Estimation-free Prediction Algorithms

NICHOLAS ASSIMAKIS[1], MARIA ADAM[2], CHRISTOS TSINOS[1], ATHANASIOS POLYZOS[1]
[1]Department of Digital Industry Technologies,
National and Kapodistrian University of Athens,
34400 Psachna Evias,
GREECE

[2]Department of Computer Science and Biomedical Informatics,
University of Thessaly,
2-4 Papasiopoulou Str., 35131, Lamia,
GREECE

Abstract: - For Time-varying, Time-invariant, and steady-state systems, Kalman Filter can be implemented as a prediction algorithm, since it produces the state prediction and the corresponding prediction error covariance matrix via the state estimation and the corresponding estimation error covariance matrix. Lainiotis Filter is equivalent to Kalman Filter and can be used to compute the prediction. In this paper, for Time-varying, Time-invariant and steady state systems, estimation-free Prediction Algorithms are derived via Kalman and Lainiotis filters; they are equivalent and compute iteratively the prediction and the corresponding prediction error covariance matrix. The estimation and the corresponding estimation error covariance matrix are not needed and are not computed. The proposed estimation-free prediction algorithms are faster than the Kalman filter.

Key-Words: - Kalman filter, Lainiotis Filter, Time-varying System, Time-invariant System, Estimation, Prediction.

Received: March 23, 2023. Revised: December 5, 2023. Accepted: December 23, 2023. Published: December 31, 2023.

1 Introduction

Prediction and estimation play an important role in many fields of science: applications to the aerospace industry, chemical process, communication systems design, control, civil engineering, filtering noise from 2-dimensional images, pollution prediction, and power systems are mentioned in [1]. The estimation problem arises in linear estimation and is associated with discrete-time systems described by the following state space equations:

$$\begin{Bmatrix} x(k+1) = F(k+1)x(k) + w(k) \\ z(k) = H(k)x(k) + v(k) \end{Bmatrix} \tag{1}$$

where $x(k)$ is the $n \times 1$ state vector, $z(k)$ is the $m \times 1$ measurement vector, $F(k+1)$ is the $n \times n$ transition matrix, $H(k)$ is the $m \times n$ output matrix, $w(k)$ is the $n \times 1$ state noise and $v(k)$ is the $m \times 1$ measurement noise at time $k \geq 0$.

The statistical model expresses the nature of the state and the measurements. The basic assumption is that the state noise $\{w(k)\}$ and the measurement noise $\{v(k)\}$ are white noises, i.e. a stochastic process with uncorrelated successive values: $\{w(k)\}$ is a zero mean, Gaussian process with known covariance $Q(k)$ of dimension $n \times n$ and $\{v(k)\}$ is a zero mean, Gaussian process with known covariance $R(k)$ of dimension $m \times m$. The following assumptions also hold: (a) the initial value of the state $x(0)$ is a Gaussian random variable with mean x_0 and covariance P_0; (b) the stochastic processes $\{w(k)\}$, $\{v(k)\}$ and the random variable $x(0)$ are independent.

The discrete-time Kalman filter, [1] and Lainiotis filter, [2] are well-known algorithms that solve the filtering problem, producing the state estimation $x(k/k)$ and the corresponding estimation error covariance matrix $P(k/k)$. The filters can be Time-Varying(TV), Time-invariant (TI) or Steady State (SS). Kalman filter can be seen as a prediction algorithm as well, because it produces the state prediction $x(k+1/k)$ and the corresponding prediction error covariance matrix $P(k+1/k)$.

The importance of filtering algorithms is without doubt: Kalman filter has been used in electric load estimation, [3], power generation prediction, [4], weather forecasts, [5], cases and deaths prediction of Covid-19, [6], satellite orbit determination, [7], multi-observation fusion applications related to timescale, [8] which is widely used in satellite navigation, [9].

In this paper estimation-free Prediction Algorithms are derived via Kalman and Lainiotis filters, for Time-varying, Time-invariant, and steady state cases. The proposed algorithms can be applied in many applications that require computation of prediction: short-term electric load forecasting, [10], weather prediction, [11], prediction of air pollution levels, [12], stock price prediction, [13], [14], prediction of the control effectiveness of the actuator on behalf of an actuator stuck fault incident occurring on airplanes, [15], Kalman filter prediction that accounts for measurement differences, for the case of time-correlated measurement errors, [16], Global Positioning System (GPS) and Inertial Navigation System (INS) integration during GPS outages using machine learning augmented with Kalman filter, [17].

The paper is organized as follows: Time-varying, Time-invariant and steady state Kalman and Lainiotis filters are summarized in section 2. Time-varying, Time-invariant, and steady state estimation-free Prediction Algorithms are derived via Kalman filter in section 3. Time-varying, Time-invariant and steady state estimation-free Prediction Algorithms are derived via Lainiotis filter in section 4. It is established that the Kalman filter and the Lainiotis filter based prediction algorithms are equivalent concerning their behavior, since they produce the same predictions. In section 5 the FIR form of the steady state estimation-free prediction algorithms is presented. In section 6 the multiple steps prediction algorithms are derived. The computational requirements of estimation-free prediction algorithms are determined in section 7. It is shown that the estimation-free prediction algorithms are faster than Kalman filter. Finally, Section 8 summarizes the conclusions.

2 Kalman and Lainiotis Filters

Time-varying, Time-invariant and steady state Kalman and Lainiotis filters are summarized in this section.

2.1 Kalman Filter

Kalman filter produces the state estimation and the estimation error covariance, as well as the state prediction and the corresponding prediction error covariance matrix.

For Time-varying systems, the Time-varying Kalman Filter is derived:

Time-varying Kalman Filter (TVKF)

$$\begin{aligned}
&x(k+1/k) = F(k+1)x(k/k)\\
&P(k+1/k) = Q(k) + F(k+1)P(k/k)F^T(k+1)\\
&K(k+1) = P(k+1/k)H^T(k+1)\\
&\qquad [H(k+1)P(k+1/k)H^T(k+1) + R(k+1)]^{-1}\\
&x(k+1/k+1) = [I - K(k+1)H(k+1)]x(k+1/k)\\
&\qquad\qquad +K(k+1)z(k+1)\\
&P(k+1/k+1) = [I - K(k+1)H(k+1)]P(k+1/k)
\end{aligned} \tag{2}$$

for $k = 0,1,...$, with initial conditions $x(0/0)$, $P(0/0)$.

The notation M^T is used for the transpose matrix of matrix M.

The notation I is used for the identity matrix.

Note that these initial conditions are connected to classically used initial conditions, [2]
$x(0/-1) = x_0$, $P(0/-1) = P_0$ through the following equations:
$x(0/0) = [I - K(0)H(0)]x_0 + K(0)z(0)$
$P(0/0) = [I - K(0)H(0)]P_0$
$K(0) = P_0H^T(0)[H(0)P_0H^T(0) + R(0)]^{-1}$

The choice of these initial conditions is due to reasons of uniformity concerning all algorithms of this paper.

For Time-invariant systems, where the transition matrix $\mathrm{F} = \mathrm{F(k+1)}$, the output matrix $\mathrm{H} = \mathrm{H(k)}$, as well as the plant and measurement noise covariance matrices $\mathrm{Q} = \mathrm{Q(k)}$ and $\mathrm{R} = \mathrm{R(k)}$ are constant matrices, the Time-invariant Kalman Filter is derived:

Time-invariant Kalman Filter (TIKF)

$$\begin{aligned}
&x(k+1/k) = Fx(k/k)\\
&P(k+1/k) = Q + FP(k/k)F^T\\
&K(k+1) = P(k+1/k)H^T[HP(k+1/k)H^T + R]^{-1}\\
&x(k+1/k+1) = [I - K(k+1)H]x(k+1/k)\\
&\qquad\qquad +K(k+1)z(k+1)\\
&P(k+1/k+1) = [I - K(k+1)H]P(k+1/k)
\end{aligned} \tag{3}$$

for $k = 0,1,...$, with initial conditions $x(0/0)$, $P(0/0)$

Note that these initial conditions are connected to classically used initial conditions, [2]
$x(0/-1) = x_0$, $P(0/-1) = P_0$ through the following equations:
$x(0/0) = [I - K(0)H]x_0 + K(0)z(0)$
$P(0/0) = [I - K(0)H]P_0$
$K(0) = P_0H^T[HP_0H^T + R]^{-1}$

For Time-invariant systems, it is well known, [1], that if the signal process model is asymptotically stable, then there exists a steady state value P_p of the prediction error covariance matrix. This value remains constant after the steady state time is reached. The Steady State Kalman Filter is derived:

Steady State Kalman Filter (SSKF)

$$x(k+1/k+1) = A_{KF}x(k/k) + B_{KF}z(k+1) \quad (4)$$

for $k = 0,1,...$, with initial condition $x(0/0)$

where

$$\left\{\begin{matrix} A_{KF} = [I - KH]F \\ B_{KF} = K \end{matrix}\right\} \quad (5)$$

and

$$K = P_pH^T\left[HP_pH^T + R\right]^{-1} \quad (6)$$

is the steady state Kalman Filter gain
and P_p is the solution of the Riccati equation

$$P_p = Q + FP_pF^T - FP_pH^T\left[HP_pH^T + R\right]^{-1}HP_pF^T \quad (7)$$

The steady state coefficients in (5) are calculated offline by first solving the corresponding discrete-time Riccati equation emanating from the Kalman filter, [1].

Steady State Kalman Filter can be seen as a prediction algorithm as well by computing the prediction $x(k+1/k) = Fx(k/k)$.

Note that this initial condition is connected to classically used initial conditions, [2]
$x(0/-1) = x_0$, $P(0/-1) = P_0$ through the following equations:

$$x(0/0) = [I - K(0)H]x_0 + K(0)z(0)$$
$$K(0) = P_0H^T[HP_0H^T + R]^{-1}$$

2.2 Lainiotis Filter

Lainiotis filter produces the state estimation and the estimation error covariance. It can be used to compute the state prediction and the prediction error covariance, using Kalman filter equations.

For Time-varying systems, the Time-varying Lainiotis Filter is derived:

Time-varying Lainiotis Filter (TVLF)

$$\begin{aligned} x(k+1/k+1) &= K_n(k+1)z(k+1) \\ &\quad + F_n(k+1)[I + P(k/k)O_n(k+1)]^{-1} \\ &\quad [P(k/k)K_m(k+1)z(k+1) + x(k/k)] \\ P(k+1/k+1) &= P_n(k+1) \\ &\quad + F_n(k+1)[I + P(k/k)O_n(k+1)]^{-1} \\ &\quad P(k/k)F_n^T(k+1) \end{aligned} \quad (8)$$

for $k = 0,1,...$, with initial conditions $x(0/0)$, $P(0/0)$,

where

$$\begin{aligned} L(k+1) &= [H(k+1)Q(k)H^T(k+1) + R(k+1)]^{-1} \\ K_n(k+1) &= Q(k)H^T(k+1)L(k+1) \\ K_m(k+1) &= F^T(k+1)H^T(k+1)L(k+1) \\ P_n(k+1) &= [I - K_n(k+1)H(k+1)]Q(k) \\ F_n(k+1) &= [I - K_n(k+1)H(k+1)]F(k+1) \\ O_n(k+1) &= F^T(k+1)H^T(k+1) \\ &\quad L(k+1)H(k+1)F(k+1) \end{aligned} \quad (9)$$

Note that these initial conditions are connected to classically used initial conditions, [2]
$x(0/-1) = x_0$, $P(0/-1) = P_0$ through the following equations:

$$x(0/0) = [I - K(0)H(0)]x_0 + K(0)z(0)$$
$$P(0/0) = [I - K(0)H(0)]P_0$$
$$K(0) = P_0H^T(0)[H(0)P_0H^T(0) + R(0)]^{-1}$$

The choice of these initial conditions is due to reasons of uniformity concerning all algorithms of this paper.

Time-varying Kalman and Lainiotis filters are equivalent with respect to their behavior, since they produce the same estimations and the same estimation error covariance matrices, [2].

Time-varying Lainiotis Filter can be used to compute the state prediction and the prediction error covariance, using Kalman filter equations.

$$x(k+1/k) = F(k+1)x(k/k)$$
$$P(k+1/k) = Q(k) + F(k+1)P(k/k)\,F^T(k+1)$$

Remark 1.
Time-varying Kalman and Lainiotis filters have the same structure:

From (2) we get:

$$\left\{\begin{matrix} x(k+1/k+1) = A_{KF}(k+1)x(k/k) \\ \quad + B_{KF}(k+1)z(k+1) \\ A_{KF}(k+1) = [I - K(k+1)H(k+1)]F(k+1) \\ B_{KF}(k+1) = K(k+1) \end{matrix}\right\} \quad (10)$$

while from (8) we get:

$$\left\{\begin{matrix} x(k+1/k+1) = A_{LF}(k+1)x(k/k) \\ \quad + B_{LF}(k+1)z(k+1) \\ A_{LF}(k+1) = F_n(k+1)[I + P(k/k)O_n(k+1)]^{-1} \\ B_{LF}(k+1) = K_n(k+1) \\ + F_n(k+1)[I + P(k/k)O_n(k+1)]^{-1}P(k/k)K_m(k+1) \end{matrix}\right\} \quad (11)$$

Due to the fact that the two filters are equivalent, we obtain:

$$\left\{\begin{matrix} A_{KF}(k+1) = A_{LF}(k+1) \\ B_{KF}(k+1) = B_{LF}(k+1) \end{matrix}\right\} \quad (12)$$

For Time-invariant systems, where the transition matrix $F = F(k+1)$, the output matrix $H = H(k)$, as well as the plant and measurement noise covariance matrices $Q = Q(k)$ and $R = R(k)$ are constant matrices, the Time-invariant Lainiotis Filter is derived:

Time-invariant Lainiotis Filter (TILF)

$$\begin{aligned} x(k+1/k+1) &= K_n z(k+1) + F_n[I + P(k/k)O_n]^{-1} \\ &\quad [P(k/k)K_m z(k+1) + x(k/k)] \\ P(k+1/k+1) &= P_n \\ &\quad + F_n[I + P(k/k)O_n]^{-1}P(k/k)F_n^T \end{aligned} \quad (13)$$

for $k = 0,1, \ldots$
with initial conditions $x(0/0) = x_0$, $P(0/0) = P_0$.

where

$$\begin{aligned} L &= [HQH^T + R]^{-1} \\ K_n &= QH^T L \\ K_m &= F^T H^T L \\ P_n &= [I - K_n H]Q \\ F_n &= [I - K_n H]F \\ O_n &= F^T H^T LHF \end{aligned} \quad (14)$$

Obviously, the constant matrices in (14) are computed off-line.

Note that these initial conditions are connected to classically used initial conditions, [2] $x(0/-1) = x_0$, $P(0/-1) = P_0$ through the following equations:

$x(0/0) = [I - K(0)H]x_0 + K(0)z(0)$
$P(0/0) = [I - K(0)H]P_0$
$K(0) = P_0 H^T[HP_0 H^T R]^{-1}$

Time-invariant Kalman and Lainiotis filters are equivalent concerning their behavior, since they produce the same estimations and the same estimation error covariance matrices, [2].

Time-invariant Lainiotis Filter can be used to compute the state prediction and the prediction error covariance, using Kalman filter equations.

$x(k+1/k) = Fx(k/k)$
$P(k+1/k) = Q + FP(k/k)\,F^T$

Remark 2.
Time-invariant Kalman and Lainiotis filters have the same structure.

From (3) we get:

$$\left\{ \begin{aligned} x(k+1/k+1) &= A_{KF}(k+1)x(k/k) \\ &\quad + B_{KF}(k+1)z(k+1) \\ A_{KF}(k+1) &= [I - K(k+1)H]F \\ B_{KF}(k+1) &= K(k+1) \end{aligned} \right\} \quad (15)$$

while from (13) we get:

$$\left\{ \begin{aligned} x(k+1/k+1) &= A_{LF}(k+1)x(k/k) \\ &\quad + B_{LF}(k+1)z(k+1) \\ A_{LF}(k+1) &= F_n[I + P(k/k)O_n]^{-1} \\ B_{LF}(k+1) &= K_n + \\ &\quad + F_n[I + P(k/k)O_n]^{-1}P(k/k)K_m \end{aligned} \right\} \quad (16)$$

Because the two filters are equivalent, we obtain:

$$\left\{ \begin{aligned} A_{KF}(k+1) &= A_{LF}(k+1) \\ B_{KF}(k+1) &= B_{LF}(k+1) \end{aligned} \right\} \quad (17)$$

For Time-invariant systems, it is well known, [1], that if the signal process model is asymptotically stable, then there exists a steady state value P_e of the estimation error covariance matrix. This value remains constant after the steady state time is reached. The Steady State Lainiotis Filter is derived:

Steady State Lainiotis Filter (SSLF)

$$x(k+1/k+1) = A_{LF}x(k/k) + B_{LF}z(k+1) \quad (18)$$

for $k = 0,1, \ldots$, with initial condition $x(0/0) = x_0$,

where

$$\left\{ \begin{aligned} A_{LF} &= F_n[I + P_e O_n]^{-1} \\ B_{LF} &= K_n + F_n[I + P_e O_n]^{-1}P_e K_m \end{aligned} \right\} \quad (19)$$

and P_e is the solution of the Riccati equation

$$P_e = P_n + F_n[I + P_e O_n]^{-1}P_e F_n^T \quad (20)$$

The steady state coefficients in (19) are calculated off-line by first solving the corresponding discrete-time Riccati equation emanating from the Lainiotis filter, [18].

Steady State Lainiotis filter can be seen as a prediction algorithm as well by computing the prediction $x(k+1/k) = Fx(k/k)$. The steady state prediction error covariance can be computed by the steady state estimation error covariance:

$P_p = Q + FP_e\,F^T$

Note that this initial condition is connected to classically used initial conditions, [2]
$x(0/-1) = x_0$, $P(0/-1) = P_0$ through the following equations:

$x(0/0) = [I - K(0)H]x_0 + K(0)z(0)$
$K(0) = P_0 H^T[HP_0 H^T R]^{-1}$

The Steady State Kalman Filter (SSKF) and the Steady State Lainiotis Filter (SSLF) are equivalent since they produce the same estimations, [2].

Remark 3.
Steady State Kalman and Lainiotis filters have the same structure.

Due to the fact that the two filters are equivalent, we obtain, [2]:

$$\begin{Bmatrix} A_{KF} = A_{LF} \\ B_{KF} = B_{LF} \end{Bmatrix} \quad (21)$$

3 Estimation-free Prediction Algorithms via Kalman Filter

Estimation-free Prediction Algorithms are derived via Kalman filter for Time-varying, Time-invariant and steady state systems. The prediction and the corresponding prediction error covariance are computed iteratively; the estimation and the corresponding estimation error covariance are not needed and are not computed.

3.1 Time-varying Prediction Algorithm via KF

The Time-varying Prediction Algorithm via KF is derived from Time-varying Kalman Filter equations.

Time-varying Prediction Algorithm via KF (TVPAKF)

$$\begin{aligned} K(k) &= P(k/k-1)H^T(k) \\ &\quad [H(k)P(k/k-1)H^T(k) + R(k)]^{-1} \\ D_{KF}(k) &= F(k+1)K(k) \\ C_{KF}(k) &= F(k+1) - D_{KF}(k)H(k) \\ P(k+1/k) &= Q(k) \\ &\quad + C_{KF}(k)P(k/k-1)\, F^T(k+1) \\ x(k+1/k) &= C_{KF}(k)x(k/k-1) + D_{KF}(k)z(k) \end{aligned} \quad (22)$$

for $k = 0,1, \ldots$, with initial conditions $x(0/-1) = x_0$, $P(0/-1) = P_0$.

Proof.

From Time-varying Kalman Filter equations we can write the Kalman Filter gain as:

$$K(k) = P(k/k-1)H^T(k) \\ [H(k)P(k/k-1)H^T(k) + R(k)]^{-1}$$

Concerning the prediction, from (2) we have:

$$\begin{aligned} x(k+1/k) &= F(k+1)x(k/k) \\ &= F(k+1) \\ &\quad \{[I - K(k)H(k)]x(k/k-1) + K(k)z(k)\} \\ &= F(k+1)[I - K(k)H(k)]x(k/k-1) \\ &\quad + F(k+1,k)K(k)z(k) \\ &= [F(k+1) - F(k+1)K(k)H(k)]x(k/k-1) \\ &\quad + [F(k+1)K(k)]z(k) \end{aligned}$$

Setting

$$D_{KF}(k) = F(k+1)K(k)$$
$$C_{KF}(k) = F(k+1) - D_{KF}(k)H(k)$$

we get

$$x(k+1/k) = C_{KF}(k)x(k/k-1) + D_{KF}(k)z(k)$$

Concerning the prediction error covariance, from (2) we have:

$$\begin{aligned} P(k+1/k) &= Q(k) + F(k+1)P(k/k)\, F^T(k+1) \\ &= Q(k) + F(k+1) \\ &\quad [I - K(k)H(k)]P(k/k-1)\, F^T(k+1) \\ &= Q(k) + F(k+1)P(k/k-1)\, F^T(k+1) \\ &\quad - F(k+1)K(k)H(k)P(k/k-1)\, F^T(k+1) \\ &= Q(k) + [F(k+1) - F(k+1)K(k)H(k)] \\ &\quad P(k/k-1)\, F^T(k+1) \\ &= Q(k) + C_{KF}(k)P(k/k-1)\, F^T(k+1) \end{aligned}$$

3.2 Time-invariant Prediction Algorithm via KF

For Time-invariant systems, where the transition matrix $F = F(k+1)$, the output matrix $H = H(k)$, as well as the plant and measurement noise covariance matrices $Q = Q(k)$ and $R = R(k)$ are constant matrices, the Time-invariant Prediction Algorithm via KF is derived.

Time-invariant Prediction Algorithm via KF (TIPAKF)

$$\begin{aligned} K(k) &= P(k/k-1)H^T[HP(k/k-1)H^T + R]^{-1} \\ D_{KF}(k) &= FK(k) \\ C_{KF}(k) &= F - D_{KF}(k)H(k) \\ P(k+1/k) &= Q + C_{KF}(k)P(k/k-1)\, F^T \\ x(k+1/k) &= C_{KF}(k)x(k/k-1) + D_{KF}(k)z(k) \end{aligned} \quad (23)$$

for $k = 0,1, \ldots$, with initial conditions $x(0/-1) = x_0$, $P(0/-1) = P_0$.

3.3 Steady State Prediction Algorithm via KF

For Time-invariant systems, it is well known, [1], that if the signal process model is asymptotically stable, then there exists a steady state value P_p of the prediction error covariance matrix. This value remains constant after the steady state time is reached. The Steady State Prediction Algorithm via KF is derived.

Steady State Prediction Algorithm via KF (SSPAKF)

$$x(k+1/k) = C_{KF}(k)x(k/k-1) + D_{KF}(k)z(k) \quad (24)$$

for $k = 0,1, \ldots$, with initial condition
$x(0/-1)) = x_0$

where

$$\begin{Bmatrix} D_{KF} = FK \\ C_{KF} = F - D_{KF}H \end{Bmatrix} \tag{25}$$

and

$K = P_p H^T \left[H P_p H^T + R \right]^{-1}$ is the steady state Kalman Filter gain and P_p is the solution of the Riccati equation:

$$P_p = Q + F P_p F^T - F P_p H^T \left[H P_p H^T + R \right]^{-1} H P_p F^T.$$

The steady state coefficients in (25) are calculated off-line by first solving the corresponding discrete time Riccati equation emanating from Kalman filter [1].

4 Estimation-free Prediction Algorithms via Lainiotis Filter

Estimation-free Prediction Algorithms are derived via Lainiotis filter for Time-varying, Time-invariant and steady state systems. The prediction and the corresponding prediction error covariance are computed iteratively; the estimation and the corresponding estimation error covariance are not needed and are not computed.

4.1 Time-varying Prediction Algorithm via LF

The Time-varying Prediction Algorithm via LF is derived from Time-varying Lainiotis Filter equations.

Time-varying Prediction Algorithm via LF (TVPALF)

$$\begin{aligned}
&C_{LF}(k) = F(k+1)F_n(k+1) \\
&\quad [I + F^{-1}(k)[P(k/k-1) - Q(k-1)]F^{-T}(k)O_n(k)]^{-1}F^{-1}(k) \\
&D_{LF}(k) = K_n(k) + F(k+1)F_n(k+1) \\
&\quad [I + F^{-1}(k)[P(k/k-1) - Q(k-1)]F^{-T}(k)O_n(k)]^{-1}K_m(k) \\
&P(k+1/k) = Q(k) + C_{LF}(k)P(k/k-1)\,F^T(k+1) \\
&x(k+1/k) = C_{LF}(k)x(k/k-1) + D_{LF}(k)z(k)
\end{aligned} \tag{26}$$

for $k = 0,1,\ldots$, with initial conditions $x(0/-1) = x_0, P(0/-1) = P_0$.

Proof.

Concerning the prediction, due to the fact that Kalman and Lainiotis filters are equivalent, [2], from Time-varying Kalman Filter equations we have:

$x(k+1/k) = F(k+1)x(k/k) \Rightarrow x(k/k) = F^{-1}(k+1)x(k+1/k)$

with the assumption that the matrices $F(k+1,k)$ are nonsingular.

Then from (8) we have:

$$\begin{aligned}
&x(k/k) = A_{LF}(k)x(k/k-1) + B_{LF}(k)z(k) \\
&\Rightarrow F^{-1}(k+1)x(k+1/k) \\
&\qquad = A_{LF}(k)F^{-1}(k+1)x(k/k-1) \\
&\qquad + B_{LF}(k)z(k) \\
&\Rightarrow x(k+1/k) = \\
&\qquad F(k+1)A_{LF}(k)F^{-1}(k+1)x(k/k-1) \\
&\qquad + F(k+1)B_{LF}(k)z(k)
\end{aligned}$$

Thus

$$x(k+1/k) = C_{LF}(k)x(k/k-1) + D_{LF}(k)z(k)$$

where

$$\begin{aligned}
&C_{LF}(k) = F(k+1)A_{LF}(k)F^{-1}(k+1) \\
&= F(k+1)F_n(k+1) \\
&\qquad [I + P(k-1/k-1)O_n(k)]^{-1}F^{-1}(k) \\
&= F(k+1)F_n(k+1) \\
&\qquad [P^{-1}(k-1/k-1) + O_n(k)]^{-1} \\
&\qquad P^{-1}(k-1/k-1)F^{-1}(k) \\
&D_{LF}(k) = K_n(k) + F(k+1)B_{LF}(k) \\
&= F(k+1)F_n(k+1) \\
&\qquad [I + P(k-1/k-1)O_n(k)]^{-1} \\
&\qquad P(k-1/k-1)K_m(k) \\
&= K_n(k) + F(k+1)F_n(k+1) \\
&[P^{-1}(k-1/k-1) + O_n(k)]^{-1}K_m(k)
\end{aligned}$$

and

$$\begin{aligned}
&P(k/k-1) = Q(k-1) \\
&\qquad + F(k)P(k-1/k-1)F^T(k) \\
&\Rightarrow P(k-1/k-1) \\
&= F^{-1}(k)[P(k/k-1) - Q(k-1)]F^{-T}(k)
\end{aligned}$$

Thus

$$C_{LF}(k) = F(k+1)F_n(k+1) \begin{bmatrix} I + F^{-1}(k) \\ [P(k/k-1) - Q(k-1)] \\ F^{-T}(k)O_n(k) \end{bmatrix}^{-1} F^{-1}(k)$$

$$D_{LF}(k) = K_n(k) + F(k+1)F_n(k+1) \begin{bmatrix} I + F^{-1}(k) \\ [P(k/k-1) - Q(k-1)] \\ F^{-T}(k)O_n(k) \end{bmatrix}^{-1} K_m(k)$$

Furthermore, recall (12) and hence:

$$\begin{aligned}
&x(k+1/k) = C_{LF}(k)x(k/k-1) + D_{LF}(k)z(k) \\
&= F(k+1)A_{LF}(k)F^{-1}(k+1)x(k/k-1) \\
&\qquad + F(k+1)B_{LF}(k)z(k) \\
&= F(k+1)A_{KF}(k)F^{-1}(k+1)x(k/k-1) \\
&\qquad + F(k+1)B_{KF}(k)z(k) \\
&= F(k+1)[I - K(k)H(k)] \\
&\qquad F(k)F^{-1}(k+1)x(k/k-1) \\
&\qquad + F(k+1)K(k)z(k) \\
&= F(k+1)[I - K(k)H(k)]x(k/k-1) \\
&\qquad + F(k+1)K(k)z(k)
\end{aligned}$$

$= C_{KF}(k)x(k/k-1) + D_{KF}(k)z(k)$

Concerning the prediction error covariance, from (8) we have:

$$\begin{aligned} P(k+1/k+1) &= P_n(k+1) \\ &+ F_n(k+1)[I + P(k/k)O_n(k+1)]^{-1} P(k/k)F_n^T(k+1) \\ &= P_n(k+1) + F_n(k+1) [P^{-1}(k/k) + O_n(k+1)]^{-1} P(k/k)F_n^T(k+1) \end{aligned}$$

From (2) we have:

$$\begin{aligned} P(k+1/k) &= Q(k) + F(k+1)P(k/k)\,F^T(k+1) \\ &= Q(k) + F(k+1) [P^{-1}(k/k-1) + H^T(k)R^{-1}(k)H(k)]^{-1}\,F^T(k+1) \\ &= Q(k) + F(k+1) P(k/k-1) - P(k/k-1)H^T(k) [H(k)P(k/k-1)H^T(k) + R(k)]^{-1}H(k)P(k/k-1) F^T(k+1) \\ &= Q(k) + F(k+1) \{[I - K(k)H(k)]P(k/k-1)\}\,F^T(k+1) \\ &= Q(k) + F(k+1)\,F^{-1}(k+1) C_{KF}(k)P(k/k-1)F^T(k+1) \\ &= Q(k) + C_{KF}(k)P(k/k-1)F^T(k+1) \end{aligned}$$

But

$$\begin{aligned} C_{KF}(k) &= F(k+1)A_{KF}(k)F^{-1}(k+1) \\ &= F(k+1)A_{LF}(k)F^{-1}(k+1) = C_{LF}(k) \\ D_{KF}(k) &= F(k+1)B_{KF} = F(k+1)B_{LF} = D_{LF}(k) \end{aligned}$$

Then

$$P(k+1/k) = Q(k) + C_{LF}(k)P(k/k-1)\,F^T(k+1)$$

Remark 4.
Time-varying Prediction Algorithm via KF and Time-varying Prediction Algorithm via LF have the same structure:

From (22) we get:
$x(k+1/k) = C_{KF}(k)x(k/k-1) + D_{KF}(k)z(k)$

while from (26) we get:
$x(k+1/k) = C_{LF}(k)x(k/k-1) + D_{LF}(k)z(k)$

Due to the fact that the two filters are equivalent, we obtain:

$$\left\{ \begin{array}{l} C_{KF}(k+1) = C_{LF}(k+1) \\ D_{KF}(k+1) = D_{LF}(k+1) \end{array} \right\} \quad (27)$$

Remark 5.
The following relations between estimation and prediction coefficients hold:

$$\left\{ \begin{array}{l} D_{KF}(k) = F(k+1)B_{KF}(k) \\ C_{KF}(k) = F(k+1)A_{KF}(k+1)F^{-1}(k+1) \end{array} \right\} \quad (28)$$

4.2 Time-invariant Prediction Algorithm via LF

For Time-invariant systems, where the transition matrix $F = F(k+1)$, the output matrix $H = H(k)$, as well as the plant and measurement noise covariance matrices $Q = Q(k)$ and $R = R(k)$ are constant matrices, the Time-invariant Prediction Algorithm via LF is derived.

Time-invariant Prediction Algorithm via LF (TIPALF)

$$\boxed{\begin{array}{l} C_{LF}(k) = FF_n\ \ [I + F^{-1}[P(k/k-1) - Q]F^{-T}O_n\,]^{-1}F^{-1} \\ D_{LF}(k) = K_n + FF_n[F^T[P(k/k-1) - Q]^{-1}F + O_n\,]^{-1}K_m \\ P(k+1/k) = Q + C_{LF}(k)P(k/k-1)\,F^T \\ x(k+1/k) = C_{LF}(k)x(k/k-1) + D_{LF}(k)z(k) \end{array}} \quad (29)$$

for $k = 0,1,...$, with initial conditions $x(0/-1) = x_0$, $P(0/-1) = P_0$.

Remark 6.
Time-invariant Prediction Algorithm via KF and Time-invariant Prediction Algorithm via LF have the same structure:

From (23) we get:
$x(k+1/k) = C_{KF}(k)x(k/k-1) + D_{KF}(k)z(k)$

while from (29) we get:
$x(k+1/k) = C_{LF}(k)x(k/k-1) + D_{LF}(k)z(k)$

Due to the fact that the two filters are equivalent, we obtain:

$$\left\{ \begin{array}{l} C_{KF}(k+1) = C_{LF}(k+1) \\ D_{KF}(k+1) = D_{LF}(k+1) \end{array} \right\} \quad (30)$$

4.3 Steady State Prediction Algorithm via LF

For Time-invariant systems, it is well known [1] that if the signal process model is asymptotically stable, then there exists a steady state value P_p of the prediction error covariance matrix. This value remains constant after the steady state time is reached. The Steady State Prediction Algorithm via LF is derived.

Steady State Prediction Algorithm via LF (SSPALF)

$$\boxed{x(k+1/k) = C_{LF}(k)x(k/k-1) + D_{LF}(k)z(k)} \quad (31)$$

for $k = 0,1,...$, with initial condition
$x(0/-1) = x_0$.

where

$$\begin{Bmatrix} C_{LF} = FF_n \ [I + F^{-1}[P_p - Q]F^{-T}O_n]^{-1}F^{-1} \\ D_{LF} = K_n + FF_n[F^{-1}[P_p - Q]F^{-T} + O_n]^{-1}K_m \end{Bmatrix} \quad (32)$$

and P_p is the solution of the Riccati equation:

$$P_p = Q + FP_pF^T - FP_pH^T[HP_pH^T + R]^{-1}HP_pF^T.$$

The steady state coefficients in (32) are calculated off-line by first solving the corresponding discrete time Riccati equation emanating from Kalman filter, [1].

Remark 7.
Steady State Prediction Algorithm via KF and Steady State Prediction Algorithm via LF have the same structure:

From (24) we get:
$x(k+1/k) = C_{KF}x(k/k-1) + D_{KF}(k)z(k)$
while from (31) we get:
$x(k+1/k) = C_{LF}x(k/k-1) + D_{LF}z(k)$

Due to the fact that the two filters are equivalent, we obtain:

$$\begin{Bmatrix} C_{KF} = C_{LF} \\ D_{KF} = D_{LF} \end{Bmatrix} \quad (33)$$

Remark 8.
The following relations between steady state estimation and prediction coefficients hold:

$$\begin{Bmatrix} D_{KF} = FB_{KF} \\ C_{KF} = FA_{KF}F^{-1} \end{Bmatrix} \quad (34)$$

5 FIR Form of the Steady State Prediction Algorithms

The FIR form of the steady state estimation-free prediction algorithms is presented in the following.
From (24) we take:
$x(k+1/k) = C_{KF}x(k/k-1) + D_{KF}z(k)$

Then

$$\begin{aligned} x(1/0) &= C_{KF}x(0/-1) + D_{KF}z(0) \\ x(2/1) &= C_{KF}x(1/0) + D_{KF}z(1) \\ &= C_{KF}^2x(0/-1) + C_{KF}D_{KF}z(0) \\ &\quad + D_{KF}z(1) \end{aligned}$$

…

$$\begin{aligned} x(k+1/k) &= C_{KF}^k x(0/-1) + \mathrm{C_{KF}^{k-1}D_{KF}z(0)} \\ &\quad + \mathrm{C_{KF}^{k-2}D_{KF}z(1)} \dots + C_{KF}D_{KF}z(k-1) \\ &\quad + D_{KF}z(k) \end{aligned}$$

If $|C_{KF}| < 1$, then $\lim_{k\to\infty} C_{KF}^k = 0$, i.e there exists l:$C_{KF}^{l-1} \neq 0$, $C_{KF}^l = 0$

Thus, for $k > l$ we take:

$$\begin{aligned} x(k+1/k) &= \mathrm{C_{KF}^{k-1}D_{KF}z(0)} + \mathrm{C_{KF}^{k-2}D_{KF}z(1)} \dots \\ &\quad + C_{KF}D_{KF}z(k-1) + D_{KF}z(k) \\ &= \sum_{i=0}^{k}\{(\mathrm{C_{KF}^{k-i}D_{KF}})z(i)\} \end{aligned}$$

Hence we derive the following FIR form of the Steady State Prediction Algorithm via KF:

FIR form of Steady State Prediction Algorithm via KF

$$\boxed{\begin{aligned} x(k+1/k) &= \sum_{i=0}^{k}\{c(i)z(i)\} \\ c(i) &= \begin{cases} \mathrm{C_{KF}^{k-i}D_{KF}}, & i \geq k \\ 0, & i < k \end{cases} \end{aligned}} \quad (35)$$

Similarly, we derive the following FIR form of the Steady State Prediction Algorithm via LF:

FIR form of Steady State Prediction Algorithm via LF

$$\boxed{\begin{aligned} x(k+1/k) &= \sum_{i=0}^{k}\{c(i)z(i)\} \\ c(i) &= \begin{cases} \mathrm{C_{LF}^{k-i}D_{LF}}, & i \geq k \\ 0, & i < k \end{cases} \end{aligned}} \quad (36)$$

Remark 9.
The FIR Steady State Prediction Algorithm coefficients are calculated a-priori.
Remark 10.
The prediction depends only on a well-defined set of measurements.

6 Multiple Steps Prediction Algorithms

All the presented estimation-free prediction algorithms compute the one step prediction $x(k+1/k)$ and the corresponding one step prediction error covariance $P(k+1/k)$ and can be used to compute multiple steps prediction and the corresponding multiple step prediction error covariance.

For Time-varying systems, we derive:

$$\begin{Bmatrix} x(\ell/k) = F(\ell,k)x(k/k) \\ P(\ell/k) = Q(\ell-1) + F(\ell)P(k/k)F^T(\ell) \end{Bmatrix} \quad (37)$$

where

$$F(\ell,k) = F(\ell)F(\ell-1)\dots F(k+1) \quad (38)$$

For Time-invariant systems, we derive:

$$\begin{Bmatrix} x(\ell/k) = F^{k-\ell+2}x(k/k) \\ P(\ell/k) = \sum_{i=0}^{\ell-k}\{F^i Q (F^T)^i\} \end{Bmatrix} \quad (39)$$

For steady state systems, we derive:

$$x(\ell/k) = F^{k-\ell+2}x(k/k) \quad (40)$$

7 Computational Requirements

Kalman Filter is the classical prediction algorithm: it uses estimation on order to compute estimation. It is established that Kalman and Lainiotis filters are equivalent and can be used to compute the prediction.

Estimation-free prediction algorithms were derived by Kalman and Lainiotis filters; they are equivalent and compute the prediction.

In order to investigate possible computational advantages of estimation-free prediction algorithms versus classical Kalman filters, we are going to compare estimation-free Prediction Algorithm via KF to Kalman Filter, for Time-varying, Time-invariant and steady state systems. All algorithms are iterative. Then, it is reasonable to assume that they compute the prediction and the prediction error covariance executing the same number of iterations. Thus, in order to compare the algorithms with respect to their computational time, we have to compare their per step (iteration) calculation burden (CB) required for the on-line calculations; the calculation burden of the off-line calculations (initialization process for Time-invariant and steady state algorithms) is not taken into account.

Scalar operations are involved in matrix manipulation operations, which are needed for the implementation of the filtering algorithms. Table 1 summarizes the calculation burden of needed matrix operations. Note that a symmetric matrix is denoted by S. The details are given in [2].

Table 1. Calculation burden of matrix operations

Matrix Operation	Matrix Dimensions	Calculation Burden
$C = A + B$	$(n \times m) + (n \times m)$	nm
$S = A + B$	$(n \times n) + (n \times n)$	$\frac{1}{2}n^2 + \frac{1}{2}n$
$B = I + A$	$(n \times n) + (n \times n)$	n
$C = A \cdot B$	$(n \times m) \cdot (m \times \ell)$	$2nm\ell - n\ell$
$S = A \cdot B$	$(n \times m) \cdot (m \times n)$	$n^2m + nm - \frac{1}{2}n^2 - \frac{1}{2}n$
$B = A^{-1}$	$n \times n, n \geq 2$	$\frac{1}{6}(16n^3 - 3n^2 - n)$

The per iteration calculation burdens of the classical prediction algorithm Kalman Filter (KF) and the proposed prediction algorithm estimation-free Prediction Algorithm via Kalman Filter (PAKF) are analytically calculated in the Appendix and summarized in Table 2.

Table 2. Per iteration calculation burden of prediction algorithms: Kalman Filter (KF) and estimation-free Prediction Algorithm via Kalman Filter (PAKF)

System	Algorithm	Calculation Burden
Time Varying	KF	$CB_{TVKF} = \frac{1}{2}(8n^3 + 7n^2 - 3n)$ $+\frac{1}{6}(16m^3 - 3m^2 - m)$ $+4n^2m + 3nm^2 + nm$
Time Varying	PAKF	$CB_{TVPAKF} = (3n^3 + 3n^2 - n)$ $+\frac{1}{6}(16m^3 - 3m^2 - m)$ $+4n^2m + 3nm^2$
Time Invariant	KF	$CB_{TIKF} = \frac{1}{2}(8n^3 + 7n^2 - 3n)$ $+\frac{1}{6}(16m^3 - 3m^2 - m)$ $+4n^2m + 3nm^2 + nm$
Time Invariant	PAKF	$CB_{TIPAKF} = (3n^3 + 3n^2 - n)$ $+\frac{1}{6}(16m^3 - 3m^2 - m)$ $+4n^2m + 3nm^2$
Steady State	KF	$CB_{SSKF} = 2n^2 + 2nm - n$
Steady State	PAKF	$CB_{SSPAKF} = 2n^2 + 2nm - n$

From Table 2, it is clear that:

- for Time-varying and Time-invariant systems, the estimation-free prediction algorithms are faster than Kalman filter, since
$$CB_{TVKF} - CB_{TVPAKF} = CB_{TIKF} - CB_{TIPAKF} = n^3 + \frac{1}{2}n^2 - \frac{1}{2}n + nm = \frac{1}{2}n(2n^2 + n - 1) + nm > 0 \quad (41)$$
- for steady state systems, the estimation-free prediction algorithm is faster than Kalman filter, since Steady State Kalman Filter can be seen as a prediction algorithm as well by additionally computing the prediction $x(k+1/k) = Fx(k/k)$.

8 Conclusion

Many applications require computation of prediction instead of estimation. For Time-varying, Time-invariant and steady state systems, Kalman Filter can be implemented as a classical prediction algorithm, since it produces the state prediction and the corresponding prediction error covariance matrix via the state estimation and the corresponding estimation covariance matrix. Lainiotis Filter is equivalent to Kalman Filter and can be used to compute the prediction.

In this paper, for Time-varying, Time-invariant and steady state systems, estimation-free Prediction

Algorithms are derived via Kalman and Lainiotis filters; they are equivalent and compute iteratively the prediction and the corresponding prediction error covariance matrix. The estimation and the corresponding estimation error covariance are not needed and are not computed.

The FIR form of the steady state estimation-free prediction algorithms is derived.

The multiple steps prediction algorithms are derived.

The computational requirements of estimation-free prediction algorithms are determined and it shown that the proposed estimation-free prediction algorithms are faster than Kalman filter; this is the main advantage of the proposed algorithms over the classical Kalman filter.

A subject of future research is to investigate the application the proposed estimation free prediction algorithms to dynamical continuous-time systems, [19], to Linear Quadratic Regulator (LQR), [20]. Another area of future research may be the use of the proposed algorithms in the derivation of Time-varying and Time-invariant information filters, using the inverse of the prediction error covariance matrix.

References:

[1] B. D. O. Anderson and J. B. Moore, Optimal Filtering, *Dover Publications*, New York, 2005.

[2] N. Assimakis and M. Adam, Discrete time Kalman and Lainiotis filters comparison, *Int. Journal of Mathematical Analysis (IJMA)* 1(13) (2007) 635-659.

[3] R. Shankar, K. Chatterjee and T.K. Chatterjee, A Very Short-Term Load forecasting using Kalman filter for Load Frequency Control with Economic Load Dispatch, *Journal of Engineering Science and Technology Review*, vol. 5, no 1, pp. 97-103, 2012.

[4] Y. Yang, T. Yu, W. Zhao, X. Zhu, Kalman Filter Photovoltaic Power Prediction Model Based on Forecasting Experience, *Front. Energy Res., Sec. Smart Grids*, vol. 9, 2021, https://doi.org/10.3389/fenrg.2021.682852.

[5] G. Giunta, R. Vernazza, R. Salerno, A. Ceppi, G. Ercolani, M. Mancini, Hourly weather forecasts for gasturbine power generation, *Meteorol.* Z. 26 (2017) 307–317.

[6] Rao V. C. S., Devi B. G., Pratapagiri S., Srinivas C., Venkatramulu S., Raghavakumari, D., Prediction of Covid-19 using Kalman filter algorithm, *2021 International Conference on Research in Sciences, Engineering and Technology, ICRSET 2021*, AIP Conference Proceedings, vol. 2418, Issue 1, id.030067, 8 pp., 2022, DOI: 10.1063/5.0081995.

[7] X. Ren, Y. Yang, J. Zhu, T. Xu, Comparing satellite orbit determination by batch processing and extended Kalman filtering using inter-satellite link measurements of the next-generation beidou satellites. *Gps Solutions* 23(1), 25 (2019).

[8] X. Wang, Y. Yang, B. Wang, Y. Lin, C. Han, Resilient timekeeping algorithm with multi-observation fusion Kalman filter, *Satellite Navigation* 4(1), 25 (2023).

[9] C. Han, L. Liu, Z. Cai, Y. Lin, The space–time references of beidou navigation satellite system. *Satellite navigation* 2, 1–10 (2021).

[10] H.M. Al-Hamadi, S.A. Soliman, Short-term electric load forecasting based on Kalman filtering algorithm with moving window weather and load model, *Electric Power Systems Research* 68 (2004) 47-59.

[11] G. Galanis, P. Louka, P. Katsafados, I. Pytharoulis, and G. Kallos, Applications of Kalman filters based on non-linear functions to numerical weather predictions, *Ann. Geophys.* 24 (2006) 2451–2460.

[12] Y. Sawaragi, T. Soeda, T. Yoshimura, S. Ohe, Y. Chujo, H. Ishihara, The predictions of air pollution levels by nonphysical models based on Kalman filtering method, *J. Dynamic Syst., Meas. And Contr.* 98(12) (1976).

[13] C.Y. Tang, J.B. Peng, Y.H. Deng, Application of a real-time tracking model based on Kalman filter in the prediction of stock price. *Computer Simulation* 22(9), 218–221 (2005).

[14] X. Yan, Z. Guosheng, Application of Kalman filter in the prediction of stock price, *5th International Symposium on Knowledge Acquisition and Modeling (KAM 2015)*, pp. 197–198. Atlantis press (2015).

[15] A. Guven and C. Hajiyev, Two-Stage Kalman Filter Based Estimation of Boeing 747 Actuator/Control Surface Stuck Faults, *WSEAS Transactions on Signal Processing*, vol. 19, 2023, pp. 32-40, DOI: 10.37394/232014.2023.19.4

[16] C. Hajiyev and U. Hacizade, A Covariance Matching-Based Adaptive Measurement Differencing Kalman Filter for INS's Error Compensation, *WSEAS Transactions on Systems and Control*, vol. 18, 2023, pp. 478-486, https://doi.org/10.37394/23203.2023.18.51.

[17] R. Verma, L. Shrinivasan and K. Shreedarshan, GPS/INS integration during GPS outages using machine learning augmented with Kalman filter, *WSEAS Transactions on Systems and Control*, vol. 16, 2021, pp. 294-301, DOI: 10.37394/23203.2021.16.25

[18] D. G. Lainiotis, N. D. Assimakis, S. K. Katsikas, Fast and numerically robust recursive algorithms for solving the discrete time Riccati equation: The case of nonsingular plant noise covariance matrix, *Neural, Parallel, and Scientific Computations* 3(4) (1995) 565-584.

[19] Z.-P. Jiang, T. Bian, and W. Gao, Learning-Based Control: A Tutorial and Some Recent Results, *Foundations, and Trends in Systems and Control*, vol. 8(3), 2022, pp. 985–1007, (176-284).

[20] M.T. Augustine, A note on linear quadratic regulator and Kalman filter, 2023, http://dx.doi.org/10.48550/arXiv.2308.15798.

APPENDIX

A. Calculation Burden of Kalman Filter

1. Time-varying Kalman Filter

The computation burden of Time-varying Kalman Filter (eq. 2) is analytically calculated:

Matrix Operation	Calculation Burden
$x(k+1/k) = F(k+1)x(k/k)$	$2n^2 - n$
$W_1(k) = F(k+1)P(k/k)$	$2n^3 - n^2$
$W_2(k) = W_1(k)\,F^T(k+1)$	$n^3 + \frac{1}{2}n^2 - \frac{1}{2}n$
$P(k+1/k) = Q(k) + W_2(k)$	$\frac{1}{2}n^2 + \frac{1}{2}n$
$W_3(k) = H(k+1)P(k+1/k)$	$2n^2m - nm$
$W_4(k) = W_3(k)H^T(k+1)$	$nm^2 + nm - \frac{1}{2}m^2 - \frac{1}{2}m$
$W_5(k) = W_4(k) + R(k+1)$	$\frac{1}{2}m^2 + \frac{1}{2}m$
$W_6(k) = [W_5(k)]^{-1}$	$\frac{1}{6}(16m^3 - 3m^2 - m)$
$K(k+1) = [W_3(k)]^T W_6(k)$	$2nm^2 - nm$
$W_7(k) = K(k+1)\,H(k+1)$	$2n^2m - n^2$
$W_8(k) = I - W_7(k)$	n
$W_9(k) = W_8(k)x(k+1/k)$	$2n^2 - n$
$W_{10}(k) = K(k+1)z(k+1)$	$2nm - n$
$x(k+1/k+1) = W_9(k) + W_{10}(k)$	n
$P(k+1/k+1) = W_8(k)P(k+1/k)$	$n^3 + \frac{1}{2}n^2 - \frac{1}{2}n$
$CB_{TVKF} = \frac{1}{2}(8n^3 + 7n^2 - 3n)$ $+\frac{1}{6}(16m^3 - 3m^2 - m)$ $+4n^2m + 3nm^2 + nm$	

2. Time-invariant Kalman Filter

The computation burden of Time-invariant Kalman Filter (eq. 3) is equal to the computation burden of Time-varying Kalman Filter (eq. 2):
$\mathrm{CB_{TIKF}} = \mathrm{CB_{TVKF}}$

3. Steady State Kalman Filter

The computation burden of Steady State Kalman Filter (eq. 4) is analytically calculated:

Matrix Operation	Calculation Burden
$W_1(k) = A_{KF}(k)x(k/k)$	$2n^2 - n$
$W_2(k) = B_{KF}(k)z(k+1)$	$2nm - n$
$x(k+1/k+1) = W_1(k) + W_2(k)$	n
$CB_{SSKF} = 2n^2 + 2nm - n$	

B. Calculation Burden of Estimation-Free Prediction Algorithms via Kalman Filter

1. Time-varying Prediction Algorithm via Kalman Filter

The computation burden of Time-varying Prediction Algorithm via Kalman Filter (TVPAKF) Kalman Filter (eq. 22) is analytically calculated:

Matrix Operation	Calculation Burden
$W_1(k) = H(k)P(k/k-1)$	$2n^2m - nm$
$W_2(k) = W_1(k)H^T(k)$	$nm^2 + nm - \frac{1}{2}m^2 - \frac{1}{2}m$
$W_3(k) = W_2(k) + R(k)$	$\frac{1}{2}m^2 + \frac{1}{2}m$
$W_4(k) = [W_3(k)]^{-1}$	$\frac{1}{6}(16m^3 - 3m^2 - m)$
$K(k) = [W_1(k)]^T W_4(k)$	$2nm^2 - nm$
$D_{KF}(k) = F(k+1)K(k)$	$2n^2m - nm$
$C_{KF}(k) = F(k+1) - D_{KF}(k)H(k)$	n^2
$W_5(k) = C_{KF}(k)P(k/k-1)$	$2n^3 - n^2$
$W_6(k) = W_5(k)F^T(k+1,k)$	$n^3 + \frac{1}{2}n^2 - \frac{1}{2}n$
$P(k+1/k) = Q(k) + W_6(k)$	$\frac{1}{2}n^2 + \frac{1}{2}n$
$W_7(k) = C_{KF}(k)x(k/k-1)$	$2n^2 - n$
$W_8(k) = D_{KF}(k)z(k)$	$2nm - n$
$x(k+1/k) = W_7(k) + W_8(k)$	n
$CB_{TVPAKF} = (3n^3 + 3n^2 - n) + \frac{1}{6}(16m^3 - 3m^2 - m) + 4n^2m + 3nm^2$	

2. Time-invariant Prediction Algorithm via Kalman Filter
The computation burden of Time-invariant Prediction Algorithm via Kalman Filter (eq. 23) is equal to the computation burden of Time-varying Prediction Algorithm via Kalman Filter (eq. 22):
$CB_{TIPAKF} = CB_{TVPAKF}$

3. Steady State Prediction Algorithm via Kalman Filter
The computation burden of the Steady State Prediction Algorithm via Kalman Filter (eq. 24) is analytically calculated:

Matrix Operation	Calculation Burden
$W_1(k) = C_{KF}(k)x(k/k-1)$	$2n^2 - n$
$W_2(k) = D_{KF}(k)z(k)$	$2nm - n$
$x(k+1/k) = W_1(k) + W_2(k)$	n
$CB_{PAKF} = 2n^2 + 2nm - n$	

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)
The authors equally contributed to the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself
No funding was received for conducting this study.

Conflict of Interest
The authors have no conflicts of interest to declare.