_n z_n + B_{1n} u(t) + B_{2n} w(t) \tag{67}$$

with output

$$y(t) = C_n z(kT) + v(kT), \;\; t \in [kT, (k+1)T), \tag{68}$$

where

$$z_n = \begin{bmatrix} z_0(t) \\ z_1(t) \\ z_2(t) \\ \vdots \\ \vdots \\ \vdots \\ z_N(t) \\ z_{N+1}(t) \end{bmatrix}. \tag{69}$$

The finite difference approximations of $B_1$, $B_2$, $C$ and $A$ are $(N+1)\times 1$, $(N+1)\times 1$, $1\times(N+1)$ and $(N+1)\times(N+1)$ matrices

$$A_n = \begin{bmatrix} r & -2r & r & 0 & 0 & 0 & 0 & \dots & 0 \\ r & -2r & r & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & r & -2r & -r & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & r & -2r & r & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \dots & \dots & \dots & \dots & \dots & \dots & r & -2r & r \\ 0 & 0 & 0 & 0 & \dots & \dots & r & -2r & r \end{bmatrix}, \quad B_{1n} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \quad B_{2n} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix},$$

$$C_n = \begin{bmatrix} 0 & 0 & \dots & \dots & \frac{1}{a} & \frac{1}{a} & \dots & \dots & 0 & 0 \end{bmatrix}. \tag{70}$$

where $r = \dfrac{\alpha}{\Delta x^2}$. The non-zero elements of $B_{1n}$ are placed where the mesh points lie in $[x_1, x_2]$. Similarly, the non-zero elements of $B_{2n}$ are placed where the mesh points lie in $[x_3, x_4]$. The non-zero entries of $C_n$ are placed where the mesh points lie in $[x_s - d_s, x_s + d_s]$. Since the distance between adjacent mesh points is $\Delta x = \dfrac{L}{N}$ and the width of sensor is $2d_S$, there are approximately $\dfrac{2d_S}{\Delta x} = \dfrac{2Nd_s}{L}$ mesh points that lie in $[x_s - d_s, x_s + d_s]$. The positive integer $a$ represents the number of mesh points in $[x_s - d_s, x_s + d_s]$ and $\dfrac{1}{a}$ is used for normalization. The central difference method is used to find the approximation of operator $A$ in (61),

$$Az(x_i, t) = \alpha \frac{\partial^2 z(x_i, t)}{\partial x_i^2} = \alpha \frac{z(x_i + \Delta x, t) - 2z(x_i, t) + z(x_i - \Delta x, t)}{\Delta x^2} + O(\Delta x). \tag{71}$$

Since the temperature is fixed at both ends, $z_n$, $A_n$, $B_{1n}$, $B_{2n}$ and $C_n$ matrices can be reduced to $(N-1)\times 1$, $(N-1)\times(N-1)$, $(N-1)\times 1$, $(N-1)\times 1$ and $1\times(N-1)$ matrices for the interior points

$$z_n = \begin{bmatrix} z_1(t) \\ z_2(t) \\ \vdots \\ \vdots \\ \vdots \\ z_{N-2}(t) \\ z_{N-1}(t) \end{bmatrix}, \tag{72}$$

$$A_n = \begin{bmatrix} -2r & r & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ r & -2r & r & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & r & -2r & -r & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & r & -2r & r & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \dots & \dots & \dots & \dots & \dots & \dots & r & -2r & r \\ 0 & 0 & 0 & 0 & \dots & \dots & 0 & r & -2r \end{bmatrix}, \quad B_{1n} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \quad B_{2n} = \begin{bmatrix} 0 \\ 0 \\ . \\ 1 \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix},$$

$$C_n = \begin{bmatrix} 0 & 0 & \dots & \dots & \frac{1}{a} & \frac{1}{a} & \dots & \dots & 0 & 0 \end{bmatrix}. \tag{73}$$

To design a continuous-time Kalman filter, the Riccati equation associated with minimum-variance estimation problem is solved with (73). The Riccati equation associated with the observer (65) and (19) is

$$A\hat{\Pi} + \hat{\Pi}A^* - \hat{\Pi}C^*V^{-1}C\hat{\Pi} + B_2QB_2^* = 0 \tag{74}$$

and the finite-dimensional approximation of (74) is

$$A_n\hat{\Pi}_n + \hat{\Pi}_nA_n^* - \hat{\Pi}_nC_n^*V^{-1}C\hat{\Pi}_n + B_{2n}Q_nB_{2n}^* = 0 \tag{75}$$

where $Q$ and $V$ are variance of $w(t)$ and $v(t)$ respectively. The Kalman gain can be found by solving (74) and using (21). MATLAB is used to solve the algebraic Riccati equation (75). The following figures show the observer gain, true state and estimated state for the different number of mesh points. The physical properties of the system are given in Table 1. Properties of the input function are given in Table 2. It is assumed that the measurements are taken every 1 second, i.e. $T = 1$ s.

Table 1: Physical properties of the system

| | |
|---|---|
| $\alpha(\frac{m^2}{s})$ | 0.001 |
| L$(m)$ | 1 |
| $d_s(m)$ | 0.005 |
| $x_s$ | 0.5 |
| $[x_1, x_2]$ | $[0.125, 0.25]$ |
| $[x_3, x_4]$ | $[0.25, 0.75]$ |
| Q | 5 |
| V | 10 |

Table 2: Initial conditions and input function

| | |
|---|---|
| Input function | $10\sin(t)$ |
| Initial condition (C) | $z_0(x) = 400$ |
| Initial condition of observer (C) | 0 |

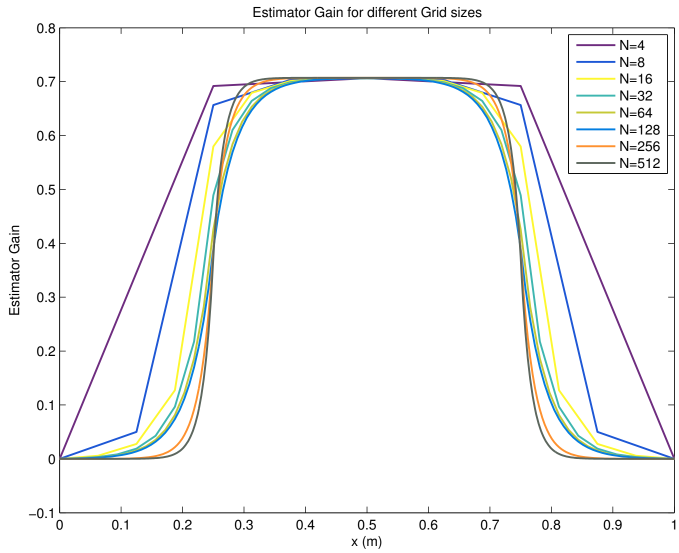
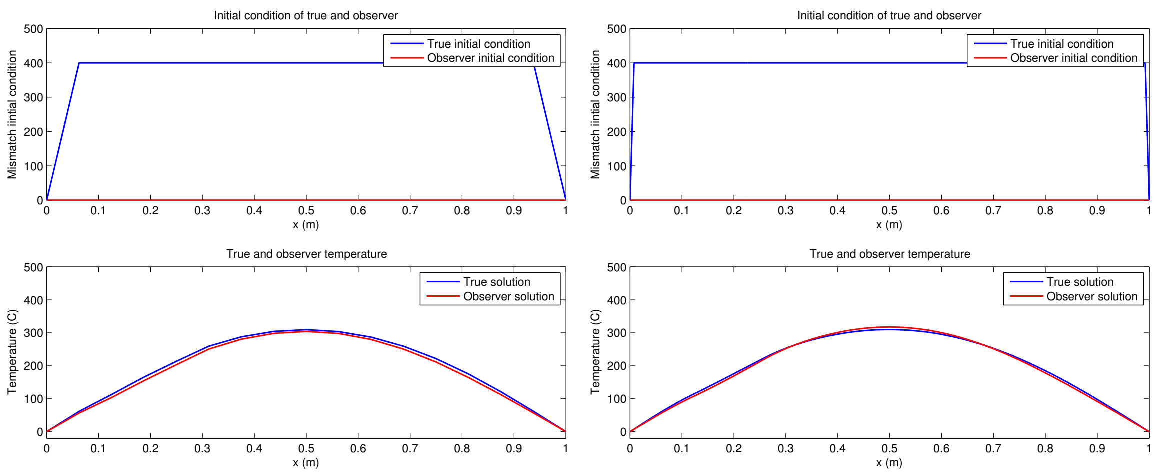



Figure 1: Observer gain for different number of grid points.




(a) True state and estimated state at $t = 50s$ for $N = 16$.

(b) True state and estimated state at $t = 50s$ for $N = 128$.

Figure 2: Snapshot of true state and estimated state at $t = 50$s.

Since the noise on the state of the system (57) is uniformly distributed in $[0.25, 0.75]$, the gain in this interval is expected to be larger than other parts of the domain. As more grid points are considered, the estimated state becomes closer to the true state. Hence, the observer gain approaches zero in regions with no random noise as $N$ increases. Figure 3 illustrates that the estimates state converges to the true state within 100 seconds.

## 6.4 Continuous-time Fourier method

In this section, the estimator (65) is designed using the eigenfunctions of the system (57). The solution to the infinite-dimensional system (57), with no input and no random noise, can be expressed as a sum of eigenfunctions, $\sin(\frac{n\pi x}{L})$, that is

$$z(x,t) = \sum_{i=1}^{\infty} k_n(t) \sin(\frac{n\pi x}{L}). \tag{76}$$

The new state becomes the Fourier sine coefficients of $z(t)$ and the dynamical system (57) can be written in the new state space representation. Consider the Nth order approximation where only the first N coefficients are included. The approximation reduces the infinite-dimensional system to a finite-dimensional system. The finite-dimensional approximation can be written as

$$\dot{z}_n(t) = A_n z_n + B_{1n} u(t) + B_{2n} w(t), \tag{77}$$

with output

$$y(t) = C_n z(kT) + v(kT), \quad t \in [kT, (k+1)T), \tag{78}$$

where

$$z_n(t) = \begin{bmatrix} k_1(t) \\ k_2(t) \\ k_3(t) \\ \vdots \\ k_{N-2} \\ k_{N-1} \\ k_N \end{bmatrix}, \quad A_n = \begin{bmatrix} \frac{-\pi^2}{L^2} & 0 & 0 & 0 & \dots \\ 0 & \frac{-4\pi^2}{L^2} & 0 & 0 & \dots \\ 0 & 0 & \frac{-9\pi^2}{L^2} & 0 & \dots \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \frac{-N^2\pi^2}{L^2} \end{bmatrix}, \quad B_{1n} = \begin{bmatrix} b_{11} \\ b_{12} \\ b_{13} \\ \vdots \\ b_{1(N-2)} \\ b_{1(N-1)} \\ b_{1N} \end{bmatrix}, \tag{79}$$

$$B_{2n} = \begin{bmatrix} b_{21} \\ b_{22} \\ b_{23} \\ \vdots \\ b_{2(N-2)} \\ b_{2(N-1)} \\ b_{2N} \end{bmatrix}, \quad C_n = \begin{bmatrix} c_1 & c_2 & c_3 & \dots & c_{N-2} & c_{N-1} & c_N \end{bmatrix}, \tag{80}$$

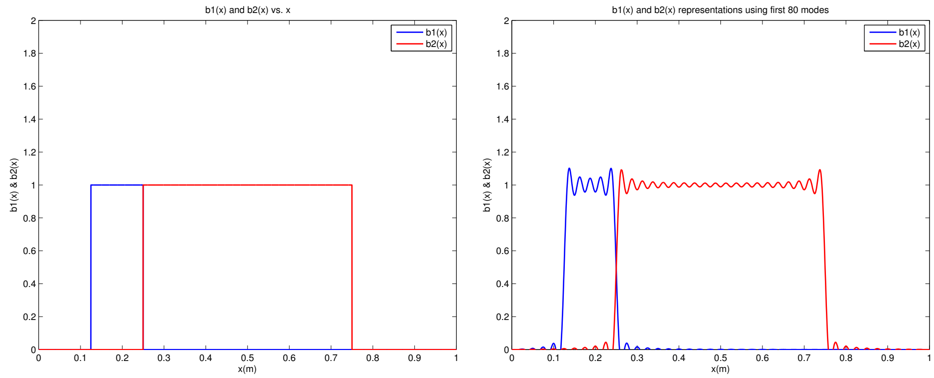
where

$$b_{1n} = \frac{2}{L}\int_0^L b_1(x)\sin(\frac{n\pi x}{L})dx,$$

$$b_{2n} = \frac{2}{L}\int_0^L b_2(x)\sin(\frac{n\pi x}{L})dx, \tag{81}$$

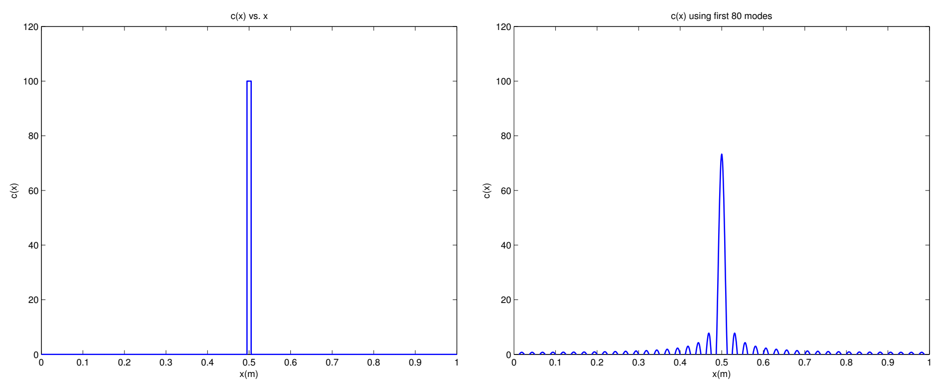
$$c_n = \frac{2}{L}\int_0^L c(x)\sin(\frac{n\pi x}{L})dx.$$

Figures 4 and 5 show $b_1(x)$, $b_2(x)$ and $c(x)$ along with their Fourier approximations.




(a) $b_1(x)$ and $b_2(x)$. (b) Approximation of $b_1(x)$ and $b_2(x)$.

Figure 3: $b_1(x)$ and $b_2(x)$ along with their Fourier approximations, $N = 80$.




(a) $c(x)$. (b) Approximation of $c(x)$.

Figure 4: $c(x)$ along with its Fourier approximation, $N = 80$.

To design a Kalman filter, the Riccati equation (75) is solved using (79), (80) and (81) in MATLAB. The following figures show the Kalman gain for the continuous-time plant, snapshots of the true state and the estimated. Physical properties and initial conditions are the same as in Table 1 and 2.

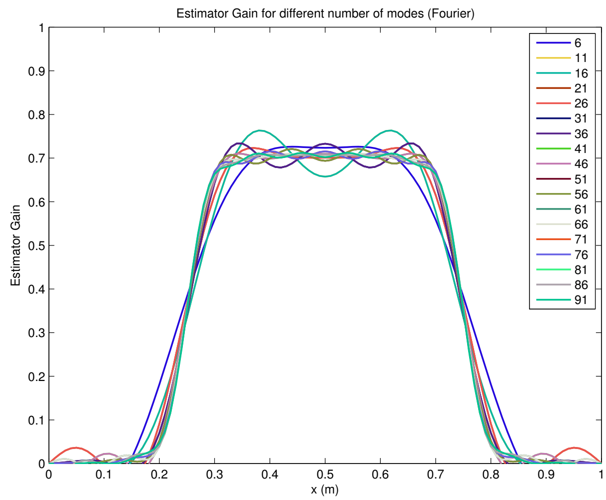
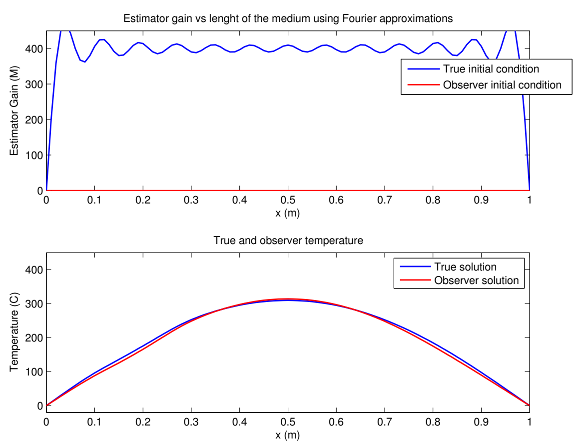
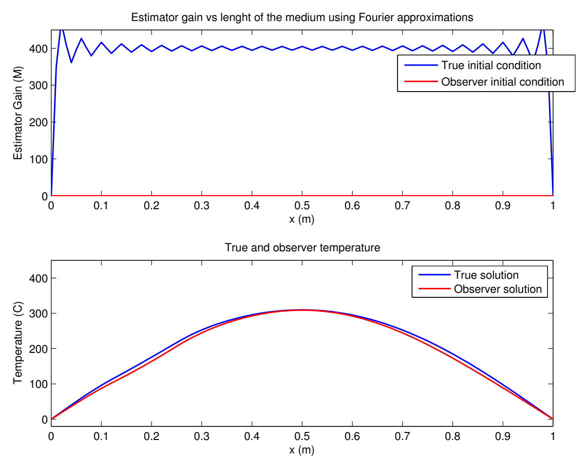



Figure 5: Observer gain for different number of eigenfunctions.




(a) True state and estimated state at $t = 50$s and $N = 25$.



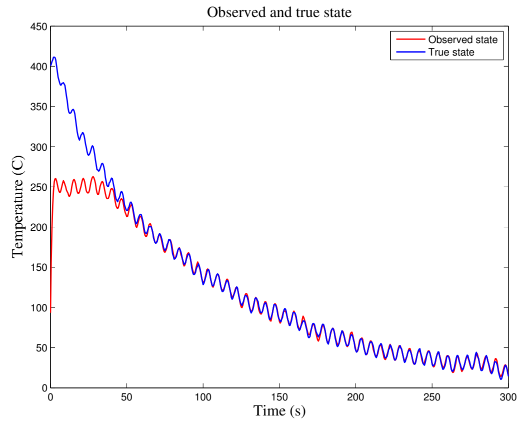

(b) Exact and observer solutions at $t = 50$s and $N = 50$.

Figure 6: True solution and observer for different number of eigenfunctions.

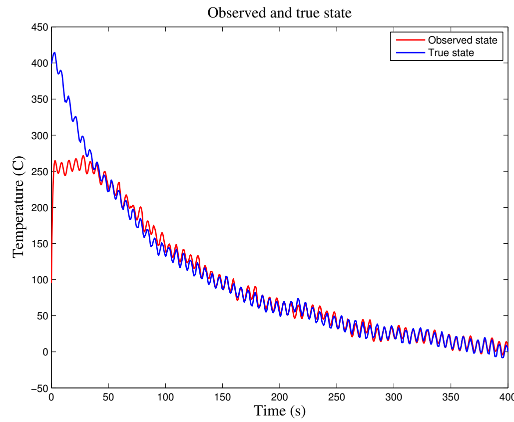
The convergence of the Kalman filter can be explained in two steps. First, note that the finite-dimensional approximations of Kalman filter gains, $L_n$, stabilize the original infinite-dimensional estimator. Secondly, a finite number of sensors are used to measure the temperature, and the output space is finite-dimensional. Hence, L is compact and C is a bounded operator in (65). The hypothesis of corollary (5.2) are satisfied. Therefore, the sampled data finite-dimensional observed state converges to the continuous-time infinite-dimensional state.

## 6.5 Transient response of the sample-data observer

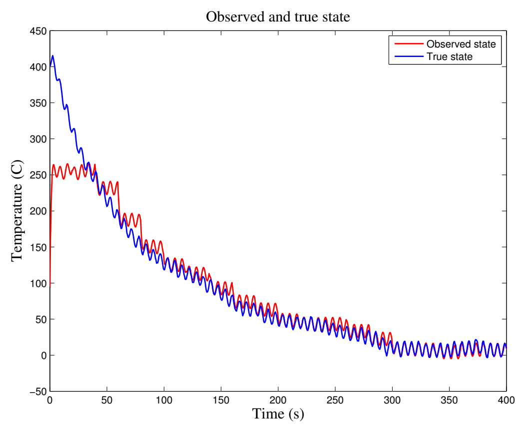
The construction of $\tau^*$ in Theorem 5.1 is not discussed in [10]. Different sampling periods are used in this section to estimate this parameter for the sampled-data observer (65). It is expected intuitively that the estimated state converges to true state faster as sampling periods become smaller. Finite difference and Fourier methods are used to find the estimated state at $x = 0.25m$ for $T = 1s$, $T = 10s$ and $T = 20s$. The following figures illustrate the transient response of the observer for different sampling periods.




(a) Transient response at $x = 0.25ms$, $T = 1s$ and $N = 16$.

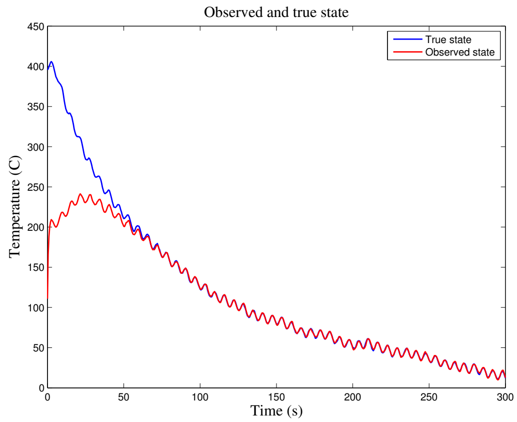



(b) Transient response at $x = 0.25ms$, $T = 10s$ and $N = 16$.

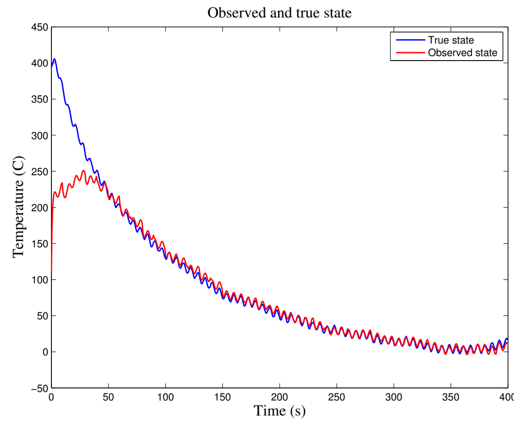



(c) Transient response at $x = 0.25ms$, $T = 20s$ and $N = 16$.

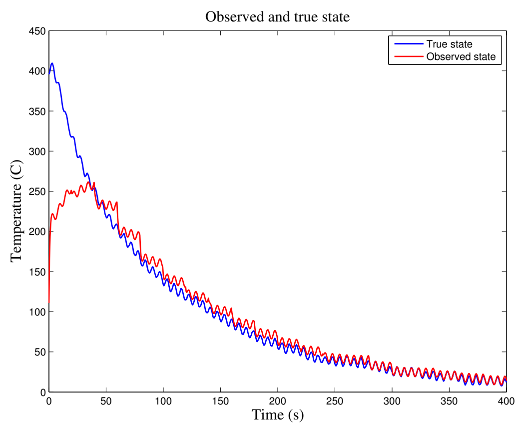
Figure 7: Transient response for different sampling periods (Finite difference method).




(a) Transient response at $x = 0.25ms$, $T = 1s$ and $N = 10$.




(b) Transient response at $x = 0.25ms$, $T = 10s$ and $N = 10$.




(c) Transient response at $x = 0.25ms$, $T = 20s$ and $N = 10$.

Figure 8: Transient response for different sampling periods (Fourier method).

Figure 7 and 8 confirm that the small sampling periods result in high convergence rates. The convergence occurs within 50 seconds when $T = 1s$ is used as a sampling period. When $T = 10s$ or $T = 20s$ , the convergence rates become smaller. It is illustrated in Figure 7 and 8 that the convergence occurs within 150 and 300 seconds respectively.

Other sampling periods ($T = 30s$, $T = 45s$ and $T = 50s$) were also used to find the upper bound for the sampling period. It was observed that the estimated state converges to true state in all cases. One explanation for this observation might be the fact that the upper bound for the sampling period depends on the dynamical system under investigation. Hence, choosing a sampling period might not be an issue for designing the observer (65) for a dynamical system with a simple dynamics similar to (57).

## 6.6 Discrete-time observer

Practical implementation of observers for continuous-time systems is typically performed in discrete-time. To design a discrete-time observer for the dynamical system (57), $A_n$, $B_{1n}$, $B_{2n}$ and $C_n$ in (67), (68), (77) and (78) are replaced by new matrices $A_d$, $B_{1d}$, $B_{2d}$ and $C_d$ [9]. These new matrices are transition matrices of state, disturbance, noise and measurement between the sampling times. In addition, it is necessary to find the power spectral densities of the process and measurement noise for the discrete-time system. The material of this section can be found in [9].

The solution to (67) and (77) can be written as

$$\begin{aligned} z(t) = \exp(A_n(t-t_0))z(t_0) + \left(\int_{t_0}^{t} \exp(A_n(t-\tau))B_{1n}u(\tau)d\tau\right) + \\ \int_{t_0}^{t} \exp\left(A_n(t-\tau)\right)B_{2n}w(\tau)\mathrm{d}\tau. \end{aligned} \tag{82}$$

Suppose $u(t)$ and the measurement are sampled every $T$ seconds. To describe the state propagation between samples, set $t_0 = kT$, $t = (k+1)T$ where $k \geq 0$. Define the sampled state as $z_k := z(kT)$. Then, (82) becomes

$$z_{k+1} = \exp(A_nT)z_k + \left(\int_{kT}^{(k+1)T} \exp(A_n[(k+1)T-\tau])B_{1n}u(\tau)\mathrm{d}\tau\right) + \int_{kT}^{(k+1)T} \exp(A_n[(k+1)T-\tau])B_{2n}w(\tau)\mathrm{d}\tau. \tag{83}$$

Assuming the control input, $u(t)$, is reconstructed from the discrete control sequence $u_k$ by using sample-hold method, $u(\tau)$ has a constant value of $u(kT) = u_k$ over the integration interval. Define

$$w_k = \int_{kT}^{(k+1)T} \exp(A_n[(k+1)T-\tau])B_{2n}w(\tau)\mathrm{d}\tau. \tag{84}$$

Equation (83) becomes

$$z_{k+1} = \exp(A_nT)z_k + \left(\int_{kT}^{(k+1)T} \exp(A_n[(k+1)T-\tau])B_{1n}\mathrm{d}\tau\right)u_k + w_k. \tag{85}$$

By choosing $\lambda = \tau - kT$ and $\tau = T - \lambda$, Equation (85) can be written as

$$z_{k+1} = \exp(A_nT)z_k + \left(\int_{0}^{T} \exp(A_n\tau)B_{1n}\mathrm{d}\tau\right)u_k + w_k. \tag{86}$$

The measurement equation, (68), has no dynamics. Hence, (68) and (78) can be written as

$$y_k = C_nz_k + v_k. \tag{87}$$

The state-space representation of (61) for discrete-time plant can be written as

$$\begin{aligned} z_{k+1} &= A_dz_k + B_{1d}u_k + w_k, \\ y_k &= C_dz_k + v_k, \end{aligned} \tag{88}$$

where

$$
\begin{aligned}
A_d &= \exp(A_n T), \\
B_{1d} &= \int_0^T \exp(A_n \tau) B_{1n} \mathrm{d}\tau, \\
C_d &= C_n, \\
w_k &= \int_k^{(k+1)T} \exp(A_n[(k+1)T - \tau]) B_{2n} w(\tau) \mathrm{d}\tau, \\
v_k &= v(kT).
\end{aligned} \tag{89}
$$

Due to sampling, the variance of random noise should be redefined for the discrete-time system. It is shown in [9] that the variance of measurement and process noise are

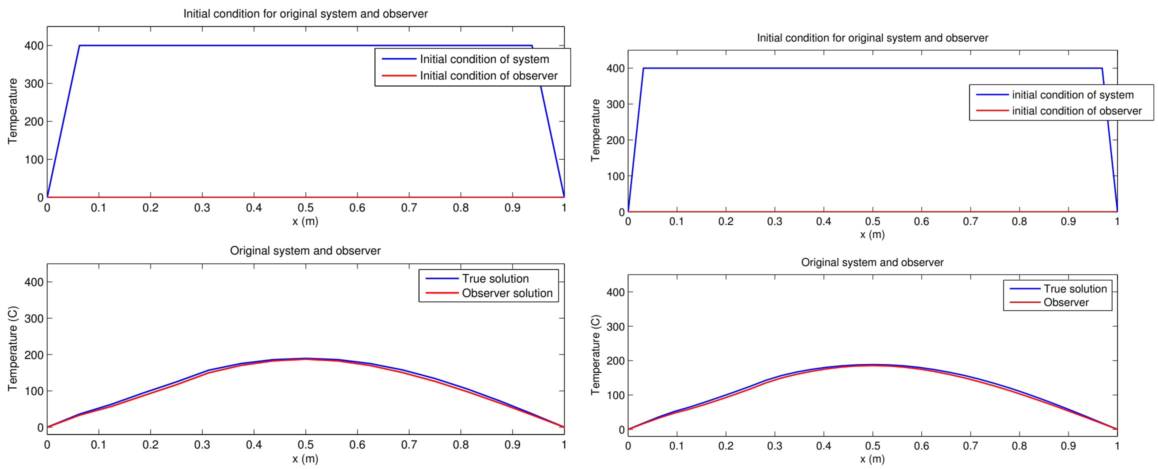
$$
\begin{aligned}
V_d &= \frac{V}{T}, \\
Q_d &= QT + O(T^2),
\end{aligned} \tag{90}
$$

where $T$ is the sampling period and $V$ and $Q$ are the variance of the random noise defined in (57) and (60). The derivation of Kalman filter for the discrete-time plant (88) is discussed in [9]. It is shown that the steady-state Kalman filter for the discrete-time system (88) is

$$
\begin{aligned}
\hat{z}(k+1|k) &= A_d\big(I - K_d C_d\big)\hat{z}(k|k-1) + A_d K_d y_k + B_{1d} u_k, \\
\hat{z}(k+1|k+1) &= \big(I - K_d C_d\big)\hat{z}(k+1|k) + K_d y_{k+1},
\end{aligned} \tag{91}
$$

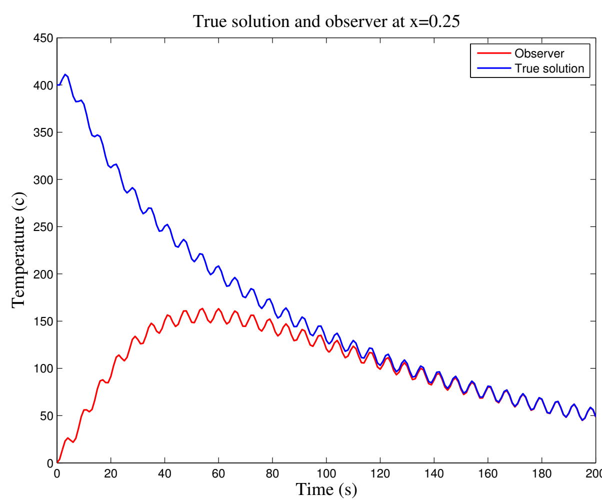
where $K_d$ is the steady-state discrete-time Kalman gain, $\hat{z}(k+1|k)$ is the prediction of the next state using all the previous measurements but excluding the current measurement, and $\hat{z}(k+1|k+1)$ is the corrected estimate using the current measurement, [9].

The following figures show the snapshots of the estimate and the true solution for the discrete-time system associated with (57). The finite difference and Fourier methods are used to design an observer. Data in Table 1 and 2 is used.



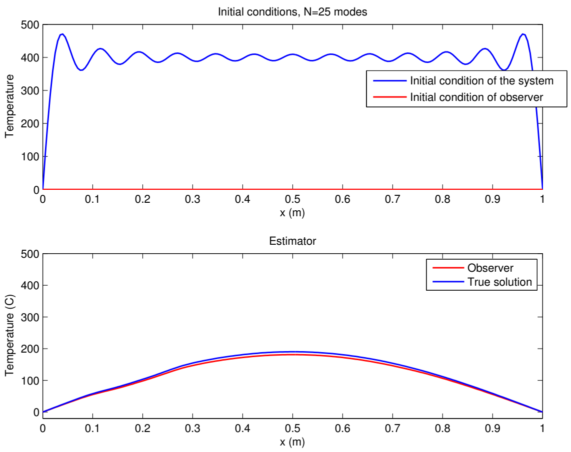

(a) True state and estimated state at t=100, N=16. (b) True state and estimated state at t=100, N=32.

Figure 9: Finite difference, discrete-time Kalman filter.

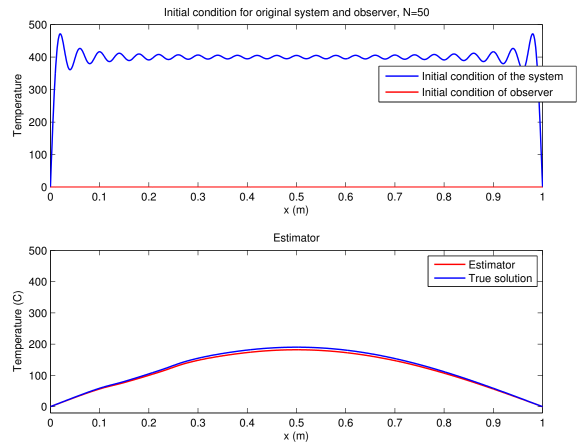



[h]

Figure 10: Finite difference, transient response at $x = 0.25$m, $N = 32$.

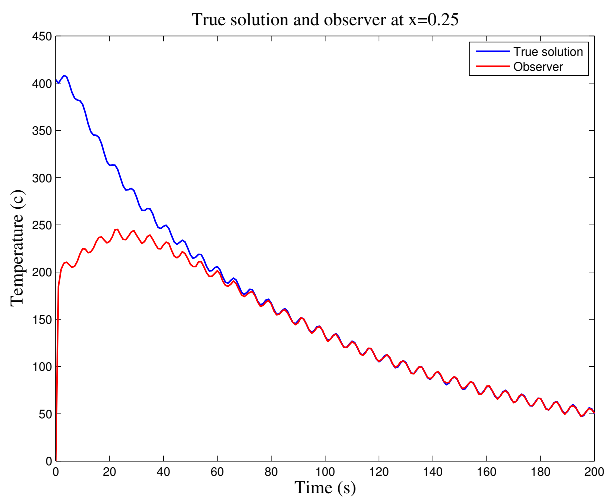



(a) True state and estimated state at t=10, N=25 modes.




(b) True state and estimated state at t=50, N=50 modes.

Figure 11: Fourier approximation, discrete-time Kalman filter.




[h]

Figure 12: Fourier method, transient response at $x = 0.25$m, $N = 25$ modes.

# 7 Conclusion

State estimation of dynamical systems in presence of random noise is crucial in many different applications [6]. State of many dynamical systems evolves in an infinite-dimensional space. Since it is not possible to measure all the states of an infinite-dimensional system, it is necessary to design an observer. Finite-dimensional approximation of an infinite-dimensional system is necessary for numerical computations.

The conditions under which the finite-dimensional approximation of an observer converges to the one that estimates the infinite-dimensional system were presented. In addition, the measurements are often taken discretely in time. The conditions under which the sampled-data infinite-dimensional observer estimates the state of the infinite-dimensional system were also presented.

Finite difference and Fourier methods were used to design a Kalman filter for the one-dimensional heat equation. Continuous and discrete-time Kalman filters were designed numerically in MATLAB. It was shown that the observer gain converges to the observer gain of the infinite-dimensional estimator. Also, the finite-dimensional observer estimates the infinite-dimensional system for both continuous and discrete-time systems.

# Appendix A

The following MATLAB codes are used in section 6 to design Kalman filters for continuous-time and discrete-time systems.

```
function[]=Continious_time_Finite_difference_gain(L,k_max,alpha,x_3,
x_4,x_s,d_s,R,Q)
% Amir Issaei
% SN: 20488100
% Master of Applied Mathematics, University of Waterloo, Spring 2014


% This code produces plot of estimator game for continious-time using a
% finite difference method.

%   L: length of medium
%   k_max: maximum number of grid points
%   alpha: thermal diffucivity
%   R: variance of the noise on masurements
%   Q: variance of the noise on the states
%   x_1: Start of the interval of input
%   x_2: End of the interval of the input
%   x_s: Sensor location
%   d_s: Half-width of the sensor
%------------------------------------------------------------------

%Clear screen and close all windiws
clc;
format long;
close all;

%Start grid size for-loop

for k=2:k_max

% Number of points in our grid
N=2^k;

% Grid size in space
dx=L/N;
```

```
x=[0:dx:L];

%Create A, B_2, and C matrices for the finite-dimensional
%approximation of the system, These matrices are needed to solve ARE
A=make_A(alpha,N,dx);
B=make_B(x,N,x_3,x_4);
C=make_C(x,N,x_s,d_s);

%Find the covariance matrix
W=B*B'*Q;

%Solve continious-time ARE
[p,l,g]=care(A',C',W,R);

% Find gain
M=p*C'*R^(-1);

%Plot estimator gain vs. length of the medium

plot(x,[0;M;0],'LineWidth',1.4,'color',[rand(1,1) rand(1,1) rand(1,1)]);
xlabel('x (m)');
ylabel('Estimator Gain');
title('Estimator Gain for different Grid sizes');
axis([0 1 0 1])
pause(1);

end
end

function[]=Continious_time_Fourier_gain(L,dx,n_max, alpha,R,Q,
x_1,x_2,x_s,d_s)
% Amir Issaei
% SN: 20488100
% Master of Applied Mathematics, University of Waterloo, Spring 2014

% This code solves steady-state, continious-time ARE
% and finds Kalman gain for different number of modes


%   L: Length of the medium
```

```
%   n_max: Maximum number of modes for simulation
%   alpha: Thermal diffusivity constant
%   R: Variance of the measurements noise
%   Q: Variance of process noise
%   x_1: Start point of noise interval
%   x_2: End point of noise interval
%   x_s: Sensor location
%   d_s: Sensor half-width

%---------------------------------------------------------------


% Close all windows and clear screen
clc;
format long;
close all;

%Start mode loop

for n=1:n_max

% Number of modes
N=n;

% Grid size in space
x=(0:dx:L);
%Create A, B_2 and C matrices to solve ARE.
%Use Fourier series approximation
A=make_A_Fourier(alpha,N,L);
B=make_B_Fourier(N,L,x_1,x_2);
C=make_C_Fourier(N,L,x_s,d_s);

%Creat covariance matrix
W=B*B'*Q;



%Solve continious-time ARE
[p,l,g]=care(A',C',W,R);

% Find gain
```

```
M=p*C'*R^(-1);

% Create table of space coefficients for Fourier series

S=Make_space(L,x,N);


%Plot gain in space

M_final=M'*S;
plot(x,M_final','LineWidth',1.4,'color',[rand(1,1) rand(1,1) rand(1,1)]);
ylim([0,1]);
xlabel('x (m)');
ylabel('Estimator Gain');
title('Estimator Gain for different number of modes (Fourier)');
pause(1);

end
end

function[]=Continious_time_observer_FInite_difference(L,k_max,dt,t_final,
alpha, T_initial,R,Q,x_1,x_2,x_3,x_4,x_s,d_s,u_max,y_lim_min,y_lim_max)
% Amir Issaei
% SN: 20488100
% Master of Applied Mathematics, University of Waterloo, Spring 2014


% This code implement Kalman filter desing for the finite-dimensional
% approximation of the system using a finite difference method.


%   L: Length of the medium
%   k_max: Maximum numbe of points on the grid
%   dt: Sampling period of the sensor
%   t_final: Final time of simulation (simulaiton duraiton)
%   alpha: Thermal diffiusivity constant
%   T_initial: Initial condition of the system
%   R: Variance of the noise on the measurements
%   Q: Variance of the noise on the states
%   u_max= Amplitude of the input function
%   y_lim_in: Minimum range of y axis
```

```
%   y_max: Maximum range of y axis
%   x_1: Start point of input
%   x_2: End point of input interval
%   x_3: Start point of noise interval
%   x_4: End point of noise interval
%   x_s: Sensor location
%   d_s: Sensor half-width
%----------------------------------------------------------------------



%Define global variables

global A;
global C;
global B;
global u;
global Y_noise;
global i;
global M;
global B_input;


%Close all windows and clear the screen

clc;
format long;
close all;

%Start for loop for different grid sizes

for k=7:k_max

% Number of points in our grid

N=2^k;
dt_temp=dt;

% Grid size in space

dx=L/N;
```

```
x=[0:dx:L];


%Number of time steps

t_step=t_final/dt;

%Observer initial condition
T_initial_vector=0*ones(N-1,1);

%Exact initial condition
T_initial_exact=T_initial*ones(N-1,1);


%Creat matrices for the system

B_input=make_B_inout(x,N,x_1,x_2);
B=make_B(x,N,x_3,x_4);

W=B*B'*Q;

%Create matrix A for finite dimensional approximaiton

A=make_A(alpha,N,dx);

%Create matrix C for y=C*x. This indicates the location of the sensor.
%Assumed sensor is located in the middle of the rod.

C=make_C(x,N,x_s,d_s);



%Solve ARE
[p,l,g]=care(A',C',W,R);


% Find gain
M=p*C'*R^(-1);

%Plot initial condition
h1=subplot(2,1,1);
```

```
plot(x,[0;T_initial_exact;0],'LineWidth',1.3,'color','b')
hold on
plot(x,[0;T_initial_vector;0],'LineWidth',1.3,'color','r')
set(h1, 'YLim', [0 y_lim_max])
xlabel(h1, 'x (m)');
ylabel(h1, 'Temperature(c)');
title(h1, 'Initial conditions');

% Plot initial condition of the observer
h2=subplot(2,1,2);
plot(x,[0;T_initial_exact;0],'LineWidth',1.3,'color','b')
hold on
plot(x,[0;T_initial_vector;0],'LineWidth',1.3,'color','r')
set(h2, 'YLim', [y_lim_min y_lim_max])
xlabel(h2, 'x (m)');
ylabel(h2, 'Temperature (C)');
title(h2, 'True and observer temperature');
t=0;
indd=find(x==0.25);
T_ess(1,1)=T_initial_vector(indd,1);
T_exx(1,1)=T_initial_exact(indd,1);
sys=ss(A,[B_input],C,0);%,B%
t_span=[0:dt:t_final];
dt_1=dt;

%Simulate the system response
u=u_max*sin(t_span);
Y=lsim(sys,[u'],t_span',T_initial_exact');
Y_noise=Y+sqrt(R)*randn(1,t_step+1)';


% Move forward in time and find true solution and observer

for i=2:t_step+1

[time,X]=ode45(@solve_ode,[t,dt_1],T_initial_vector');
[time_2,X_exact]=ode45(@exact_sol,[t,dt_1],T_initial_exact');
size_X_e=size(X_exact);
T_initial_exact=X_exact(size_X_e(1,1),:);
size_X=size(X);
T_initial_vector=X(size_X(1,1),:);
```

```
% Plot exact solution

h2=subplot(2,1,2);
plot(x,[0,T_initial_exact,0],'LineWidth',1.3,'color','b')
hold on;

%plot observer

plot(x,[0,T_initial_vector,0],'LineWidth',1.3,'color','r')
set(h2, 'YLim', [y_lim_min y_lim_max])
xlabel(h2, 'x (m)');
ylabel(h2, 'Temperature (C)');
title(h2, 'True and observer temperature')
T_es=[0,T_initial_vector,0];
T_ex=[0,T_initial_exact,0];

t=t+dt;
dt_1=t+dt;

T_ess(1,i)=T_es(1,indd);
T_exx(1,i)=T_ex(1,indd);
pause(0.01)
delete(h2);
end

% Plot transient solution at x=0.25m
figure
tempp=[0:dt:t_final];
plot(tempp,T_ess,'r','LineWidth',1.3);
hold on
plot(tempp,T_exx,'b','LineWidth',1.3);
xlabel('t(s)');
ylabel('Temperature(c)');
title('True and observer temperature at x=0.25');
keyboard;
end


%Find exact solution
```

```
function [dx_dt]=solve_ode(t,x)
dx_dt=(A-M*C)*x+B_input*u(1,i-1)+M*Y_noise(i-1,1);
return
end

% Solve continious-time observer
function [dx_dt]=exact_sol(t,x)
dx_dt=A*x+B_input*u(1,i-1);
return
end

end

function[]=Continious_time_observer_Fourier(L,dx,n_max,dt,t_final,alpha,
T_initial,R,Q,x_1,x_2,x_3,x_4,x_s,d_s,u_max,y_lim_min,y_lim_max)
% Amir Issaei
% SN: 20488100
% Master of Applied Mathematics, University of Waterloo, Spring 2014


% This code implement Kalman filter desing for the finite-dimensional
% approximation of the system using Fourier method.


%   L: Length of the medium
%   dx: Space grid size
%   n_max: Maximum numbe of modes
%   dt: Sampling period of the sensor
%   t_final: Final time of simulation (simulaiton duraiton)
%   alpha: Thermal diffiusivity constant
%   T_initial: Initial condition of the system
%   R: Variance of the noise on the measurements
%   Q: Variance of the noise on the states
%   u_max= Amplitude of the input function
%   y_lim_in: Minimum range of y axis
%   y_max: Maximum range of y axis
%   x_1: Start point of input
%   x_2: End point of input interval
%   x_3: Start point of noise interval
%   x_4: End point of noise interval
%   x_s: Sensor location
```

```
%   d_s: Sensor half-width
%----------------------------------------------------------------------



% Define Global variables

global A;
global C;
global Y_noise;
global i;
global u;
global M;
global B_input;

% Clear screen and close all windows
clc;
close all;

%Start mode-loop, i.e increase the number of modes starting from 20
for n=20:n_max

% Number of points in our grid

N=n;

% Space grids

x=[0:dx:L];

% Make a copy of original variance for later use

Q_temp=Q;
R_temp=R;

%Number of time steps

t_step=t_final/dt;

%Initial temperature of the observer
```

```
T_initial_vector=make_initial(N,L,0);


%Exact initial temperature

T_initial_exact=make_initial(N,L,T_initial);


%Creat system matrices using fourier series

A=make_A_Fourier(alpha,N,L);
B_input=make_B_inout_Fouries(N,L,x_1,x_2);
B=make_B_Fourier(N,L,x_3,x_4);
C=make_C_Fourier(N,L,x_s,d_s);
W=B*B'*Q;
S=Make_space(L,x,N);

% Simulate the system

T_initial_sim=T_initial_exact;
sys=ss(A,[B_input],C,0);%,B%
t_span=(0:dt:t_final);
u=u_max*sin(t_span);
Y=lsim(sys,[u'],t_span',T_initial_sim');
Y_noise=Y+sqrt(R_temp)*randn(1,t_step+1)';

% Covariance of continious-plant with discrete measurements

%Solve ARE
[p,l,g]=care(A',C',W,R);

% Find gain
M=p*C'*R^(-1);




%Plot initial condition of the system

h1=subplot(2,1,1);
```

```
plot(x,T_initial_exact'*S,'LineWidth',1.3,'color','b')
hold on
plot(x,T_initial_vector'*S,'LineWidth',1.3,'color','r')
set(h1, 'YLim', [0 y_lim_max])
xlabel(h1, 'x (m)');
ylabel(h1, 'Estimator Gain (M)');
title(h1, 'Estimator gain vs length of the medium using Fourier
approximations');
pause(1)
hold on

%Plot initial condition of the observer
h2=subplot(2,1,2);
plot(x,T_initial_exact'*S,'LineWidth',1.3,'color','b')
hold on
plot(x,T_initial_vector'*S,'LineWidth',1.3,'color','r')
set(h2, 'YLim', [y_lim_min y_lim_max])
xlabel(h2, 'x (m)');
ylabel(h2, 'Temperature (C)');
title(h2, 'True and observer temperature');
indd=find(x==0.25);
T_e_temp=T_initial_exact'*S;
T_ex(1,1)=T_e_temp(1,indd);
T_e_temp=T_initial_vector'*S;
T_es(1,1)=T_e_temp(1,indd);
%Initializa the time
t=0;
clear x_hat;
dt_1=dt;

% Move forward in time to find observer and tru solution

for i=2:t_step+1
[time,X]=ode45(@solve_ode,[t,dt_1],T_initial_vector');
[time_2,X_exact]=ode45(@exact_sol,[t,dt_1],T_initial_exact');
size_X_e=size(X_exact);
T_initial_exact=X_exact(size_X_e(1,1),:);
T_e_temp_1=T_initial_exact*S;
T_ex(1,i)=T_e_temp_1(1,indd);

size_X=size(X);
```

```
T_initial_vector=X(size_X(1,1),:);

%Plot true solution

h2=subplot(2,1,2);
plot(x,T_initial_exact*S,'LineWidth',1.3,'color','b')
hold on

%Plot observer

plot(x,T_initial_vector*S,'r','LineWidth',1.3);
set(h2, 'YLim', [y_lim_min y_lim_max])
xlabel(h2, 'x (m)');
ylabel(h2, 'Temperature (C)');
title(h2, 'True and observer temperature');
t=t+dt;
dt_1=dt+t;
T_e_temp_2=T_initial_vector*S;
T_es(1,i)=T_e_temp_2(1,indd);
pause(0.1)
delete(h2);
end

%Plot transietn solution at x=0.25m
figure;
t_vector=[0:dt:t_final];
plot(t_vector,T_ex,'b','LineWidth',1.3);
hold on
plot(t_vector,T_es,'r','LineWidth',1.3);
keyboard;
end

%Find exact solution
function [dx_dt]=solve_ode(t,x)
dx_dt=(A-M*C)*x+B_input*u(1,i-1)+M*Y_noise(i-1,1);
return
end

%Solve continious-time observer

function [dx_dt]=exact_sol(t,x)
```

```
dx_dt=A*x+B_input*u(1,i-1);
return
end

end

function[]=Discrete_time_Estimator_WD(L,k_max,dt,t_final,
alpha, T_initial,R,Q,x_1,x_2,x_3,x_4,x_s,d_s,u_max,y_lim_min,y_lim_max)
% Amir Issaei
% SN: 20488100
% Master of Applied Mathematics, University of Waterloo, Spring 2014

%In this code we design an observer for continious-time plant with
% discrete measurements for different grid sizes using a finite difference
% method.

%   L: Length of the medium
%   k_max: Maximum numbe of mesh point
%   dt: Sampling period of the sensor
%   t_final: Final time of simulation (simulaiton duraiton)
%   alpha: Thermal diffiusivity constant
%   T_initial: Initial condition of the system
%   R: Variance of the noise on the measurements
%   Q: Variance of the noise on the states
%   u_max= Amplitude of the input function
%   y_lim_in: Minimum range of y axis
%   y_max: Maximum range of y axis
%   x_1: Start point of noise interval
%   x_2: End point of noise interval
%   x_3: Start point of input interval
%   x_4: End point of input interval
%   x_s: Sensor location
%   d_s: Sensor half-width
%   u_max: Amplitude of input function
%   y_lim_min: Minimum of the Y axis
%   y_lim_max: Maximum of the Y axis
%---------------------------------------------------------------------------

% Define global variables

global A_d;
```

```
global C_d;
global Y_noise;
global i;
global u;
global M;
global B_input;



% Close all windows and clear screen
clc;
format long;
close all;

%Start for loop for different grid sizes

for k=5:k_max

% Number of points on the grid
N=2^k;

% Grid size in space

dx=L/N;
x=[0:dx:L];


%Number of time steps for
t_step=t_final/dt;

%Initial temperature of the rod for estimator desing.
%Since we do not know what it is, it is set to zero.
% This is a miss-match initial condition

T_initial_vector=0*ones(N-1,1);

%Exact initial temperature;
T_initial_exact=T_initial*ones(N-1,1);

%Creat A, B_1,B_2 and C matrices to design the estimator
```

```
A=make_A(alpha,N,dx);
B=make_B(x,N,x_1,x_2);
B_input=make_B_inout(x,N,x_3,x_4);
C=make_C(x,N,x_s,d_s);
D=0;

%Transform the  continious-time plant to discrete-time plant
sysc=ss(A,B_input,C,D);
sysd=c2d(sysc,dt,'zoh');
[A_d,B_d,C_d,D]=ssdata(sysd);
[Kfd2,Pd2]=lqed(A,B,C,Q,R,dt);

%Keep copy of original variances for later use

Q_temp=Q;
R_temp=R;

% Find gain of Kalman filter
M=Kfd2;

% Update the covariance matrix and variance of measurement
% and process noise

W=B*B'*Q*dt;
R=R/dt;
Q=Q*dt;




%Plot the exact initial condition

h1=subplot(2,1,1);
plot(x,[0;T_initial_exact;0],'LineWidth',1.3,'color','b')
hold on
plot(x,[0;T_initial_vector;0],'LineWidth',1.3,'color','r')
set(h1, 'YLim', [0 y_lim_max])
xlabel(h1, 'x (m)');
ylabel(h1, 'Temperature');
title(h1, 'Initial condition for original system and observer');
```

```
% Plot observer initial temperature

h2=subplot(2,1,2);
plot(x,[0;T_initial_exact;0],'LineWidth',1.3,'color','b')
hold on
plot(x,[0;T_initial_vector;0],'LineWidth',1.3,'color','r')
set(h2, 'YLim', [y_lim_min y_lim_max])
xlabel(h2, 'x (m)');
ylabel(h2, 'Temperature (C)');
title(h2, 'Original system and observer');


%Initializa the time

t=0;

%Simulate the original system to find noisy measurmeents

sys=ss(A,[B_input],C,0);
t_span=[0:dt:t_final];
u=u_max*sin(t_span);
Y=lsim(sys,[u'],t_span',T_initial_exact');
Y_noise=Y+sqrt(R_temp)*randn(1,t_step+1)';
indd=find(x==0.25);
T_es(1,1)=T_initial_vector(indd,1);
T_ex(1,1)=T_initial_exact(indd,1);

%Create a vector of observer for one time-step

x_final=zeros(t_step+1,N-1);
dt_1=dt;

%Move forward in time and solve steady-state Kalman-filter

for i=2:t_step+1

%Predict the states
x_hat=A_d*(eye(N-1)-M*C_d)*T_initial_vector+A_d*M*Y_noise(i-1,1)+
B_d*u(1,i-1);

%Correct the states
```

```
x_final(i,:)=(eye(N-1)-M*C_d)*x_hat+M*Y_noise(i,1);


%Plot the true solution and observer

[time_2,X_exact]=ode45(@exact_sol,[t,dt_1],T_initial_exact');
size_X_e=size(X_exact);
T_initial_exact=X_exact(size_X_e(1,1),:);
h3=subplot(2,1,2);
plot(x,[0,T_initial_exact,0],'LineWidth',1.3,'color','b')
hold on;

plot(x,[0,x_final(i,:),0],'LineWidth',1.3,'color','r');
T_initial_vector=x_final(i,:)';
set(h3, 'YLim', [y_lim_min y_lim_max])
xlabel(h3, 'x (m)');
ylabel(h3, 'Temperature (C)');
title(h3, 'Original system and observer');

%Make a copy of solution for transient resonse

T_es_temp=[0,x_final(i,:),0];
T_ex_temp=[0,T_initial_exact,0];
T_es(1,i)=T_es_temp(1,indd);
T_ex(1,i)=T_ex_temp(1,indd);
t=t+dt;
dt_1=t+dt;
pause(0.1)
delete(h3);
end

% Plot transietn response at x=0.25m
figure;
t_vector=[0:dt:t_final];
plot(t_vector,T_es,'r','LineWidth',1.3);
hold on
plot(t_vector,T_ex,'b','LineWidth',1.3);
keyboard;
end

% Function to find the true solution of the system
```

```
function [dx_dt]=exact_sol(t,x)
dx_dt=A*x+B_input*u(1,i-1);
end
end
```

```
function[]=Discrete_time_Estimator_Fourier_WD(L,dx,n_max,dt,t_final,
alpha, T_initial,R,Q,x_1,x_2,x_3,x_4,x_s,d_s,u_max,y_lim_min,y_lim_max)
% Amir Issaei
% SN: 20488100
% Master of Applied Mathematics, University of Waterloo, Spring 2014

% This code implement Kalman filter desing for the finite-dimensional
% approximation of the system using Fourier method.


%   L: Length of the medium
%   dx: Space grid size
%   n_max: Maximum numbe of modes
%   dt: Sampling period of the sensor
%   t_final: Final time of simulation (simulaiton duraiton)
%   alpha: Thermal diffiusivity constant
%   T_initial: Initial condition of the system
%   R: Variance of the noise on the measurements
%   Q: Variance of the noise on the states
%   u_max= Amplitude of the input function
%   y_lim_in: Minimum range of y axis
%   y_max: Maximum range of y axis
%   x_1: Start point of input
%   x_2: End point of input interval
%   x_3: Start point of noise interval
%   x_4: End point of noise interval
%   x_s: Sensor location
%   d_s: Sensor half-width
%----------------------------------------------------------------------



% Define Global variables

global A_d;
global C_d;
```

```
global Y_noise;
global i;
global u;
global M;
global B_input;

% Clear screen and close all windows
clc;
close all;

%Start mode-loop, i.e increase the number of modes starting from 20
for n=20:n_max

% Number of points in our grid

N=n;

% Space grids

x=[0:dx:L];

% Make a copy of original variance for later use

Q_temp=Q;
R_temp=R;

%Number of time steps

t_step=t_final/dt;

%Initial condition of the observer
T_initial_vector=make_initial(N,L,0);


%Exact initial temperature

T_initial_exact=make_initial(N,L,T_initial);


%Creat system matrices using Fourier series
```

```
A=make_A_Fourier(alpha,N,L);
B_input=make_B_inout_Fouries(N,L,x_1,x_2);
B=make_B_Fourier(N,L,x_3,x_4);
C=make_C_Fourier(N,L,x_s,d_s);


% Simulate the system to find discrete sensor measurements


T_initial_sim=T_initial_exact;
sys=ss(A,[B_input],C,0);%,B%
t_span=[0:dt:t_final];
u=u_max*sin(t_span);
Y=lsim(sys,[u'],t_span',T_initial_sim');
Y_noise=Y+sqrt(R_temp)*randn(1,t_step+1)';

% Covariance of continious-plant with discrete measurements
R=R/dt;
Q=Q*dt;

% Equivalent matrices for continious-plant with discrete measurements
D=0;
sysc=ss(A,B_input,C,D);
sysd=c2d(sysc,dt,'zoh');
[A_d,B_d,C_d,D]=ssdata(sysd);

% Find steady-state Kalman gain

[Kfd2,Pd2]=lqed(A,B,C,Q_temp,R_temp,dt);
S=Make_space(L,x,N);
M_p=Kfd2'*S;
M=Kfd2;





%Plot The exact initial condition
h1=subplot(2,1,1);
plot(x,T_initial_exact'*S,'LineWidth',1.3,'color','b')
```

```
hold on
plot(x,T_initial_vector'*S,'LineWidth',1.3,'color','r')
set(h1, 'YLim', [0 y_lim_max])
xlabel(h1, 'x (m)');
ylabel(h1, 'Temperature');
title(h1, 'Initial condition for original system and observer');
pause(1)
hold on
%Plot the initial of observer
h2=subplot(2,1,2);
plot(x,T_initial_exact'*S,'LineWidth',1.3,'color','b')
hold on
plot(x,T_initial_vector'*S,'LineWidth',1,'color','r')
set(h2, 'YLim', [y_lim_min y_lim_max])
xlabel(h2, 'x (m)');
ylabel(h2, 'Temperature (C)');
title(h2, 'Original system and observer');
indd=find(x==0.25);
T_e_temp=T_initial_exact'*S;
T_ex(1,1)=T_e_temp(1,indd);
T_e_temp=T_initial_vector'*S;
T_es(1,1)=T_e_temp(1,indd);




%Initializa the time
t=0;
clear x_hat;
x_final=zeros(t_step+1,N);

% Move forward in time and find observer and true solution
dt_1=dt;
for i=2:t_step+1

x_hat=A_d*(eye(N)-M*C_d)*T_initial_vector+A_d*M*Y_noise(i-1,1)
+B_d*u(1,i-1);
x_final(i,:)=(eye(N)-M*C_d)*x_hat+M*Y_noise(i,1);
[time_2,X_exact]=ode45(@exact_sol,[t,dt_1],T_initial_exact');
size_X_e=size(X_exact);
T_initial_exact=X_exact(size_X_e(1,1),:);
```

```
T_e_temp_1=T_initial_exact*S;
T_ex(1,i)=T_e_temp_1(1,indd);

T_initial_exact_temp=T_initial_exact;
h2=subplot(2,1,2);
plot(x,x_final(i,:)*S,'LineWidth',1.3,'color','r');
hold on
plot(x,T_initial_exact_temp*S,'LineWidth',1.3,'color','b')
T_initial_vector=x_final(i,:)';
set(h2, 'YLim', [y_lim_min y_lim_max])
xlabel(h2, 'x (m)');
ylabel(h2, 'Temperature (C)');
title(h2, 'Estimator');

t=t+dt;
dt_1=dt+t;

T_e_temp_2=T_initial_vector'*S;
T_es(1,i)=T_e_temp_2(1,indd);
pause(0.1)
delete(h2);
end

% Plot transietn solution at x=0.25m
figure;
t_vector=[0:dt:t_final];
plot(t_vector,T_ex,'b','LineWidth',1.3);
hold on
plot(t_vector,T_es,'r','LineWidth',1.3);
end

% Find exact solution of the system
function [dx_dt]=exact_sol(t,x)
dx_dt=A*x+B_input*u(1,i-1);
return
end

end
```

# References

[1] R.C. Booton. An optimization theory for time-varying linear systems with non-stationary statistical inputs. *Proceedings of the IRE*, 40(8):977–981, 1952.

[2] R.F. Curtain and H. Zwart. *An introduction to infinite-dimensional linear systems theory*, volume 21. Springer, 1995.

[3] S. Darlington. Linear least-squares smoothing and prediction, with applications. *Bell System Technical Journal*, 37(5):1221–1294, 1958.

[4] R.C. Davis. On the theory of prediction of nonstationary stochastic processes. *Journal of Applied Physics*, 23(9):1047–1053, 2004.

[5] J.S. Gibson and A. Adamian. Approximation theory for linear-quadratic-gaussian control of flexible structures. *SIAM journal on control and optimization*, 29(1):1–37, 1991.

[6] T. Jiang, N.D. Sidiropoulos, and G.B. Giannakis. Kalman filtering for power estimation in mobile communications. *Wireless Communications, IEEE Transactions on*, 2(1):151–161, 2003.

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

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

[9] F.L. Lewis, L. Xie, and D. Popa. *Optimal and robust estimation: with an introduction to stochastic control theory*, volume 26. CRC, 2008.

[10] H. Logemann, R. Rebarber, and S. Townley. Stability of infinite-dimensional sampled-data systems. *Transactions of the American Mathematical Society*, 355(8):3301–3328, 2003.

[11] K.A. Morris. Control of systems governed by partial differential equations. *The Control Theory Handbook, 2010.*

[12] F. Riesz and B. Sz.Nagy. Lectures on functional analysis. *Mir, Moscow*, 1979.

[13] K.W. Simon and A.R. Stubberud. Duality of linear estimation and control. *Journal of Optimization Theory and Applications*, 6(1):55–67, 1970.

[14] J.L. Speyer and W.H. Chung. *Stochastic processes, estimation, and control*, volume 17. SIAM, 2008.

[15] N. Wiener. *Extrapolation, interpolation, and smoothing of stationary time series: with engineering applications*, volume 8. MIT press, 1964.