CEAS-GNC-2024-087

EuroGNC Navigation Guidance Control CEAS Bristol, UK June 11th-13th 2024
University of BRISTOL ROYAL AERONAUTICAL SOCIETY AIAA

Kalman Filter-Based Framework for Predicting Aerial Telerobotic Operations with Input and Output Delays

Nazmus Sakib Graduate Student, Aerospace and Ocean Engineering, Virginia Tech, Blacksburg, VA, USA. nazmus.sakib@vt.edu

Zakia Ahmed Graduate Student, Mechanical Engineering, Virginia Tech, Blacksburg, VA, USA. zakiaahmed@vt.edu

Craig A. Woolsey Professor, Aerospace and Ocean Engineering, Virginia Tech, Blacksburg, VA, USA. cwoolsey@vt.edu

ABSTRACT

This paper concerns the development and implementation of a novel model-based predictor for the position and orientation of a telerobotic system subject to time delays. The considered time delays are present both in the input and the output, meaning an input applied at the current time affects the system after some delay and the result is presented to the operator after another delay. Mathematical descriptions of two predictors are presented: a Smith predictor (SP), which is widely used for mitigating time-delayed systems, and a Kalman predictor (KP). The SP and KP are then implemented on simulated flight data generated using a small, fixed-wing uncrewed aerial vehicle model. The KP provides more accurate predictions of all the aircraft states when compared to the SP. The simulations are generated for two measurement noise conditions and it is observed that the KP is more robust to measurement noise than the SP.

1 Introduction

Rapid advances in cyber-physical systems are redefining how humans interact with machines, with new technologies such as touch interfaces, augmented reality displays, etc. The field of telerobotic operation, in particular, is expected to experience positive growth due to these technological advancements. Sheridan [1] defined "teleoperation" as an extension of an individual's ability to perceive and interact with a distant environment and "telerobotics" represents a type of teleoperation where a human operator assumes the role of an overseer. The human operator periodically communicates with a subordinate "telerobot," providing details regarding objectives, limitations, strategies, unforeseen circumstances, expectations, suggestions, and instructions pertaining to a specific task. In return, the operator receives updates on progress, challenges, concerns, and, when necessary, unprocessed sensory data from the robot. The telerobot carries out the task by combining the information received from the human operator with its own artificial perception and intelligence.

Telerobotics is a broad field within robotics that has a diverse array of applications like space missions [2], military endeavors [3], underwater exploration [4], mining [5], handling of hazardous materials [6], construction site monitoring [7, 8], and robotic-assisted medical procedures [9]. Any recreational, academic, military, or business operations involving uncrewed aerial vehicles (UAVs) are thus teleoperations by the definition given in the previous paragraph. The main focus of this paper is telerobotic UAV

operation. A significant challenge for any telerobotic operation is achieving seamless and responsive control over a robot or robotic system when operated remotely by a human operator. This challenge can be attributed to several factors: time delay, limited sensor feedback, data transmission bandwidth, and operator training. Of these factors, time delay is the most critical. In the scenarios considered here, the total time delay is the time difference between a human operator's control input and the telerobot's corresponding sensory feedback to the operator [10]. According to [10, 11], predictive feedback augmentations (such as visual display or haptic feedback) can mitigate delay-induced degradation of a telerobotic operation. Sheridan [2] considers a predictive display to have a computer-generated visual indicator that displays the motion of the telerobotic system. The indicator is then projected ahead in time to offer the operator immediate understanding of what might occur to the system given its current state and operator inputs.

Numerous predictive display algorithms have been developed to mitigate time delays in teleoperations. Brudnak's predictive display [12] algorithm has been applied to a teleoperated ground vehicle in simulation. Jung *et al* [13] have developed another predictive display algorithm for a robot-camera system that mimicked human head-neck motion. Cox *et al* [14] implemented their predictive display algorithm in UAV-based teleoperations and performed real-time flight tests. While these methods are successful in mitigating time delay, there is a loss of information presented to the human operator by the corresponding predictive display; see Figs. 29-30 in [12], Figs. 4-5 in [13], and Fig. 8 in [14].

In prior work [15, 16], the authors of this paper described a novel predictive display algorithm that addresses the loss of information issue while mitigating time delays. The predictive display is capable of mitigating time delays without limiting the visual information through the use of heterogeneous stereo cameras. Additionally, the model-based predictive algorithm can perform well under low data transmission bandwidth and the authors show, through human subject experiments, that additional operator training is not required when the predictive display is active. The use of heterogeneous cameras is one of the novel elements in [15, 16]. These cameras are not popular in computer vision applications because of complex hardware setup, higher computational requirements, and lack of existing theory. However, Kang *et al* [17] showed that heterogeneous stereo systems can be used similarly to homogeneous stereo systems in performing different tasks like object tracking and depth perception. The heterogeneous system used in [15, 16] consists of a pan-tilt-zoom (PTZ) camera for "central vision" and an omnidirectional camera with fisheye lenses for "peripheral vision". The rationale for using these two different cameras is to use the high-resolution imagery of the PTZ camera for human-supervised structural inspection, for example, or for defect detection by a trained artificial intelligence algorithm [18], while the lower-quality imagery from the wide field-of-view omnidirectional camera provides supplemental information to improve the human operator's situational awareness. The predictive display algorithm developed based on such a heterogeneous system backfills empty regions in the predicted pan-tilt-zoom camera view with imagery from the wide field-of-view omnidirectional camera; see Fig. 7 in [16]. This combination of camera imagery is achieved by comparing the currently available PTZ camera position and orientation with a model-based prediction of the camera position and orientation due to operator input. The primary focus of this paper is to implement and compare model-based predictors for the motion of a fixed-wing aircraft subject to time delays in telerobotic operation. The outputs of the predictor can then be used as inputs to the predictive display algorithm to mitigate the effect of delays in the telerobotic operation.

The paper is organized as follows: Section 2 defines the problem associated with telerobotic operations and reviews relevant literature. Section 3 provides a mathematical description of the telerobotic system with delays and lists the assumptions. Section 4 derives the mathematical formulas for two predictors, noting the differences between them. Section 5 describes the fixed-wing aircraft model used for simulations. Section 6 contains the parameters used in the simulation studies. Section 7 describes the results of the simulation and presents some key findings. Finally, Section 8 provides conclusions from the effort.

2 Overview of the Problem and Prior Work

A representative telerobotic operation scheme is illustrated in Fig. 1a where a human operator sends commands to a system or a plant. The input commands experience time delays in the communication network. These delayed inputs then affect the plant and the corresponding output is generated. However, this output is again delayed as it uses the same communication network as the input. This delayed output finally reaches the system and the feedback reaches the human operator through some interfacing device, such as a visual display. The human operator does not see the effect of an input until the total time delay, the sum of the input and the output delays, has passed.

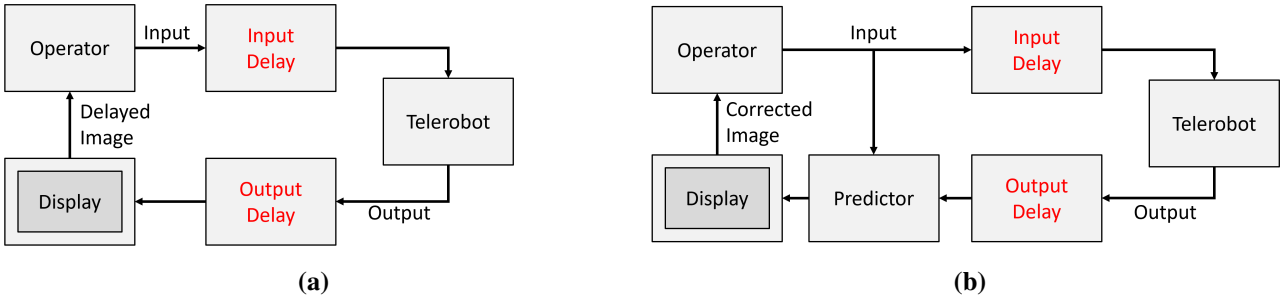


Fig. 1 (a) An overview of the telerobotic operation. (b) Telerobotic operation with a predictive display.

A predictive display takes the predicted outputs from a predictor and modifies the feedback to show the human operator what would happen almost instantly. The human operator receives the information immediately after issuing a command without needing to wait through the total time delay. As mentioned in Section 1, the focus of this paper is on the implementation of a predictor that can be used by the predictive display algorithm developed in [15, 16] that predicts over both input and output delays. The predictor has access to the past and current commands sent to the plant and the delayed outputs sent from the plant. Based on some known model of the plant, the predictor then predicts the output that would be observed if there were no delays present in the system.

The Smith predictor (SP) [19] is the most well-known time delay compensation technique. It has been widely used in industry since its introduction. Originally designed for chemical engineering processes, the SP aims to improve the performance of classical proportional-integral-derivative (PID) controllers in a system with large time delays. Application of the SP can be found in controlling chemical processes [20, 21] where the focus is to compensate for the destabilizing effect of time delay in the feedback loop in order to recover the response of a delay-free system. Studies done in [22–25] combine the time delay compensation technique of the SP with different PI and PID controllers and show that controllers using the predicted output from the SP can stabilize otherwise unstable plants. Other studies have provided parameter tuning guidelines for the SP-based controller when the delay is unknown [26, 27]. Several modifications and improvements of the SP have been proposed in the last few decades to control a delayed process with an integrator [28], to design an adaptive controller for a system with time delays [29], to analyze plants with time delays and saturation [30], and to compensate for time delays in nonlinear plants [31, 32]. Aside from chemical engineering processes, the SP has also been used in tracking a path by ground-based mobile robots [33, 34], increasing position tracking accuracy and minimizing drift in space telerobotic systems [35], and designing predictive compensators for time delay mitigation in UAVs [36].

Some of the weaknesses of the SP, according to [37], are:

- The SP requires on an accurate model of the time delay in the system. If the delay is not accurately modeled or if it varies over time, its performance can be compromised.
- The SP requires an accurate mathematical model of the plant. Even a small inaccuracy in the model can lead to suboptimal or unstable control performance.

- The SP is primarily designed for linear systems with repeatable time delays. It can be challenging to apply this technique to nonlinear systems or systems with varying delay characteristics.
- The SP performs poorly when there is a lot of process or measurement noise.
- When implementing the SP, incorrect estimates of the initial conditions result in incorrect prediction.

Besides the SP, there are other predictive schemes like Moore's analytical predictor (AP) [38]. Designed specifically for discrete-time systems, Moore's proposed algorithm includes a sampling period correction. Wong *et al* [39] showed that the prediction of the AP matches that of the SP only for the special case of a first-order system with time delays. It isn't easy to design APs for higher-order process models and the predictor was developed for proportional-integral (PI) feedback, which limits its applicability. Other prediction approaches exist like using Taylor series approximations to nonlinear systems [40] or disturbance observer-based prediction [41]. One study focused on controlling time-delayed systems with a steady-state Kalman predictor (KP) [42] where the authors use the Kalman filter-based state estimator to obtain the state estimate and a predictor to propagate the estimated state to overcome the time delay. The study also showed that a KP can filter out past disturbances and noise affecting system.

In the work mentioned above, the focus was to control time-delayed systems using predicted measurements or state information. The development of the predictors was a by-product of the original goal. Here, we focus entirely on the development, comparison, and performance analysis of the predictors. The primary contributions of this paper are to:

- develop a generalized Kalman predictor (KP) framework for predicting the states or the outputs of a discrete-time telerobotic system containing both input and output delays.
- compare the predictor performance of the KP with the SP under the influence of process and measurement noise.
- analyze the performance of the KP and the SP in predicting the motion of a fixed-wing aircraft using a six-degree-of-freedom (6-DoF), nonlinear dynamic model.

3 Mathematical Preliminaries

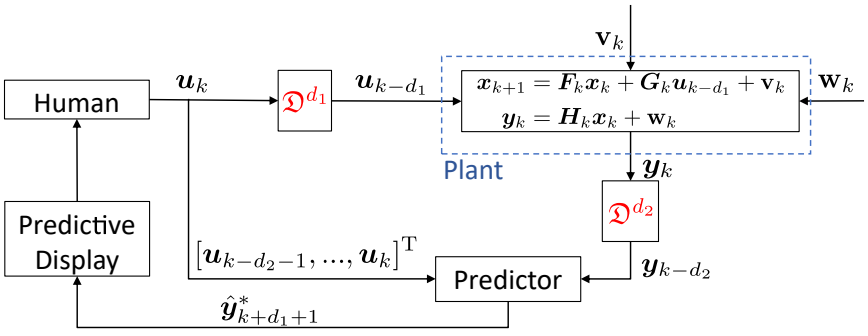


Fig. 2 A discrete-time representation of a telerobotic predictor system with input and output delays.

Figure 2 represents a discrete-time, linear, time-varying telerobotic system with pure input and output delays. Here and in the subsequent discussion, it is assumed that the system is sampled at constant intervals with sample period T_s. That is, inputs are applied and outputs are measured at times $t_k = kT_\mathrm{s}$ where $k \in \{0, 1, 2, \ldots\}$ is an integer time index. The input delay is $\tau_1 = d_1 T_\mathrm{s}$ and the output delay is $\tau_2 = d_2 T_\mathrm{s}$ for some non-negative integers d_1 and d_2. In Fig. 2, the delay operator $\mathfrak{D}^d$, also known as the shift operator, shifts the signal by d steps. A continuous-time signal $\boldsymbol{y}(t)$ that is delayed by time τ_1 is denoted $\boldsymbol{y}_\mathrm{del}(t) = \boldsymbol{y}(t - \tau_1)$. In discrete-time, this delay corresponds to a shift by d_1 time steps: $\boldsymbol{y}_\mathrm{del}(t_k) = \mathfrak{D}^{d_1}\boldsymbol{y}(t_k) = \boldsymbol{y}(t_{k-d_1})$. We use the more compact notation $\boldsymbol{y}_{\mathrm{del},k} = \mathfrak{D}^{d_1}\boldsymbol{y}_k = \boldsymbol{y}_{k-d_1}$ where $\boldsymbol{y}_k = \boldsymbol{y}(t_k)$.

For the linear discrete-time system

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{G}_k \boldsymbol{u}_{k-d_1} + \mathbf{v}_k \tag{1}$$

$$\boldsymbol{y}_k = \boldsymbol{H}_k \boldsymbol{x}_k + \mathbf{w}_k \tag{2}$$

shown in Fig. 2, $\boldsymbol{x}_{(\cdot)} \in \mathbb{R}^{n_x}$ is the state vector, $\boldsymbol{F}_{(\cdot)} \in \mathbb{R}^{n_x \times n_x}$ is the state matrix, $\boldsymbol{u}_{(\cdot)} \in \mathbb{R}^{n_u}$ is the input vector, $\boldsymbol{G}_{(\cdot)} \in \mathbb{R}^{n_x \times n_u}$ is the input matrix, $\boldsymbol{y}_{(\cdot)} \in \mathbb{R}^{n_y}$ is the output vector, $\boldsymbol{H}_{(\cdot)} \in \mathbb{R}^{n_y \times n_x}$ is the output matrix, $\mathbf{v}_{(\cdot)} \in \mathbb{R}^{n_x}$ is the process noise vector, and $\mathbf{w}_{(\cdot)} \in \mathbb{R}^{n_y}$ is the measurement noise vector.

In Eqn. (1), the plant is subjected to the delayed operator input $\boldsymbol{u}_{k-d_1}$ instead of the current input $\boldsymbol{u}_k$. Moreover, the human operator's knowledge of the system state is based on the delayed output $\boldsymbol{y}_{k-d_2} = \mathfrak{D}^{d_2} \boldsymbol{y}_k$ rather than the current output $\boldsymbol{y}_k$. *The goal of the predictor is to provide the operator with a predicted output at a time $d_1 + d_2 + 1$ steps ahead of the present time, using only the system model, the current output $\boldsymbol{y}_{k-d_2}$, and the input history $\boldsymbol{u}_{(\cdot)}$ from time $t_{k-d_1-d_2-1}$ to t_k.* This predicted output $\hat{\boldsymbol{y}}^*_{k+d_1+1}$ should capture the effect of the currently generated command $\boldsymbol{u}_k$ and be presented to the operator immediately. (In the motivating application of a predictive display, the information would be presented to the operator as an image, but more general systems are considered here.)

Note that

$$\begin{aligned} \boldsymbol{y}_{k-d_2} &= \mathfrak{D}^{d_2} \boldsymbol{y}_k \\ &= \mathfrak{D}^{d_2} [\boldsymbol{H}_k \boldsymbol{x}_k + \mathbf{w}_k] \\ &= \boldsymbol{H}_{k-d_2} \boldsymbol{x}_{k-d_2} + \mathbf{w}_{k-d_2} \\ &= \boldsymbol{H}_{k-d_2} [\boldsymbol{F}_{k-d_2-1} \boldsymbol{x}_{k-d_2-1} + \boldsymbol{G}_{k-d_2-1} \boldsymbol{u}_{k-d_1-d_2-1} + \mathbf{v}_{k-d_2-1}] + \mathbf{w}_{k-d_2} \end{aligned} \tag{3}$$

From Eqn. (3) it can be seen that the currently available measurement is the result of the input command $\boldsymbol{u}_{k-d_1-d_2-1}$. Noting that

$$\boldsymbol{y}_{k+d_1+1} = \boldsymbol{H}_{k+d_1+1} [\boldsymbol{F}_{k+d_1} \boldsymbol{x}_{k+d_1} + \boldsymbol{G}_{k+d_1} \boldsymbol{u}_k + \mathbf{v}_{k+d_1+1}] + \mathbf{w}_{k+d_1+1}$$

the current input command $\boldsymbol{u}_k$ affects the output $\boldsymbol{y}_{k+d_1+1}$. The job of the predictor is to provide an estimate of $\boldsymbol{y}_{k+d_1+1}$ immediately, despite the fact that the effect of the input $\boldsymbol{u}_k$ on the plant will be delayed (by d_1 time steps) and the observation of that effect will be delayed even further (by d_2 time steps).

The term $d_1 + d_2 + 1$ in the expressions above is called the "delay horizon." The delay horizon represents two important quantities: 1) the number of time steps the predictor must account for to mitigate the delay and 2) the number of past input values that must be stored for use by the predictor.

In formulating a prediction strategy to address the challenge above, the following assumptions are made:

- The input and output delays are known and constant.
- The model ($\{\boldsymbol{F}_{(\cdot)}, \boldsymbol{G}_{(\cdot)}, \boldsymbol{H}_{(\cdot)}\}$ is known.
- The system is (linearly) observable.
- The measurement and process noise sequences $\mathbf{v}_k$ and $\mathbf{w}_k$ are additive, Gaussian, purely random sequences with $\mathbb{E}[\mathbf{v}_k] = 0$, $\mathbb{E}[\mathbf{w}_k] = 0$, $\mathbb{E}[\mathbf{v}_k \mathbf{v}_k^{\mathrm{T}}] = \boldsymbol{V}_k$, and $\mathbb{E}[\mathbf{w}_k \mathbf{w}_k^{\mathrm{T}}] = \boldsymbol{W}_k$.

4 Derivation of the SP and the KP

In comparing the performance of the two predictors, we consider their ability to accurately predict fixed-wing aircraft motion generated by a 6-DoF flight dynamic model and to mitigate the effects of process and measurement noise. We begin by presenting the mathematical equations for the predictors.

4.1 Equations for the SP

A Smith predictor (SP) uses a linear system model to predict future outputs. In particular, for the scenario described here, the SP is designed to predict the output at time step $k + d_1 + 1$ due to the current input $\boldsymbol{u}_k$, whose effect will be delayed by d_1 steps. To do so, the SP subtracts a model-based estimate $\hat{\boldsymbol{y}}_{k-d_2}$ from the currently available (but output delayed) measurement $\boldsymbol{y}_{k-d_2}$ and adds a model-based estimate $\hat{\boldsymbol{y}}_{k+d_1+1}$ computed as if there were no delay:

$$\hat{\boldsymbol{y}}^*_{k+d_1+1} = \boldsymbol{y}_{k-d_2} + (\hat{\boldsymbol{y}}_{k+d_1+1} - \hat{\boldsymbol{y}}_{k-d_2}) \tag{4}$$

where, from Eqn. (3),

$$\hat{\boldsymbol{y}}_{k-d_2} = \boldsymbol{H}_{k-d_2}[\boldsymbol{F}_{k-d_2-1}\hat{\boldsymbol{x}}_{k-d_2-1} + \boldsymbol{G}_{k-d_2-1}\boldsymbol{u}_{k-d_1-d_2-1}] \tag{5}$$

$$\hat{\boldsymbol{y}}_{k+d_1+1} = \boldsymbol{H}_{k+d_1+1}[\boldsymbol{F}_{k+d_1}\hat{\boldsymbol{x}}_{k+d_1} + \boldsymbol{G}_{k+d_1}\boldsymbol{u}_k] \tag{6}$$

Note that Eqns. (5)-(6) do not account for process or measurement noise and they require accurate initial state estimates $\hat{\boldsymbol{x}}_{k-d_2-1}$ and $\hat{\boldsymbol{x}}_{k+d_1}$ as well as an accurate dynamic model.

4.2 Equations for the KP

The Kalman predictor (KP) is closely related to the Kalman filter (KF), which is the minimum variance state estimator for a discrete-time linear system subject to Gaussian, purely random process and measurement noise [43]. In describing the filter, the subscript "$i|k$" will be used to indicate a value at time step i obtained using information available through time step k. The KF algorithm proceeds in five steps, after defining the initial state estimate $\hat{\boldsymbol{x}}_{0|0}$ and the initial state estimate error covariance $\hat{\boldsymbol{P}}_{0|0}$. At time t_{k-1}, the measurement that is available to update the state estimate is

$$\boldsymbol{y}_{k-d_2-1} = \boldsymbol{H}_{k-d_2-1}\boldsymbol{x}_{k-d_2-1} + \mathbf{w}_{k-d_2-1}$$

To simplify notation, the shifted time index $l = k - d_2$ is introduced. The one-step state estimate prediction starting at time step $l - 1$ is

$$\hat{\boldsymbol{x}}_{l|l-1} = \boldsymbol{F}_{l-1}\hat{\boldsymbol{x}}_{l-1|l-1} + \boldsymbol{G}_{l-1}\boldsymbol{u}_{l-d_1-1} \tag{7}$$

The corresponding state estimate error covariance is

$$\boldsymbol{P}_{l|l-1} = \boldsymbol{F}_{l-1}\boldsymbol{P}_{l-1|l-1}\boldsymbol{F}^{\mathrm{T}}_{l-1} + \boldsymbol{V}_{l-1} \tag{8}$$

The filter gain, known as the Kalman gain, is

$$\boldsymbol{L}_l = \boldsymbol{P}_{l|l-1}\boldsymbol{H}^{\mathrm{T}}_l[\boldsymbol{H}_l\boldsymbol{P}_{l|l-1}\boldsymbol{H}^{\mathrm{T}}_l + \boldsymbol{W}_l]^{-1} \tag{9}$$

Finally, the measurement updated state and covariance estimates are

$$\hat{\boldsymbol{x}}_{l|l} = \hat{\boldsymbol{x}}_{l|l-1} + \boldsymbol{L}_l(\boldsymbol{y}_l - \hat{\boldsymbol{y}}_{l|l-1}) \qquad [\text{where } \hat{\boldsymbol{y}}_{l|l-1} = \boldsymbol{H}_l\hat{\boldsymbol{x}}_{l|l-1}] \tag{10}$$

$$\boldsymbol{P}_{l|l} = (\mathbb{I} - \boldsymbol{L}_l\boldsymbol{H}_l)\boldsymbol{P}_{l|l-1} \tag{11}$$

The Kalman predictor (KP) uses the measurement updated state and covariance estimates recursively to find the predicted state $\hat{\boldsymbol{x}}^*_{k+d_1+1|l}$, predicted covariance $\boldsymbol{P}^*_{k+d_1+1|l}$, and predicted output $\hat{\boldsymbol{y}}^*_{k+d_1+1|l}$. Taking

Eqn. (10), the recursive predictor can be obtained by repeatedly applying the relationship in Eqn. (7)

$$\begin{aligned}
\hat{\boldsymbol{x}}_{l+1|l} &= \boldsymbol{F}_l \hat{\boldsymbol{x}}_{l|l} + \boldsymbol{G}_l \boldsymbol{u}_{l-d_1} \\
\Rightarrow \hat{\boldsymbol{x}}_{l+2|l} &= \boldsymbol{F}_{l+1} \hat{\boldsymbol{x}}_{l+1|l} + \boldsymbol{G}_{l+1} \boldsymbol{u}_{l-d_1+1} \\
&\vdots \\
\Rightarrow \hat{\boldsymbol{x}}_{l+d_1+d_2+1|l} &= \boldsymbol{F}_{l+d_1+d_2} \hat{\boldsymbol{x}}_{l+d_1+d_2|l} + \boldsymbol{G}_{l+d_1+d_2} \boldsymbol{u}_{l-d_1+d_1+d_2} \\
\hat{\boldsymbol{x}}_{k+d_1+1|k-d_2} &= \boldsymbol{F}_{k+d_1} \hat{\boldsymbol{x}}_{k+d_1|k-d_2} + \boldsymbol{G}_{k+d_1} \boldsymbol{u}_k \qquad [\text{Substituting } l = k - d_2]
\end{aligned} \tag{12}$$

The recursive state prediction stops when the current input $\boldsymbol{u}_k$ appears. The relationships captured in Eqn. (12) can be re-written as

$$\hat{\boldsymbol{x}}_{l+i|l} = \boldsymbol{F}_{l+i-1} \hat{\boldsymbol{x}}_{l+i-1|l} + \boldsymbol{G}_{l+i-1} \boldsymbol{u}_{l-d_1+i-1} \qquad [\text{where } i \in \{1, 2, 3, \ldots, d_1 + d_2 + 1\}] \tag{13}$$

$$\Rightarrow \hat{\boldsymbol{x}}_{l+i|l} = \left({}^{\mathrm{L}}\prod_{j=0}^{i-1} \boldsymbol{F}_{l+j} \right) \hat{\boldsymbol{x}}_{l|l} + \sum_{m=0}^{i-1} \left({}^{\mathrm{L}}\prod_{j=m+1}^{i-1} \boldsymbol{F}_{l+j} \right) \boldsymbol{G}_{l+m} \boldsymbol{u}_{l-d_1+m} \tag{14}$$

where the superscript L preceding the product symbol $\prod$ denotes left multiplication. For example, ${}^{\mathrm{L}}\prod_{j=0}^{n} \boldsymbol{Q}_j = \boldsymbol{Q}_n \cdots \boldsymbol{Q}_1 \boldsymbol{Q}_0$. The predicted output corresponding to the predicted state is obtained from Eqn. (13):

$$\hat{\boldsymbol{y}}^*_{k+d_1+1|k-d_2} = \boldsymbol{H}_{k+d_1+1} \hat{\boldsymbol{x}}_{k+d_1+1|k-d_2} \tag{15}$$

One may also compute the error covariance associated with the predicted state. Taking Eqn. (11) and applying the relationship in Eqn. (8) gives

$$\begin{aligned}
\boldsymbol{P}_{l+1|l} &= \boldsymbol{F}_l \boldsymbol{P}_{l|l} \boldsymbol{F}_l^{\mathrm{T}} + \boldsymbol{V}_l \\
\Rightarrow \boldsymbol{P}_{l+2|l} &= \boldsymbol{F}_{l+1} \boldsymbol{P}_{l+1|l} \boldsymbol{F}_{l+1}^{\mathrm{T}} + \boldsymbol{V}_{l+1} \\
&\vdots \\
\Rightarrow \boldsymbol{P}_{l+d_1+d_2+1|l} &= \boldsymbol{F}_{l+d_1+d_2} \boldsymbol{P}_{l+d_1+d_2|l} \boldsymbol{F}_{l+d_1+d_2}^{\mathrm{T}} + \boldsymbol{V}_{l+d_1+d_2}
\end{aligned} \tag{16}$$

Equation. (16) can be expressed using recursion

$$\boldsymbol{P}_{l+i|l} = \boldsymbol{F}_{l+i-1} \boldsymbol{P}_{l+i-1|l} \boldsymbol{F}_{l+i-1}^{\mathrm{T}} + \boldsymbol{V}_{l+i-1} \qquad [\text{with } i \in \{1, 2, 3, \ldots, d_1 + d_2 + 1\}] \tag{17}$$

5 Fixed-wing Aircraft Model

A fixed-wing aircraft model is used to simulate flight data to implement and compare the SP and the KP algorithms. An Earth-fixed north-east-down (NED) frame defined by the orthonormal vectors $\{\boldsymbol{i}_{\mathrm{I}}, \boldsymbol{j}_{\mathrm{I}}, \boldsymbol{k}_{\mathrm{I}}\}$ is adopted as the frame of reference; it is assumed to be an inertial frame over the time and space scales considered here. The body-fixed reference frame with origin at the aircraft's center of gravity is described by the orthonormal vectors $\{\boldsymbol{i}_{\mathrm{B}}, \boldsymbol{j}_{\mathrm{B}}, \boldsymbol{k}_{\mathrm{B}}\}$ where the positive $\boldsymbol{i}_{\mathrm{B}}$ axis points out the nose of the aircraft, the positive $\boldsymbol{j}_{\mathrm{B}}$ axis points out of the right wing of the aircraft, and the positive $\boldsymbol{k}_{\mathrm{B}}$ axis points down through the underside of the aircraft. The assumptions employed in developing the aircraft equations of motion are: (i) Earth is flat and fixed in inertial space, (ii) the air is at rest, (iii) the aircraft is a rigid body, (iv) the air density and gravitational acceleration are constant, and (v) the aircraft mass m and inertia matrix $\boldsymbol{I} \in \mathbb{R}^{3\times3}$ are constant. The kinematic and dynamic equations for rigid aircraft motion

are then

$$\dot{\boldsymbol{s}} = \boldsymbol{R}_{\mathrm{IB}}(\boldsymbol{\Theta})\boldsymbol{v} \tag{18a}$$
$$\dot{\boldsymbol{\Theta}} = \boldsymbol{L}_{\mathrm{IB}}(\boldsymbol{\Theta})\boldsymbol{\omega} \tag{18b}$$
$$\dot{\boldsymbol{v}} = \boldsymbol{v}\times\boldsymbol{\omega} + \frac{1}{m}\boldsymbol{F}_{\mathrm{A}} + \frac{1}{m}\boldsymbol{R}_{\mathrm{IB}}^{\mathrm{T}}\boldsymbol{F}_{\mathrm{G}} \tag{18c}$$
$$\dot{\boldsymbol{\omega}} = \boldsymbol{I}^{-1}(\boldsymbol{I}\boldsymbol{\omega}\times\boldsymbol{\omega}) + \boldsymbol{I}^{-1}\boldsymbol{M}_{\mathrm{A}} \tag{18d}$$

where $\boldsymbol{s} = [x, y, z]^{\mathrm{T}} \in \mathbb{R}^3$ represents the NED position of the aircraft and $\boldsymbol{\Theta} = [\phi, \theta, \psi]^{\mathrm{T}} \in \mathbb{R}^3$ is the vector of roll-pitch-yaw Euler angles parameterizing the proper rotation matrix $\boldsymbol{R}_{\mathrm{IB}}$ that maps free vectors from the body frame to the inertial frame:

$$\boldsymbol{R}_{\mathrm{IB}}(\boldsymbol{\Theta}) = \begin{pmatrix} \cos\theta\cos\psi & \cos\psi\sin\theta\sin\phi - \cos\phi\sin\psi & \cos\psi\sin\theta\cos\phi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\theta\cos\phi\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{pmatrix} \tag{19}$$

The linear and angular velocity of the aircraft with respect to the NED frame, but expressed in the body-fixed frame, are $\boldsymbol{v} = [u, v, w]^{\mathrm{T}} \in \mathbb{R}^3$ and $\boldsymbol{\omega} = [p, q, r]^T \in \mathbb{R}^3$, respectively. The matrix

$$\boldsymbol{L}_{\mathrm{IB}}(\boldsymbol{\Theta}) = \begin{pmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{pmatrix} \tag{20}$$

relates the aircraft's body angular velocity to the Euler angle rates as shown in Eqn. (18b). The aircraft weight expressed in the NED frame is $\boldsymbol{F}_{\mathrm{G}} = [0, 0, mg]^{\mathrm{T}}$. The aerodynamic force and moment are denoted $\boldsymbol{F}_{\mathrm{A}}$ and $\boldsymbol{M}_{\mathrm{A}}$, respectively. Here, we consider the MTD2 aircraft shown in Fig. 3. We adopt the propulsion and aerodynamic models that were identified empirically as described in [44, 45] and used in [46, 47]. The aero-propulsive forces and moments are given by

$$\boldsymbol{F}_{\mathrm{A}} = \frac{1}{2}\rho\|\boldsymbol{v}\|^2 S \begin{pmatrix} C_X(\boldsymbol{v},\boldsymbol{\omega},\boldsymbol{\delta}) \\ C_Y(\boldsymbol{v},\boldsymbol{\omega},\boldsymbol{\delta}) \\ C_Z(\boldsymbol{v},\boldsymbol{\omega},\boldsymbol{\delta}) \end{pmatrix} + D^4\rho\eta_e\eta_{\mathrm{n}}\delta_{\mathrm{rps}}^2 \begin{pmatrix} C_J(\boldsymbol{\delta}) \\ 0 \\ 0 \end{pmatrix} \tag{21}$$

$$\boldsymbol{M}_{\mathrm{A}} = \frac{1}{2}\rho\|\boldsymbol{v}\|^2 S \begin{pmatrix} bC_l(\boldsymbol{v},\boldsymbol{\omega},\boldsymbol{\delta}) \\ \bar{c}C_m(\boldsymbol{v},\boldsymbol{\omega},\boldsymbol{\delta}) \\ bC_n(\boldsymbol{v},\boldsymbol{\omega},\boldsymbol{\delta}) \end{pmatrix} \tag{22}$$

where $\boldsymbol{\delta} = [\delta_{\mathrm{a}}, \delta_{\mathrm{e}}, \delta_{\mathrm{r}}, \delta_{\mathrm{rps}}]^{\mathrm{T}}$ are the control inputs describing aileron, elevator, rudder, and thrust commands, S is the aircraft wing surface area, $\bar{c}$ is the mean aerodynamic chord, b is the wingspan, ρ is the air density, D is the diameter of the propeller, η_e is the propeller efficiency, and η_{n} is the number of propellers. The non-dimensional thrust, force, and moment coefficients are

$$C_J = C_{J_0} + C_J J + C_{J^2} J^2 \tag{23a}$$
$$C_X = C_{X_0} + C_{X_{\delta_{\mathrm{e}}}}\delta_{\mathrm{e}} + C_{X_\alpha}\alpha + C_{X_\alpha^2}\alpha^2 \tag{23b}$$
$$C_Y = C_{Y_p}\hat{p} + C_{Y_{\mathrm{r}}}\hat{r} + C_{Y_{\delta_{\mathrm{a}}}}\delta_{\mathrm{a}} + C_{Y_{\delta_{\mathrm{r}}}}\delta_{\mathrm{r}} + C_{Y_\beta}\beta \tag{23c}$$
$$C_Z = C_{Z_0} + C_{Z_q}\hat{q} + C_{Z_\alpha}\alpha \tag{23d}$$
$$C_l = C_{l_p}\hat{p} + C_{l_{\delta_{\mathrm{a}}}}\delta_{\mathrm{a}} + C_{l_\beta}\beta \tag{23e}$$
$$C_m = C_{m_0} + C_{m_q}\hat{q} + C_{m_{\delta_{\mathrm{e}}}}\delta_{\mathrm{e}} + C_{m_\alpha}\alpha \tag{23f}$$
$$C_n = C_{n_{\mathrm{r}}}\hat{r} + C_{n_{\delta_{\mathrm{a}}}}\delta_{\mathrm{a}} + C_{n_{\delta_{\mathrm{r}}}}\delta_{\mathrm{r}} + C_{n_\beta}\beta \tag{23g}$$

where the non-dimensional terms in Eqn. (23) are

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \quad \beta = \sin^{-1}\left(\frac{v}{\|\boldsymbol{v}\|}\right) \quad \hat{p} = \frac{pb}{2\|\boldsymbol{v}\|} \quad \hat{q} = \frac{q\bar{c}}{2\|\boldsymbol{v}\|} \quad \hat{r} = \frac{rb}{2\|\boldsymbol{v}\|} \quad J = \frac{\|\boldsymbol{v}\|}{\delta_{\text{rps}} D}$$

The SP and the KP framework require linear system models. To apply these predictors to the aircraft motion model, the nonlinear system (18) is linearized about a nominal trajectory. The particular case of constant-speed, wings-level flight at a constant altitude is considered in this paper. Recognizing that the aircraft position $\boldsymbol{s}$ plays no role in the dynamic equations, we define a reduced-dimensional state vector $\boldsymbol{x} = [\boldsymbol{\Theta}^{\mathrm{T}}, \boldsymbol{v}^{\mathrm{T}}, \boldsymbol{\omega}^{\mathrm{T}}]^{\mathrm{T}}$. Fixing the value of the propeller speed δ_{rps}, we also define a reduced-dimensional input vector $\boldsymbol{u} = [\delta_{\text{a}}, \delta_{\text{e}}, \delta_{\text{r}}]^{\mathrm{T}}$. With these simplifications, we rewrite equations (18b-d) as

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) = \begin{pmatrix} \boldsymbol{L}_{\text{IB}}(\boldsymbol{\Theta})\boldsymbol{\omega} \\ \boldsymbol{v} \times \boldsymbol{\omega} + \boldsymbol{F}_{\text{A}} + \boldsymbol{R}_{\text{IB}}(\boldsymbol{\Theta})^{\mathrm{T}} \boldsymbol{F}_{\text{G}} \\ \boldsymbol{I}^{-1}\left(\boldsymbol{I}\boldsymbol{\omega} \times \boldsymbol{\omega}\right) + \boldsymbol{I}^{-1}\boldsymbol{M}_{\text{A}} \end{pmatrix} \tag{24}$$

To apply the SP and KP algorithms, the system (24) is linearized about the equilibrium flight condition $\{\boldsymbol{x}(t), \boldsymbol{u}(t)\} = \{\boldsymbol{x}_*, \boldsymbol{u}_*\}$ resulting in the linear, time-invariant system

$$\Delta\dot{\boldsymbol{x}} = \boldsymbol{A}\Delta\boldsymbol{x} + \boldsymbol{B}\Delta\boldsymbol{u} \tag{25}$$

where $\Delta\boldsymbol{x} = \boldsymbol{x} - \boldsymbol{x}_*$ and $\Delta\boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}_*$ and where

$$\boldsymbol{A} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}_*, \boldsymbol{u}=\boldsymbol{u}_*} \quad \text{and} \quad \boldsymbol{B} = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}}\right|_{\boldsymbol{x}=\boldsymbol{x}_*, \boldsymbol{u}=\boldsymbol{u}_*} \tag{26}$$

The linear system (25) is valid for small perturbations from the nominal condition.

The LTI system given above is a continuous-time system. To obtain the discrete-time system matrices $\boldsymbol{F}_k$ and $\boldsymbol{G}_k$ in Eqn. (1), one must discretize the continuous time system, *e.g.*, by assuming a zero-order hold on the inputs and computing the sampled data solution. To obtain the position $\boldsymbol{s}$ of the aircraft in the inertial frame, the translational kinematic equation (18a) is integrated in parallel with the predictor equations based on the reduced-dimensional model (24). The position update is

$$\hat{\boldsymbol{s}}_{k+i+1} = \hat{\boldsymbol{s}}_{k+i} + \boldsymbol{R}_{\text{IB}}(\hat{\boldsymbol{\Theta}}_{k+i})\hat{\boldsymbol{v}}_{k+i} T_{\text{s}} \tag{27}$$

where $i = 1, 2, 3, \ldots, d_1 + d_2 + 1$.

Table 1 Fixed-wing UAV parameters [44].

Parameter	Symbol	Value
Mass	m	3.311 kg
Moments of inertia	I_{xx}	0.319 kg-m^2
	I_{yy}	0.267 kg-m^2
	I_{zz}	0.471 kg-m^2
	I_{xz}	0.024 kg-m^2
Wing span	b	1.8 m
Wing surface area	S	0.457 m^2
Mean aerodynamic chord	$\bar{c}$	0.254 m

Fig. 3 The MTD2 aircraft.

6 Simulation Setup

The properties of the MTD2 aircraft used to obtain simulated flight data are given in Table 1. Note that the cross products of inertia I_{xy} and I_{yz} are zero for a symmetric aircraft. Numerical values of the aerodynamic force and moment coefficients from Eqn. (23) are provided in Table 2. The aircraft was simulated along the steady trajectory for which $\boldsymbol{x}_* = [0, 0.0210, 0, 18.1568, 0, 0.3808, 0, 0, 0]^{\mathrm{T}}$ and $\boldsymbol{u}_* = [0, -0.0045, 0]^{\mathrm{T}}$ (where all values are given in SI units), which was obtained by solving the nonlinear subsystem in Eqn. (24) for a wings-level, constant altitude, equilibrium flight condition. The aircraft generated constant thrust with the propeller rotation rate held constant at $\delta_{\mathrm{rps}} = 220$ rad/s throughout the simulation. The propeller diameter was $D = 0.254$ m, the propeller efficiency was set to $\eta_e = 0.9$, and two propellers were used: $\eta_{\mathrm{n}} = 2$.

Table 2 Aerodynamic force and moment coefficients for the MTD2.

Coefficient	Value	Coefficient	Value	Coefficient	Value
C_{J_0}	-0.131	C_{Y_p}	0.221	C_{Z_0}	-0.225
C_J	-0.040	C_{Y_r}	0.230	C_{Z_q}	-12.54
C_{J^2}	0.116	$C_{Y_{\delta_{\mathrm{a}}}}$	0.118	C_{Z_α}	-4.451
C_{X_0}	-0.428	$C_{Y_{\delta_{\mathrm{r}}}}$	0.136		
$C_{X_{\delta_{\mathrm{e}}}}$	0.051	C_{Y_β}	-0.525		
C_{X_α}	0.282				
$C_{X_{\alpha^2}}$	3.292				
Coefficient	Value	Coefficient	Value	Coefficient	Value
C_{l_p}	-0.386	C_{m_0}	0.008	C_{n_r}	-0.119
$C_{l_{\delta_{\mathrm{a}}}}$	-0.137	C_{m_q}	-14.02	$C_{n_{\delta_{\mathrm{a}}}}$	0.013
C_{l_β}	-0.039	$C_{m_{\delta_{\mathrm{e}}}}$	-0.415	$C_{n_{\delta_{\mathrm{r}}}}$	-0.068
		C_{m_α}	-0.471	C_{n_β}	0.103

The process and measurement noise covariance matrices for the KP were chosen as $\boldsymbol{V}_k = 0.001 \cdot \mathbb{I}$ and $\boldsymbol{W}_k = \mathrm{diag}([0.01, 0.01, 0.01, 0.1, 0.1, 0.1, 0.01, 0.01, 0.01])$ where, again, the values are given in SI units. The initial states of the SP and KP were chosen to be the steady motion of the aircraft: $\hat{\boldsymbol{x}}_{k-d_{20}} = \boldsymbol{x}_*$. The initial position estimate of the aircraft was $\hat{\boldsymbol{s}}_0 = [0, 0, -100]^{\mathrm{T}}$. For the SP, the initial predicted state $\hat{\boldsymbol{x}}_{k+d_{10}}$ was obtained by recursively propagating $\hat{\boldsymbol{x}}_{k-d_{20}}$ using the linear model in Eqn. (25). The simulation was carried out with standard sea level density and with no wind.

To analyze the performance of the predictors, simulation studies were done for the following test conditions:

- Two different perturbations from the nominal wings-level equilibrium flight condition: elevator doublets of amplitude (a) 2° and (b) 16°
- Two different noise covariances: (i) $\boldsymbol{W}_k$ and (ii) $10\boldsymbol{W}_k$

The 2° elevator doublet is in the range of a typical small perturbation from equilibrium, resulting in a motion well described by a linearized dynamic model. This maneuver is used to analyze the accuracy of the SP and the KP under small perturbations. The more aggressive 16° elevator doublet takes the aircraft well away from equilibrium and is used to gauge how well the linear predictors like the SP and the KP perform in predicting the nonlinear behavior of the aircraft. The input and the output delays used for both of the cases were $\tau_1 = \tau_2 = 1$ s. The sampling time was chosen to be $T_{\mathrm{s}} = 0.1$ s. Figure 4 shows the commanded input histories as well as the trajectory of the aircraft due to the input. Figure 4a shows

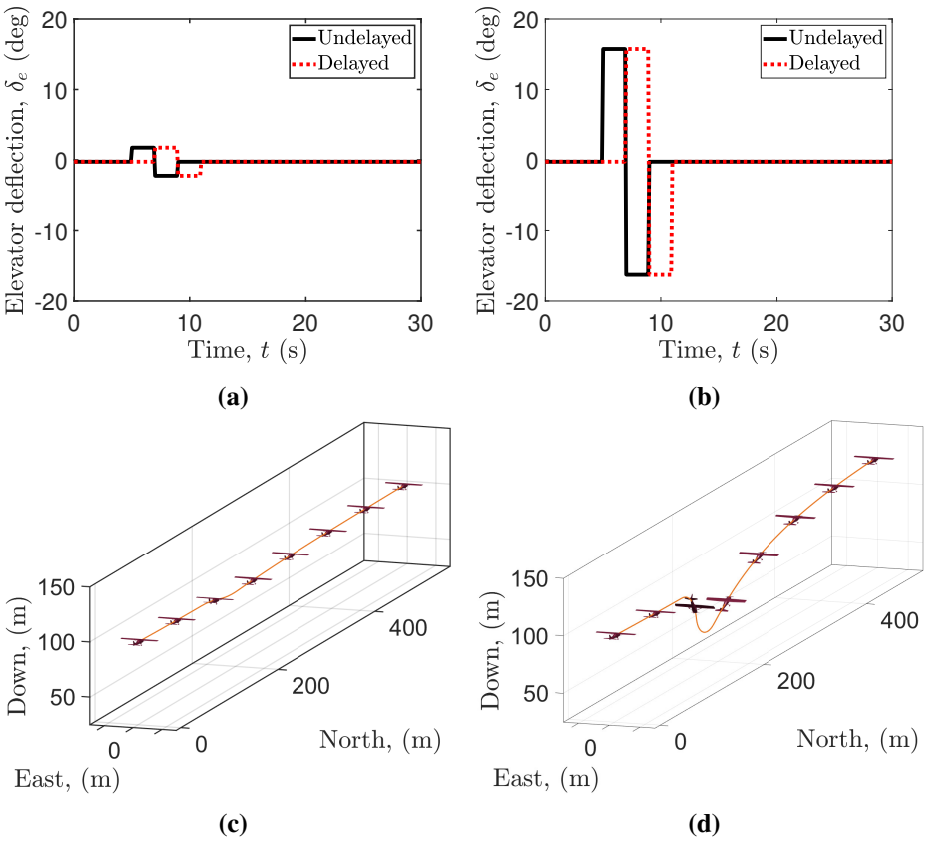


Fig. 4 Commanded (undelayed) and realized (delayed) elevator input histories (top) and aircraft position histories (bottom) for two doublet amplitudes with noise covariance W_k.

the execution of an elevator doublet of amplitude 2° and duration 2 s. The solid black line indicates the doublet commanded by the human operator starting at $t = 5$ s. The dashed red line indicates the delayed doublet that affects the system after the input delay starting at $t = 6$ s. Figure 4b is similar to Fig. 4a in all aspects except for the amplitude. The trajectories generated due to these commanded inputs are depicted in Figs. 4c and 4d. Both the trajectories start from the same initial state but Fig. 4d shows a steeper descent than Fig. 4c as the doublet amplitude is higher.

7 Results and Discussions

Figure 5 shows the aircraft's northward position under the action of the 16° elevator input with different types of delays. The black trajectory labeled "Undelayed" is the trajectory that is obtained by setting both delays to zero: $\tau_1 = \tau_2 = 0$ s. The red trajectory labeled "Input delay" is the trajectory generated when the input is delayed but the output is obtained instantly, that is, $\tau_1 = 1$ s and $\tau_2 = 0$ s. This is the actual output of the system under the action of delayed input commands, but this output is not perceived by the operator when there is an output delay. The blue trajectory is the red trajectory delayed by $\tau_2 = 1$ s. It is the delayed measurement that is available to the predictor (or that an operator would observe in the absence of predictive compensation). The function of the predictor is to eliminate these time delays in the telerobotic feedback loop which is why the predicted outputs from the two predictors are compared with the undelayed output of the system.

For this simulation study, we consider full state measurement: $\boldsymbol{y} = \boldsymbol{x}$. Since the command applied to the aircraft is an elevator doublet, however, only the longitudinal dynamics of the aircraft are affected. The only state variables affected by this command are x, z, θ, u, w, q. Thus, in evaluating the performance of the two predictors, only these longitudinal states variables are considered. Figure 6 shows the trajectory

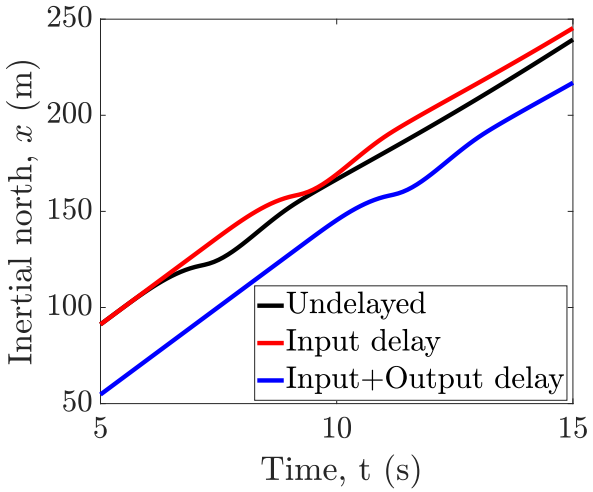


Fig. 5 Effect of different delays on the aircraft's north position.

of the aircraft under the influence of the doublet command over a 30 s window. The black trajectory in each of the subplots of Fig. 6 is the "undelayed measurement" of the system which is how the system would have behaved if there were no delays present ($\tau_1 = \tau_2 = 0$). The dashed red line is the output from the SP and the dashed blue line is the output from the KP. Inspecting Fig. 6 reveals that the KP performs better than the SP in predicting the large perturbation motion of the delayed system, adhering more closely to the fictitious undelayed output. The KP also filters noise better than the SP.

To assess the performance of the predictors qualitatively, the root mean square error (RMSE) between the predicted output and the noise-free undelayed output was computed and averaged over ten simulation runs for each of the four test cases: {(a), (b), (i), (ii)}. The total difference between the predicted output and the undelayed output at some time t_k was computed and averaged over the total number of data points across all ten simulation runs for each test case. The square root of the value gives the RMSE for that state. The process was repeated for the two different predictors for all six longitudinal outputs across all four test cases. The complete results can be obtained from the `GitHub` repository [48] which contains the simulation code and the figures.

The resulting RMSEs are shown in the bar charts presented in Fig. 7. Each subfigure contains the average RMSE obtained for two particular state variables (indicated by the labels) from the ten simulation runs of the two predictors with two different doublet inputs: (a) 2° and (b) 16°. The upper and lower rowscorrespond to the two measurement noise covariance matrices (i) $\boldsymbol{W}_k$ and (ii) $10\boldsymbol{W}_k$, respectively. The units for these RMSE results are standard SI units: x, z in m, u, w in m/s, θ in °, and q in °/s. Key observations from these simulation studies are summarized below.

- **Effect of doublet amplitude:** As the amplitude of the doublet increases from 2° to 16°, the linear predictors become inaccurate, as indicated by the increased RMSE; see Fig. 7. When the doublet amplitude is small, the KP performs slightly better than the SP as indicated by the RMSE values in Fig. 7. The SP has a very high RMSE in the x and z states as the Euler integration using Eqn. (27) builds up error due to incorrect prediction and initial states. This drift can be observed in the x and z states in Fig. 6.
- **Effect of noise:** As the measurement noise covariance increases from $\boldsymbol{W}_k$ to $10\boldsymbol{W}_k$, the RMSE for both predictors increases; compare the first row of Fig. 7 with the corresponding second row to observe the change. The RMSE values across all six states and doublet commands increase as the noise factor increases. The KP accounts for noise, as the prediction algorithm is built on the KF algorithm. Comparing the first and second rows of Fig. 7, one sees that the RMSE values for the KP generally do not increase as dramatically as they do for the SP when measurement noise covariance is increased.

Summarizing the results from the simulation studies discussed above, we note that:

- The SP performs worse than the KP, often substantially worse, in all of the test variables and test cases indicating that it is not a suitable predictor for the aerial telerobotic application considered here. For lower doublet amplitudes, the KP performs well, but as the amplitude increases the performance degrades.
- The performance of both predictors suffers when the noise covariance is increased. However, the incorporation of noise covariance in the KF algorithm helps the KP mitigate some effects of noise.

The RMSE data verify that predicting large perturbation motions of a nonlinear system like a rigid fixed-wing aircraft using linear predictors can produce large errors. The KP is effective for small perturbations from the nominal flight condition, but the SP performs poorly even in this case. To accommodate larger perturbations, the KP can be extended to include nonlinearities, as is done in the extended Kalman filter. The development of this extended Kalman predictor (EKP) is described in [49], with performance comparisons to the KP and SP presented here.

8 Conclusions

A general Kalman predictor (KP) has been developed in this paper to mitigate input and output time delays in telerobotic systems. The KP was applied to a nonlinear fixed-wing aircraft simulation model and compared with the classical Smith predictor (SP). The KP is better than the SP at predicting aircraft motion and is more robust to noise. However, the performance of the KP deteriorates as the amplitude of the input perturbations increases to cause larger perturbations from the nominal state of motion. In these cases, a nonlinear predictor such as the extended Kalman predictor described in [49] should be used. Other possible improvements include modifying the KP (or extended KP) framework to accommodate varying time delays and sampling time intervals.

Appendix

The appendix contains the graphs of the results for the different maneuvers as well as the model used for simulation.

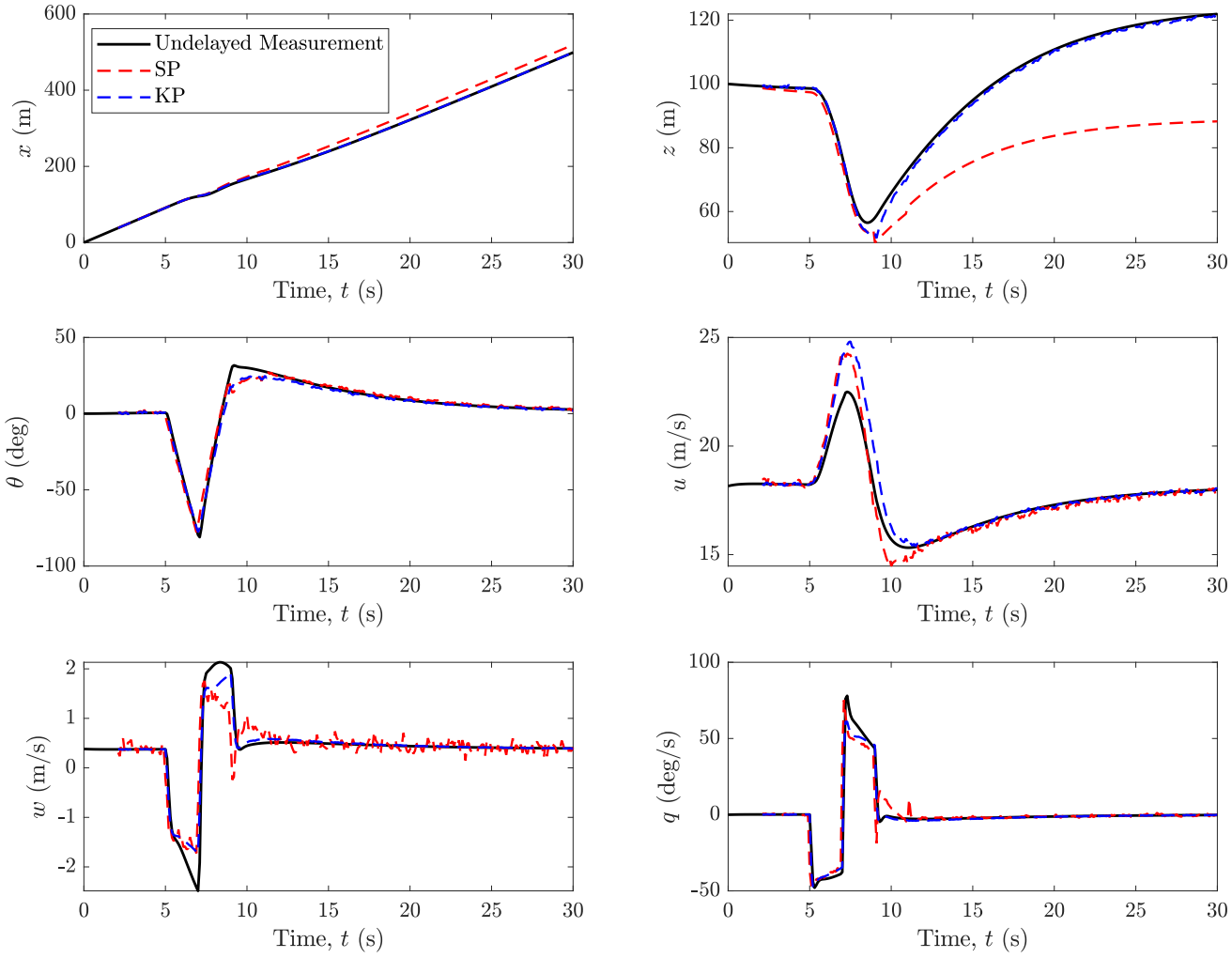
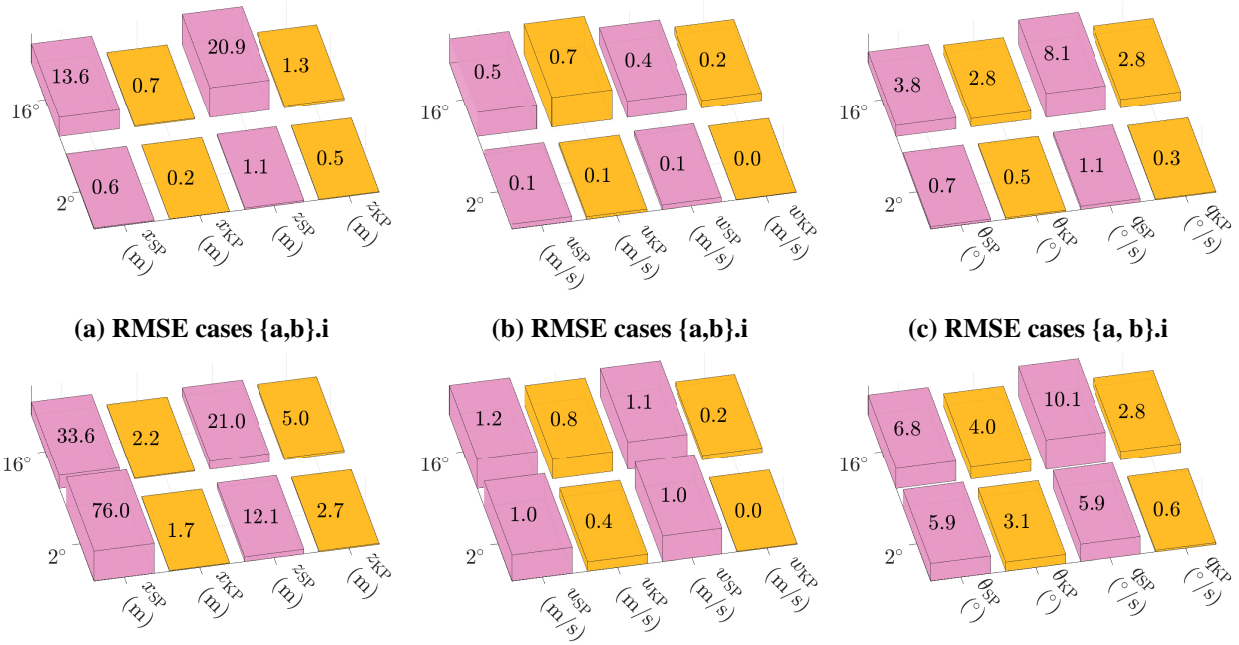


Fig. 6 Output of the two predictors compared to the undelayed output for the case with sampling time, $T_s = 0.1$ s, total delay $\tau_1 + \tau_2 = 2$ s, noise covariance W_k, and doublet amplitude of 16°.

Using the steady-state parameters $\{\boldsymbol{x}_*, \boldsymbol{u}_*\}$ given in Section 6, the nonlinear subsystem in Eqn. (24) was linearized and, subsequently, discretized using the zero-order hold method. It resulted in the following discrete-time system matrices used in the simulation:

$$\boldsymbol{F}_k = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0.0210 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0002 \\ 0 & -9.8078 & 0 & -1.4716 & 0 & 0.6209 & 0 & -0.3813 & 0 \\ 9.8078 & 0 & 0 & 0 & -0.6358 & 0 & 0.6898 & 0 & -17.8358 \\ 0 & -0.2060 & 0 & -0.9356 & 0 & -6.9233 & 0 & 15.6872 & 0 \\ 0 & 0 & 0 & 0 & -0.9816 & 0 & -10.1034 & 0 & -0.1587 \\ 0 & 0 & 0 & 0.0481 & 0 & -2.3000 & 0 & -8.6962 & 0 \\ 0 & 0 & 0 & 0 & 1.9710 & 0 & -0.5148 & 0 & -2.1096 \end{bmatrix}$$

$$\boldsymbol{G}_k = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1.4363 & 0 \\ 3.3231 & 0 & 3.8300 \\ 0 & 0 & 0 \\ & -72.0089 & 0 \quad -1.8301 \\ 0 & -36.8122 & 0 \\ 0.9633 & 0 & -24.3248 \end{bmatrix} \quad \text{and} \quad \boldsymbol{H}_k = \mathbb{I}_{9\times 9}$$



(a) RMSE cases {a,b}.i **(b) RMSE cases {a,b}.i** **(c) RMSE cases {a, b}.i**

(d) RMSE cases {a,b}.ii **(e) RMSE cases {a,b}.ii** **(f) RMSE cases {a,b}.ii**

Fig. 7 Average RMSE values of different predictors with total delay of 2 s, sampling time of 0.1 s. The captions indicate the doublet amplitudes ({a,b}) and the measurement covariance ({i,ii}) used. For example, a bar chart labeled "RMSE cases {a,b}.ii" shows the RMSE (averaged over ten simulations) for 2° doublets ("a") and 16° doublets ("b") with measurement noise covariance $10W_k$ ("ii"). The specific state variables and predictor formulations for the given chart are indicated in its x-axis labels.

References

[1] T. B. Sheridan. Telerobotics. *Automatica*, 25(4):487–507, 1989. ISSN: 0005-1098. DOI: 10.1016/0005-1098(89)90093-9.

[2] T. B. Sheridan. Space Teleoperation Through Time Delay: Review and Prognosis. *IEEE Transactions on Robotics and Automation*, 9(5):592–606, 1993. DOI: 10.1109/70.258052.

[3] J. Y. C. Chen. UAV-Guided Navigation for Ground Robot Teleoperation in a Military Reconnaissance Environment. *Ergonomics*, 53(8):940–950, 2010. DOI: 10.1080/00140139.2010.500404.

[4] R. Saltaren, R. Aracil, C. Alvarez, E. Yime, and J. M. Sabater. Field and Service Applications–Exploring Deep Sea by Teleoperated Robot–An Underwater Parallel Robot with High Navigation Capabilities. *IEEE Robotics & Automation Magazine*, 14(3):65–75, 2007. DOI: 10.1109/MRA.2007.905502.

[5] D. W. Hainsworth. Teleoperation User Interfaces for Mining Robotics. *Autonomous Robots*, 11:19–28, 07 2001. DOI: 10.1023/A:1011299910904.

[6] R. C. Goertz and W. M. Thompson. Electronically Controlled Mainipulator. *Nucleonics (U.S.) Ceased publication*, 12(11), 11 1954. URL: https://www.osti.gov/biblio/4387399.

[7] S. Halder and K. Afsari. Real-Time Construction Inspection in an Immersive Environment with an Inspector Assistant Robot. *Proceedings of the 58th Annual ASC International Conference, Atlanta, GA*, 2022. DOI: 10.29007/ck81.

[8] K. Afsari, S. Halder, M. Ensafi, S. DeVito, and J. Serdakowski. Fundamentals and Prospects of Four-Legged Robot Application in Construction Progress Monitoring. *EPiC Series in Built Environment*, 2:274–283, 2021. DOI: 10.29007/cdpd.

[9] R. D. Madder, S. VanOosterhout, A. Mulder, J. Bush, S. Martin, A. J. Rash, J. M. Tan, J. L. Parker, A. Kalafut, Y. Li, et al. Network Latency and Long-Distance Robotic Telestenting: Exploring the Potential Impact of Network Delays on Telestenting Performance. *Catheterization and Cardiovascular Interventions*, 95(5):914–919, 2020.

[10] J. Y. C. Chen, E. C. Haas, and M. J. Barnes. Human Performance Issues and User Interface Design for Teleoperated Robots. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 37(6):1231–1245, 2007. DOI: 10.1109/tsmcc.2007.905819.

[11] T. B. Sheridan and W. R. Ferrell. Remote Manipulative Control with Transmission Delay. *IEEE Transactions on Human Factors in Electronics*, HFE-4(1):25–29, 1963. DOI: 10.1109/THFE.1963.231283.

[12] M. J. Brudnak. Predictive Displays for High Latency Teleoperation. In *NDIA Ground Vehicle Systems Engineering Technology Symposium*, Sterling Heights, MI, USA, 2016. National Defense Industrial Association (NDIA), Michigan Chapter. URL: https://www.researchgate.net/publication/305904651_PREDICTIVE_DISPLAYS_FOR_HIGH_LATENCY_TELEOPERATION.

[13] Y. Jung, K. Han, and J. Bae. A Tele-Operated Display With a Predictive Display Algorithm. *IEEE Access*, 7:154447–154456, 2019. DOI: 10.1109/ACCESS.2019.2948879.

[14] J. Cox and K. C. Wong. Predictive Feedback Augmentation for Manual Control of an Unmanned Aerial Vehicle with Latency. *International Journal of Micro Air Vehicles*, 11:1756829319869645, 2019. DOI: 10.1177/1756829319869645.

[15] N. Sakib, J. L. Gresham, and C. A. Woolsey. Usability Studies of a Predictive Heterogeneous Vision System in Mitigating the Effects of Visual Display Delay. In *AIAA SciTech 2021 Forum*, 2021. DOI: 10.2514/6.2021-0017.

[16] N. Sakib, K. C. Gahan, and C. A. Woolsey. Time Delay Mitigation in Aerial Telerobotic Operations Using Heterogeneous Stereo-Vision Systems. *Journal of Aerospace Information Systems*, 20(9):526–545, 2023. DOI: 10.2514/1.I011204.

[17] C. Kang, H. Chaudhry, C. A. Woolsey, and K. Kochersberger. Development of a Peripheral-Central Vision System for Small UAS Tracking. In *AIAA SciTech 2019 Forum*, January 2019. DOI: 10.2514/6.2019-2074.

[18] E. L. Bianchi, N. Sakib, C. A. Woolsey, and M. Hebdon. Bridge Inspection Component Registration for Damage Evolution. *Structural Health Monitoring*, 0(0):14759217221083647, 2022. DOI: 10.1177/14759217221083647.

[19] O. J. Smith. Closer Control of Loops With Dead Time. *Chemical Engineering Progress*, 53:217 – 219, 1957.

[20] A. Ingimundarson and T. Hägglund. Performance comparison between pid and dead-time compensating controllers. *Journal of Process Control*, 12(8):887–895, 2002. ISSN: 0959-1524. DOI: 10.1016/S0959-1524(02)00017-3.

[21] J. E. Normey-Rico, C. Bordons, M. Berenguel, and E. F. Camacho. A Robust Adaptive Dead-Time Compensator with Application to A Solar Collector Field. *IFAC Proceedings Volumes*, 31(19):93–98, 1998. ISSN: 1474-6670. DOI: 10.1016/S1474-6670(17)41134-7.

[22] D. Limón-Marruedo, J. E. Normey-Rico, D. J. Pagano, and J. Aracil. Stability of Saturated Dead-Time Compensating PI Controllers for Uncertain Dead-Time Systems. In *1999 European Control Conference (ECC)*, pages 2251–2256, 1999. DOI: 10.23919/ECC.1999.7099655.

[23] I. Kaya. Obtaining Controller Parameters for a new PI-PD Smith Predictor Using Autotuning. *Journal of Process Control*, 13(5):465–472, 2003. ISSN: 0959-1524. DOI: 10.1016/S0959-1524(02)00086-0.

[24] T. Hägglund. An Industrial Dead-Time Compensating PI Controller. *Control Engineering Practice*, 4(6):749–756, 1996. ISSN: 0967-0661. DOI: 10.1016/0967-0661(96)00065-2.

[25] J. E. Normey-Rico, C. Bordons, and E. F. Camacho. Improving the Robustness of Dead-Time Compensating PI Controllers. *Control Engineering Practice*, 5(6):801–810, 1997. ISSN: 0967-0661. DOI: 10.1016/S0967-0661(97)00064-6.

[26] J. E. Normey-Rico and E. F. Camacho. Unified Approach for Robust Dead-Time Compensator Design. *Journal of Process Control*, 19(1):38–47, 2009. ISSN: 0959-1524. DOI: 10.1016/j.jprocont.2008.02.003.

[27] C. Santacesaria and R. Scattolini. Easy Tuning of Smith Predictor in Presence of Delay Uncertainty. *Automatica*, 29(6):1595–1597, 1993. ISSN: 0005-1098. DOI: 10.1016/0005-1098(93)90027-Q.

[28] M. R. Matausek and A. D. Micic. On the Modified Smith Predictor for Controlling a Process with an Integrator and Long Dead-Time. *IEEE Transactions on Automatic Control*, 44(8):1603–1606, 1999. DOI: 10.1109/9.780433.

[29] A. Bahill. A Simple Adaptive Smith-Predictor for Controlling Time-Delay Systems: A Tutorial. *IEEE Control Systems Magazine*, 3(2):16–22, 1983. DOI: 10.1109/MCS.1983.1104748.

[30] D. J. Pagano, J. E. Normey-Rico, and A. L. D. Franco. Stability Analysis of a Modified Smith Predictor for Integrative Plants with Dead-Time Uncertainties and Saturations. In *Proceedings of the 40th IEEE Conference on Decision and Control*, volume 2, pages 1855–1860, 2001. DOI: 10.1109/CDC.2001.981175.

[31] C. Kravaris and R. A. Wright. Deadtime Compensation for Nonlinear Processes. *AIChE Journal*, 35(9):1535–1542, 1989. DOI: 10.1002/aic.690350914.

[32] X. Chen and A. Serrani. Smith Predictors in Nonlinear Systems – Application to ISS-based Leader/Follower Trailing Control. In *2007 American Control Conference*, pages 4506–4511, 2007. DOI: 10.1109/ACC.2007.4283107.

[33] J. E. Normey-Rico, J. Gómez-Ortega, and E. F. Camacho. A Smith-predictor-Based Generalised Predictive Controller for Mobile Robot Path-Tracking. *Control Engineering Practice*, 7(6):729–740, 1999. ISSN: 0967-0661. DOI: 10.1016/S0967-0661(99)00025-8.

[34] J. Prakash, M. Vignati, E. Sabbioni, and F. Cheli. Vehicle Teleoperation: Human in the Loop Performance Comparison of Smith Predictor with Novel Successive Reference-Pose Tracking Approach. *Sensors*, 22(23), 2022. DOI: 10.3390/s22239119.

[35] H. Chen and Z. Liu. Time-Delay Prediction–Based Smith Predictive Control for Space Teleoperation. *Journal of Guidance, Control, and Dynamics*, 44(4):872–879, 2021. DOI: 10.2514/1.G005714.

[36] C. Thomas, S. Yi, S. Meadows, and R. E. Sherrill. Adaptive Smith Predictor for Teleoperation of UAVs with Time-Varying Internet Delay. *International Journal of Control, Automation and Systems*, 18:1465–1473, 2020. DOI: 10.1007/s12555-019-0121-6.

[37] T. G. Molnar, D. Hajdu, and T. Insperger. Chapter 10 – The Smith Predictor, the Modified Smith Predictor, and the Finite Spectrum Assignment: A Comparative Study. In *Stability, Control and Application of Time-delay Systems*, pages 209–226. Butterworth-Heinemann, 2019. ISBN: 978-0-12-814928-7. DOI: 10.1016/B978-0-12-814928-7.00010-X.

[38] C. F. Moore. *Selected Problems in the Design and Implementation of Direct Digital Control*. Louisiana State University and Agricultural & Mechanical College, 1969.

[39] S. K. P. Wong and D. E. Seborg. A theoretical Analysis of Smith and Analytical Predictors. *AIChE Journal*, 32(10):1597–1605, 1986. DOI: 10.1002/aic.690321003.

[40] J. Ordaz, S. Salazar, S. Mondié, H. Romero, and R. Lozano. Predictor-Based Position Control of a Quad-Rotor with Delays in GPS and Vision Measurements. *Journal of Intelligent Robotics Systems*, 70(1–4):13–26, apr 2013. ISSN: 0921-0296. DOI: 10.1007/s10846-012-9714-5.

[41] B. Zhu, M. Chen, and T. Li. State Prediction Based Control Schemes for Nonlinear Systems with Input Delay and External Disturbance. *IET Control Theory & Applications*, 15(13):1697–1707, 2021. DOI: 10.1049/cth2.12152.

[42] D. M. Lima, B. M. Lima, and J. E. Normey-Rico. A Predictor for Dead-Time Systems Based on the Kalman Filter for Improved Disturbance Rejection and Robustness. *Journal of Process Control*, 105:108–116, 2021. ISSN: 0959-1524. DOI: 10.1016/j.jprocont.2021.07.011.

[43] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., USA, 2002. ISBN: 0471221279.

[44] M. H. Halefom, J. W. Hopwood, and C. A. Woolsey. Unsteady Aerodynamics in Model-Based Wind Estimation from Small, Fixed-Wing Aircraft Motion. *Journal of Guidance, Control, and Dynamics*, 2024. To appear.

[45] J. L. Gresham, M. H. Halefom, J. W. Hopwood, Z. Ahmed, and C. A. Woolsey. MTD2 Research Aircraft Flight Dynamic Model. GitHub, 2024. https://github.com/NSL-VT/Flight-Dynamic-Models/tree/master/MTD2.

[46] M. H. Halefom, J. L. Gresham, and C. A. Woolsey. Wind Estimation from an Unsteady Aerodynamic Aircraft Motion Model. In *AIAA SciTech 2022 Forum*, 2022. DOI: 10.2514/6.2022-0554.

[47] Z. Ahmed, M. H. Halefom, and C. A. Woolsey. A Tutorial Review of Indirect Wind Estimation Methods Using Small Uncrewed Air Vehicle. *Journal of Aerospace Information Systems*, 2024. To appear.

[48] N. Sakib and Z. Ahmed. Predictor-Simulations. GitHub, 2024. https://github.com/Nazmus20/Predictor-Simulations/tree/EGNC2024.

[49] Nazmus Sakib. *Time Delay Mitigation in Aerial Telerobotic Operations Using Predictors and Predictive Displays*. PhD thesis, Virginia Polytechnic Institute and State University, 2024.