2024 American Control Conference (ACC)
July 8-12, 2024. Toronto, Canada

Adaptive Kalman Filtering Developed from Recursive Least Squares Forgetting Algorithms

Brian Lai and Dennis S. Bernstein

***Abstract*— Recursive least squares (RLS) is derived as the recursive minimizer of the least-squares cost function. Moreover, it is well known that RLS is a special case of the Kalman filter. This work presents the Kalman filter least squares (KFLS) cost function, whose recursive minimizer gives the Kalman filter. KFLS is an extension of generalized forgetting recursive least squares (GF-RLS), a general framework which contains various extensions of RLS from the literature as special cases. This then implies that extensions of RLS are also special cases of the Kalman filter. Motivated by this connection, we propose an algorithm that combines extensions of RLS with the Kalman filter, resulting in a new class of adaptive Kalman filters. A numerical example shows that one such adaptive Kalman filter provides improved state estimation for a mass-spring-damper with intermittent, unmodeled collisions. This example suggests that such adaptive Kalman filtering may provide potential benefits for systems with non-classical disturbances.**

I. Introduction

Despite their respective deterministic and stochastic foundations, least-squares and the Kalman filter share an interconnected history [1]. It is well known that the update equations for recursive least squares (RLS) (e.g. [2]) are the same as those of the Kalman filter with, for all $k \geq 0$, identity state matrix $A_k = I$, zero input matrix $B_k = 0$, process noise covariance $\Sigma_k = 0$, and measurement noise covariance $\Gamma_k = I$ (see p.51 of [3], section 3.3.5 of [4], or p.129 of [5]). As RLS became a foundational algorithm of systems and control theory for online identification of fixed parameters [3], [6], numerous extensions of RLS were developed (e.g. [7]–[16]) to improve identification of time-varying parameters. However, little work has been done to connect these extensions to the Kalman filter.

The RLS update equations are often derived as the recursive minimizer of a least squares cost function (e.g. [2]). A natural question is whether the RLS cost function can be generalized to derive the Kalman filter. While other deterministic derivations of the Kalman filter have been presented (e.g. [17], [18]), these do not follow as an extension of the RLS cost.

This work presents the Kalman filter least squares (KFLS) cost function whose recursive minimizer gives the Kalman filter update equations. KFLS is an extension of generalized forgetting recursive least squares (GF-RLS) [19], which contains various extensions of RLS from the literature as special cases. As a result, these extensions of RLS are also special cases of the Kalman filter with particular choices of the process noise covariance matrix.

This result motivates a new class of adaptive Kalman filtering, with modified prior covariance update equations to incorporate forgetting from extensions of RLS. A brief survey is given to show how several forgetting methods from the RLS literature can be applied to adaptive Kalman filtering. A numerical example shows how adaptive Kalman filtering with robust variable forgetting factor [14] improves state estimation of a mass-spring-damper system with intermittent, unmodeled collisions. This example suggests that such an adaptive Kalman filtering may be beneficial when disturbances are non-classical.

1) Notation and Terminology: For symmetric $P, Q \in \mathbb{R}^{n\times n}$, $P \prec Q$ (respectively, $P \preceq Q$) denotes that $Q - P$ is positive definite (respectively, positive semidefinite). For all $x \in \mathbb{R}^n$, let $\|x\| \triangleq \sqrt{x^{\mathrm{T}}x}$. For $x \in \mathbb{R}^n$ and positive-semidefinite $R \in \mathbb{R}^{n\times n}$, $\|x\|_R \triangleq \sqrt{x^{\mathrm{T}}Rx}$.

II. Background Material

A. The one-step Kalman Filter

Consider the discrete-time, linear, time-varying system

$$x_{k+1} = A_k x_k + B_k u_k + w_k, \tag{1}$$

$$y_k = C_k x_k + v_k, \tag{2}$$

where, for all $k \geq 0$, $x_k \in \mathbb{R}^n$ is the state, $y_k \in \mathbb{R}^p$ is the measurement, $u_k \in \mathbb{R}^m$ is the input, $w_k \sim \mathcal{N}(0, \Sigma_k)$ is the process noise, and $v_k \sim \mathcal{N}(0, \Gamma_k)$ is the measurement noise, for positive-semidefinite $\Sigma_k \in \mathbb{R}^{n\times n}$ and positive-semidefinite $\Gamma_k \in \mathbb{R}^{p\times p}$. Moreover, for all $k \geq 0$, $A_k \in \mathbb{R}^{n\times n}$, $B_k \in \mathbb{R}^{n\times m}$, and $C_k \in \mathbb{R}^{p\times n}$. The *two-step Kalman filter* [20] for the system given by (1) and (2) is can be expressed as

$$\hat{x}_{k+1|k} = A_k \hat{x}_k + B_k u_k, \tag{3}$$

$$P_{k+1|k} = A_k P_k A_k^{\mathrm{T}} + \Sigma_k, \tag{4}$$

$$K_k = P_{k+1|k} C_k^{\mathrm{T}} (C_k P_{k+1|k} C_k^{\mathrm{T}} + \Gamma_k)^{-1} \tag{5}$$

$$\hat{x}_{k+1} = \hat{x}_{k+1|k} + K_k (y_k - C_k \hat{x}_{k+1|k}), \tag{6}$$

$$P_{k+1} = P_{k+1|k} - K_k C_k P_{k+1|k}, \tag{7}$$

where, for all $k \geq 0$, positive-definite $P_{k+1|k} \in \mathbb{R}^{n\times n}$ and $P_k \in \mathbb{R}^{n\times n}$ are, respectively, the prior and posterior covariances, $\hat{x}_{k+1|k} \in \mathbb{R}^n$ and $\hat{x}_k \in \mathbb{R}^n$ are, respectively, the prior and posterior state estimates, and $K_k \in \mathbb{R}^{n\times p}$ is the Kalman gain.

Next, if, for all $k \geq 0$, $\Sigma_k + A_k P_k A_k^{\mathrm{T}}$ and Γ_k are nonsingular, then the matrix inversion lemma (Lemma 1) can

Brian Lai and Dennis S. Bernstein are with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI, USA. {brianlai, dsbaero}@umich.edu. This work was supported by the NSF Graduate Research Fellowship under Grant No. DGE 1841052.

be used to rewrite (3) through (7) as the *one-step Kalman filter* [18], where, for all $k \geq 0$,

$$P_{k+1}^{-1} = (\Sigma_k + A_k P_k A_k^{\mathrm{T}})^{-1} + C_k^{\mathrm{T}} \Gamma_k^{-1} C_k, \tag{8}$$

$$\hat{x}_{k+1} = A_k \hat{x}_k + B_k u_k + P_{k+1} C_k^{\mathrm{T}} \Gamma_k^{-1} \left(y_k - C_k (A_k \hat{x}_k + B_k u_k) \right). \tag{9}$$

B. Discrete-Time LTV State Transition Function

To facilitate expressing a least squares cost function for the Kalman filter, we first introduce the *state transition function* for discrete-time LTV system, a concise notation to transition between states at different time steps. For all $k \geq 0$, let $A_k \in \mathbb{R}^{n \times n}$, $B_k \in \mathbb{R}^{n \times m}$, $x_k \in \mathbb{R}^n$, and $u_k \in \mathbb{R}^m$. Consider the discrete-time, linear time-varying state update equation

$$x_{k+1} = A_k x_k + B_k u_k. \tag{10}$$

Assume, for all $k \geq 0$, A_k is nonsingular. For all $i, k \geq 0$, define the *state transition matrix* (from step i to step k), denoted $\Phi_{k,i} \in \mathbb{R}^{n \times n}$, by

$$\Phi_{k,i} \triangleq \begin{cases} A_{k-1} A_{k-2} \cdots A_{i+1} A_i & i < k, \\ I & i = k, \\ A_k^{-1} A_{k+1}^{-1} \cdots A_{i-2}^{-1} A_{i-1}^{-1} & k < i. \end{cases} \tag{11}$$

It follows that, for all $i, k \geq 0$, $\Phi_{i,k}^{-1} = \Phi_{k,i}$. Then, for all $0 \leq i < k$, x_k can be expressed as

$$\begin{aligned} x_k &= A_{k-1} x_{k-1} + B_{k-1} u_{k-1}, \\ &= A_{k-1} A_{k-2} x_{k-2} + A_{k-1} B_{k-2} u_{k-2} + B_{k-1} u_{k-1}, \\ &= \cdots = \Phi_{k,i} x_i + \sum_{j=i}^{k-1} \Phi_{k,j+1} B_j u_j. \end{aligned} \tag{12}$$

For all $0 \leq i < k$, we further define the matrices

$$\mathcal{B}_{k,i} \triangleq \begin{bmatrix} \Phi_{k,i+1} B_i & \cdots & \Phi_{k,k-1} B_{k-2} & \Phi_{k,k} B_{k-1} \end{bmatrix}, \tag{13}$$

$$\mathcal{U}_{k,i} \triangleq \begin{bmatrix} u_i^{\mathrm{T}} & \cdots & u_{k-2}^{\mathrm{T}} & u_{k-1}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}. \tag{14}$$

It follows that $\mathcal{B}_{k,i} \mathcal{U}_{k,i} = \sum_{j=i}^{k-1} \Phi_{k,j+1} B_j u_j$. Hence, for all $0 \leq i < k$, $x_k = \Phi_{k,i} x_i + \mathcal{B}_{k,i} \mathcal{U}_{k,i}$. On the other hand, for all $0 \leq k < i$, $x_k = \Phi_{k,i}(x_i - \mathcal{B}_{i,k} \mathcal{U}_{i,k})$. Therefore, for all $i \geq 0$ and $k \geq 0$, we define the *state transition function* (from step i to step k), denoted $\mathcal{T}_{k,i} : \mathbb{R}^n \to \mathbb{R}^n$, by

$$\mathcal{T}_{k,i}(x) \triangleq \begin{cases} \Phi_{k,i} x + \mathcal{B}_{k,i} \mathcal{U}_{k,i} & i < k, \\ x & i = k, \\ \Phi_{k,i}(x - \mathcal{B}_{i,k} \mathcal{U}_{i,k}) & k < i. \end{cases} \tag{15}$$

III. A Least Squares Cost Function which Derives the Kalman Filter

This section develops the Kalman filter least squares (KFLS) cost function whose recursive minimizer gives the one-step Kalman filter update equations. To begin, for all $k \geq 0$, let $F_k \in \mathbb{R}^{n \times n}$ be the *forgetting matrix*. Theorem 1 develops the KFLS cost function (17) in terms of F_k and shows how the update equations (21) and (22) minimize that cost. Corollary 2 will later show how, for a particular choice of F_k, the update equations of Theorem 1 are equivalent to the one-step Kalman filter.

Theorem 1. *For all $k \geq 0$, let $A_k \in \mathbb{R}^{n \times n}$ be nonsingular, $B_k \in \mathbb{R}^{n \times m}$, $C_k \in \mathbb{R}^{p \times n}$, $\Gamma_k \in \mathbb{R}^{p \times p}$ be positive definite, $u_k \in \mathbb{R}^m$, and $y_k \in \mathbb{R}^p$. Furthermore, let $P_0 \in \mathbb{R}^{n \times n}$ be positive definite and $\hat{x}_0 \in \mathbb{R}^n$. For all $k \geq 0$, Let $F_k \in \mathbb{R}^{n \times n}$ be positive semidefinite and such that*

$$F_k \prec \Phi_{0,k}^{\mathrm{T}} P_0^{-1} \Phi_{0,k} + \sum_{i=0}^{k-1} \left(\Phi_{i+1,k}^{\mathrm{T}} C_i^{\mathrm{T}} \Gamma_i^{-1} C_i \Phi_{i+1,k} - \Phi_{i,k}^{\mathrm{T}} F_i \Phi_{i,k} \right). \tag{16}$$

For all $k \geq 0$, define $J_k \colon \mathbb{R}^n \to \mathbb{R}$ as

$$J_k(\hat{x}) \triangleq J_{k,\mathrm{loss}}(\hat{x}) - J_{k,\mathrm{forget}}(\hat{x}) + J_{k,\mathrm{reg}}(\hat{x}), \tag{17}$$

where

$$J_{k,\mathrm{loss}}(\hat{x}) \triangleq \sum_{i=0}^{k} \| y_i - C_i \mathcal{T}_{i+1,k+1}(\hat{x}) \|_{\Gamma_i^{-1}}^2 \tag{18}$$

$$J_{k,\mathrm{forget}}(\hat{x}) \triangleq \sum_{i=0}^{k} \| \mathcal{T}_{i,k+1}(\hat{x}) - \hat{x}_i \|_{F_i}^2, \tag{19}$$

$$J_{k,\mathrm{reg}}(\hat{x}) \triangleq \| \mathcal{T}_{0,k+1}(\hat{x}) - \hat{x}_0 \|_{P_0^{-1}}^2. \tag{20}$$

Then, there exists a unique global minimizer of $J_k(\hat{x})$, denoted $\hat{x}_{k+1} \triangleq \arg\min_{\hat{x} \in \mathbb{R}^n} J_k(\hat{x})$, which, for all $k \geq 0$, is given recursively by

$$P_{k+1}^{-1} = A_k^{-\mathrm{T}} (P_k^{-1} - F_k) A_k^{-1} + C_k^{\mathrm{T}} \Gamma_k^{-1} C_k, \tag{21}$$

$$\hat{x}_{k+1} = A_k \hat{x}_k + B_k u_k + P_{k+1} C_k^{\mathrm{T}} \Gamma_k^{-1} \left(y_k - C_k (A_k \hat{x}_k + B_k u_k) \right), \tag{22}$$

where, for all $k \geq 0$, $P_k \in \mathbb{R}^{n \times n}$ is positive definite.

Proof. See the Appendix. □

Corollary 1. *Consider the notation and assumptions of Theorem 1. For all $k \geq 0$,*

$$P_k^{-1} = \Phi_{0,k}^{\mathrm{T}} P_0^{-1} \Phi_{0,k} + \sum_{i=0}^{k-1} \left(\Phi_{i+1,k}^{\mathrm{T}} C_i^{\mathrm{T}} \Gamma_i^{-1} C_i \Phi_{i+1,k} - \Phi_{i,k}^{\mathrm{T}} F_i \Phi_{i,k} \right), \tag{23}$$

and hence (16) *hold if and only if $P_k^{-1} - F_k \succ 0$.*

Proof. Let $k \geq 0$. Note that (23) follows from repeated substitution of (21). Next, note that the right-hand side of (23) is equivalent to the right-hand side of (16). Hence, (16) is equivalent to $P_k^{-1} - F_k \succ 0$ □

Corollary 2. *Consider the notation and assumptions of Theorem 1. If, for all $k \geq 0$, there exists positive-semidefinite $\Sigma_k \in \mathbb{R}^{n \times n}$ such that*

$$F_k = P_k^{-1} - (A_k^{-1} \Sigma_k A_k^{-\mathrm{T}} + P_k)^{-1}, \tag{24}$$

then, for all $k \geq 0$, (16) *is satisfied. Moreover,* (21) *and* (22) *are equivalent to the one-step Kalman filter update equations* (8) *and* (9).

Proof. Let $k \geq 0$. Note that (24) can be rewritten as

$$(\Sigma_k + A_k P_k A_k^{\mathrm{T}})^{-1} = A_k^{-\mathrm{T}} (P_k^{-1} - F_k) A_k^{-1}. \tag{25}$$

Substituting (25) into (21), it follows that (21) and (22) are equivalent to (8) and (9). Moreover, (24) can also be expressed as $P_k^{-1} - F_k = (A_k^{-1}\Sigma_k A_k^{-\mathrm{T}} + P_k)^{-1}$. Since Σ_k and P_k are both positive definite, it follows that $A_k^{-1}\Sigma_k A_k^{-\mathrm{T}} + P_k$ is positive definite, and hence $P_k^{-1} - F_k$ is positive definite. Therefore, by Corollary 1, condition (16) is satisfied. □

Next, Proposition 1 shows that, for all $k \geq 0$, Σ_k has the same definiteness as F_k.

Proposition 1. *For all $k \geq 0$, Σ_k is positive semidefinite (resp. positive definite) if and only on F_k is positive semidefinite (resp. positive definite).*

Proof. Let $k \geq 0$. If F_k is positive semidefinite (respectively, positive definite), then $P_k^{-1} - F_k \preceq P_k^{-1}$ (respectively, $P_k^{-1} - F_k \prec P_k^{-1}$). Since $P_k^{-1} - F_k$ is nonsingular by Corollary 1, it follows that $(P_k^{-1} - F_k)^{-1} \succeq P_k$ and hence $(P_k^{-1} - F_k)^{-1} - P_k \succeq 0$ (respectively, $(P_k^{-1} - F_k)^{-1} - P_k \succ 0$). Finally, since A_k is nonsingular by assumption, it follows from (29) that Σ_k is positive semidefinite (respectively, positive definite).

Next, if Σ_k is positive semidefinite (respectively, positive definite), then $\Sigma_k + P_k \succeq P_k$ (respectively, $\Sigma_k + P_k \succ P_k$) and hence $(\Sigma_k + P_k)^{-1} \preceq P_k^{-1}$ (respectively, $(\Sigma_k + P_k)^{-1} \prec P_k^{-1}$). Therefore, $F_k = P_k^{-1} - (\Sigma_k + P_k)^{-1} \succeq 0$ (respectively, $F_k = P_k^{-1} - (\Sigma_k + P_k)^{-1} \succ 0$). □

IV. RLS Extensions as Special Cases of the Kalman Filter

Revisiting the state update equation (10), note that if, for all $k \geq 0$, $A_k = I_n$ and $B_k = 0_{n\times m}$, then, for all $k \geq 0$, $x_{k+1} = x_k$. This also implies that the state transition function, given by (15), is identity. In particular, for all $i \geq 0$, $k \geq 0$, and $x \in \mathbb{R}^n$, $\mathcal{T}_{k,i}(x) = x$. Substituting the identity state transition function into the cost function J_k given by (17), it follows that, for all $k \geq 0$,

$$J_{k,\mathrm{loss}}(\hat{x}) = \sum_{i=0}^{k} \|y_i - C_i\hat{x}\|^2_{\Gamma_i^{-1}} \tag{26}$$

$$J_{k,\mathrm{forget}}(\hat{x}) = \sum_{i=0}^{k} \|\hat{x} - \hat{x}_i\|^2_{F_i}, \tag{27}$$

$$J_{k,\mathrm{reg}}(\hat{x}) = \|\hat{x} - \hat{x}_0\|^2_{P_0^{-1}}. \tag{28}$$

Note that, in this special case, the cost function J_k is equivalent to the generalized forgetting recursive least squares (GF-RLS) cost developed in [19], where GF-RLS uses the notation $\theta \in \mathbb{R}^n$ and $\phi_k \in \mathbb{R}^{p\times n}$ instead of $\hat{x} \in \mathbb{R}^n$ and $C_k \in \mathbb{R}^{p\times n}$, respectively. In [19], it was shown that various extensions of RLS from the literature are special cases of GF-RLS if, for all $k \geq 0$, a particular forgetting matrix $F_k \in \mathbb{R}^{n\times n}$ is chosen.

Therefore, we conclude that an extension of RLS, which is a special case of GF-RLS with forgetting matrix F_k, $k \geq 0$, is also a special case of the Kalman filter if, for all $k \geq 0$, $A_k = I_n$, $B_k = 0_{n\times m}$, and there exists positive-semidefinite $\Sigma_k \in \mathbb{R}^{n\times n}$ such that (24) holds. Explicitly solving for Σ_k, it follows that

$$\Sigma_k = A_k \left[(P_k^{-1} - F_k)^{-1} - P_k\right] A_k^{\mathrm{T}}, \tag{29}$$

where $P_k^{-1} - F_k$ is nonsingular by (16) and Corollary 1. Moreover, by Proposition 1, Σ_k is positive semidefinite if and only if F_k is positive semidefinite.

While [19] gives a thorough literature review on extensions of RLS as special cases of GF-RLS, for brevitiy, we summarize eight extensions in Table I. Given are the algorithm name and reference, assumptions of the algorithm, tuning parameters, and $\Sigma_k \in \mathbb{R}^{n\times n}$ derived from (29).

V. Adaptive Kalman Filtering Developed from RLS Forgetting Algorithms

Thus far, we have shown that various extensions of RLS from the literature are special cases of the Kalman filter, in part, by a special choice of the process noise covariance given by (29). Motivated by this relationship, we propose a class of adaptive Kalman filters by replacing the prior covariance update equation (4) with the adaptive prior covariance update equations

$$P_{\mathrm{forget},k} = P_k + \Sigma_{\mathrm{forget},k}, \tag{30}$$

$$P_{k+1|k} = A_k P_{\mathrm{forget},k} A_k^{\mathrm{T}} + \Sigma_{\mathrm{Kalman},k}, \tag{31}$$

where, for all $k \geq 0$, positive-semidefinite $\Sigma_{\mathrm{forget},k} \in \mathbb{R}^{n\times n}$ is designed from an extension of RLS (e.g. right column of Table I), and positive-semidefinite $\Sigma_{\mathrm{Kalman},k} \in \mathbb{R}^{n\times n}$ is designed using traditional methods of the Kalman filter.

While there are as many variants of this algorithm as extensions of RLS, we select a particular extension of RLS to show the potential benefits in state estimation.

A. Kalman Filter with Robust Variable Forgetting Factor

Consider the mass-spring-damper system in Figure 1, with mass $m = 10$, spring constant $k = 5$, and damping coefficient $c = 3$. The vertical displacement of the mass at time t is $z(t)$, where $z = 0$ when the mass is at rest. Assume that $z(0) = -1$ and $\dot{z}(0) = 1$ Furthermore, a downward force $F(t) = 10\sin(t)$ is applied on the mass at time t. This nominal mass-spring-damper system can be modeled as

$$\begin{bmatrix}\dot{z}(t)\\ \ddot{z}(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\ -{}^{k}\!/\!{}_{m} & -{}^{b}\!/\!{}_{m}\end{bmatrix}\begin{bmatrix}z(t)\\ \dot{z}(t)\end{bmatrix} + \begin{bmatrix}0\\ {}^{1}\!/\!{}_{m}\end{bmatrix} F(t). \tag{32}$$

However, we additionally assume there is a wall at $z = 2$. If the mass collides with this wall, the mass reverses direction with the same speed. Note that such intermittent collisions can be interpreted as impulsive disturbances on the system. Finally, we consider, for all $k \geq 0$, the measurements $y_k \in \mathbb{R}$, given by

$$y_k = z(kT_s) + \dot{z}(kT_s) + v_k, \tag{33}$$

where $T_s = 0.1$ is the sampling time and, for all $k \geq 0$, $v_k \sim \mathcal{N}(0, \Gamma_k)$ is the measurement noise with $\Gamma_k = 0.01$. The goal is to estimate the vertical displacement $z(t)$ and velocity $\dot{z}(t)$ without knowledge of the wall at $z = 2$. We will assume that the nominal model (32) and the measurement noise covariance Γ_k are known.

TABLE I: RLS Extensions as Special Cases of the Kalman Filter

Algorithm	Tuning Parameters	Process Noise Covariance Σ_k
1. Recursive Least Squares [2]	—	$\Sigma_k = 0_{n\times n}$
2. Exponential Forgetting [2], [21]	$\lambda \in (0,1]$	$\Sigma_k = (\frac{1}{\lambda} - 1)P_k$.
3. Variable-Rate Forgetting [7]	$\lambda_k \in (0,1]$, $k \geq 0$	$\Sigma_k = (\frac{1}{\lambda_k} - 1)P_k$.
4. Data-Dependent Forgetting [16]	$\mu_{-1} = 1$ and $\mu_k \in [0,1), k \geq 0$	$\Sigma_k = (\frac{\mu_k}{(1-\mu_k)\mu_{k-1}} - 1)P_k$.
5. Exponential Resetting [12]	positive-definite $P_\infty \in \mathbb{R}^{n\times n}$	$\Sigma_k = (\lambda P_k^{-1} + (1-\lambda)P_\infty^{-1})^{-1} - P_k$.
6. Covariance Resetting [22]	positive-definite $P_\infty \in \mathbb{R}^{n\times n}$ and resetting criteria	$\Sigma_k = \begin{cases} P_\infty - P_k & \text{criteria met,} \\ 0_{n\times n} & \text{otherwise.} \end{cases}$
7. Directional Forgetting by Information Matrix Decomp. [8]	$\lambda \in (0,1]$	$\Sigma_k = \frac{1-\lambda}{\lambda} C_k^{\mathrm{T}} (C_k P_k^{-1} C_k^{\mathrm{T}})^{-1} C_k$.
8. Variable-Direction Forgetting [21]	Positive-definite $\Lambda_k \in \mathbb{R}^{n\times n}$, $k \geq 0$	$\Sigma_k = \Lambda_k^{-1} P_k^{-1} \Lambda_k^{-1} - P_k$.

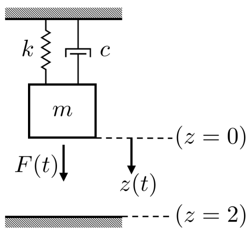

Fig. 1: Mass-spring-damper system diagram. The mass can collide with the wall at $z = 2$, reversing direction and keeping the same speed.

We begin by discretizing (32) and (33) using zero-order hold and sampling time T_s to obtain the nominal system

$$\begin{aligned} x_{k+1} &= A_k x_k + B_k u_k, \\ y_k &= C_k x_k + v_k, \end{aligned} \tag{34}$$

where, for all $k \geq 0$, $x_k \triangleq [z(kT_s)\ \dot{z}(kT_s)]^{\mathrm{T}}$, $u_k \triangleq F(kT_s)$,

$$A_k \triangleq \begin{bmatrix} 0.9975 & 0.09843 \\ -0.04922 & 0.9680 \end{bmatrix}, \quad B_k \triangleq \begin{bmatrix} 4.948 \times 10^{-4} \\ 9.843 \times 10^{-3} \end{bmatrix}, \\ C_k \triangleq \begin{bmatrix} 1 & 1 \end{bmatrix}. \tag{35}$$

We first consider state estimation using the Kalman filter with the nominal discrete system (34), and $P_0 = 0.1 I_2$, $\hat{x}_0 = [0\ 0]^{\mathrm{T}}$, and, for all $k \geq 0$, $\Gamma_k = 0.01$, $\Sigma_k = 0.01 I_2$.

Second, we will consider an adaptive Kalman filter with adaptive prior covariance update equations (30) and (31) developed from variable-rate forgetting [7]. This adaptive Kalman filter also uses the nominal discrete system (34), and $P_0 = 0.1 I_2$, $\hat{x}_0 = [0\ 0]^{\mathrm{T}}$, and, for all $k \geq 0$, $\Gamma_k = 0.01$. Then, for all $k \geq 0$, let

$$\Sigma_{\mathrm{Kalman},k} = 0.01 I_2, \quad \Sigma_{\mathrm{forget},k} = (\frac{1}{\lambda_k} - 1)P_k, \tag{36}$$

where, the forgetting factor $\lambda_k \in (0,1]$ is chosen using the robust variable forgetting factor algorithm developed in [14]. We've chosen this algorithm for its ability to improve tracking of impulsive changes of parameters [14]. For all $k \geq 0$, let

$$\hat{\sigma}_{e,k}^2 = \alpha \hat{\sigma}_{e,k-1}^2 + (1-\alpha)(y_k - C_k \hat{x}_k)^2, \tag{37}$$
$$\hat{\sigma}_{q,k}^2 = \alpha \hat{\sigma}_{q,k-1}^2 + (1-\alpha)(\hat{x}_k^{\mathrm{T}} P_k \hat{x}_k)^2, \tag{38}$$
$$\hat{\sigma}_{v,k}^2 = \beta \hat{\sigma}_{v,k-1}^2 + (1-\beta)(y_k - C_k \hat{x}_k)^2, \tag{39}$$

where $\hat{\sigma}_{e,-1} \triangleq \hat{\sigma}_{q,-1} \triangleq \hat{\sigma}_{v,-1} \triangleq 1$, $\alpha \triangleq 1 - \frac{1}{K_\alpha n}$, and $\beta \triangleq 1 - \frac{1}{K_\beta n}$, where $K_\alpha \triangleq 2$, $K_\beta \triangleq 10$, and $n = 2$ since the system is second-order.

Then, for all $k \geq 0$, if $\hat{\sigma}_{e,k} \leq \hat{\sigma}_{v,k}$, $\lambda_k = \lambda_{\max}$, otherwise

$$\lambda_k = \max\left\{ \min\left\{ \frac{\hat{\sigma}_{q,k}\hat{\sigma}_{v,k}}{\xi + |\hat{\sigma}_{e,k} - \hat{\sigma}_{v,k}|}, \lambda_{\max} \right\}, \lambda_{\min} \right\}, \tag{40}$$

where $\xi \triangleq 10^{-6}$, $\lambda_{\min} \triangleq 0.5$, and $\lambda_{\max} \triangleq 1$. For details on robust variable forgetting and tuning of parameters, see [14].

Figure 2 shows state estimation of the Kalman filter (KF) and the adaptive Kalman filter (KF*), as well as the forgetting factor λ_k, all with zero-order hold. Note that after each of the four collisions the mass makes with the wall at $z = 2$, the forgetting factor λ_k briefly but drastically decreases. This results in improved displacement and velocity estimation immediately after each collision. This can be more clearly seen in the error between the true state and the estimated state in Figure 3. Figure 4 shows $\sigma_{\hat{z}}$ and $\sigma_{\hat{\dot{z}}}$, the diagonal elements of the covariance matrix P_k, which can also be interpreted as the marginal variance of the displacement and velocity state estimate, respectively. Note that in the adaptive Kalman filter (KF*), there is a sudden increase in both $\sigma_{\hat{z}}$ and $\sigma_{\hat{\dot{z}}}$ after each collision, allowing for quicker adaptation.

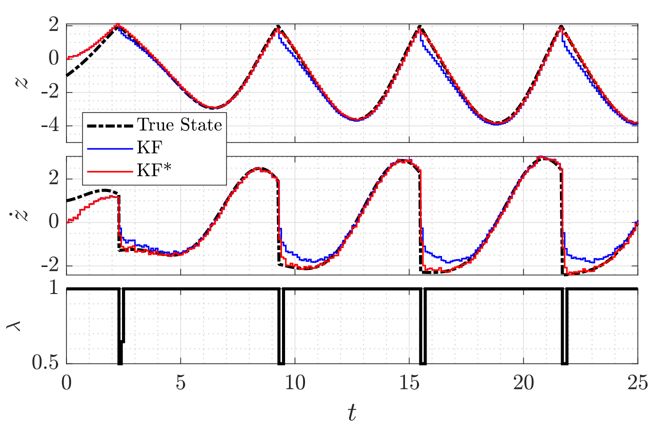

Fig. 2: Vertical displacement (z) and velocity ($\dot{z}$) estimation using Kalman filter (KF) and adaptive Kalman filter (KF*). λ shows the forgetting factor used in KF*.

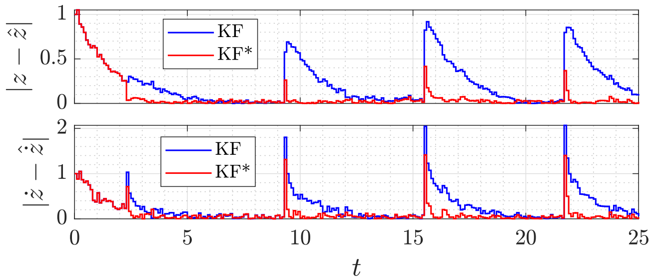

Fig. 3: Estimation error of vertical displacement (top) and velocity (bottom) using Kalman filter (KF) and adaptive Kalman filter (KF*).

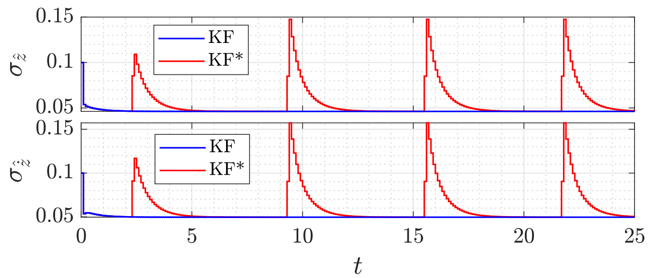

Fig. 4: Marginal variance of vertical displacement (top) and velocity (bottom) using Kalman filter (KF) and adaptive Kalman filter (KF*).

VI. Conclusion

This work presents the Kalman filter least squares cost function whose recursive minimizer gives the Kalman filter update equations. An important consequence of this cost function is that various extensions of RLS from the literature are special cases of the Kalman filter. Motivated by this result, we propose a new adaptive Kalman filters, whose prior covariance update is modified to include RLS forgetting. While the numerical example we presented shows the potential benefits of adaptive Kalman filtering with robust variable forgetting factor [14] in the presence of impulsive disturbances, there are numerous other forgetting algorithms in the RLS literature to be considered (several summarized in Table I). Future work includes further exploration into how, and in what situations, such extensions may be beneficial.

References

[1] H. W. Sorenson, "Least-squares estimation: from gauss to kalman," *IEEE Spectrum*, vol. 7, no. 7, pp. 63–68, 1970.
[2] S. Islam and D. Bernstein, "Recursive least squares for real-time implementation," *IEEE Ctrl. Sys. Mag.*, vol. 39, no. 3, p. 82, 2019.
[3] K. J. Åström and B. Wittenmark, *Adaptive control*. Courier Co., 2013.
[4] J. L. Crassidis and J. L. Junkins, *Optimal estimation of dynamic systems*. Chapman and Hall/CRC, 2004.
[5] D. Simon, *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.
[6] L. Ljung and T. Söderström, *Theory and practice of recursive identification*. MIT press, 1983.
[7] A. L. Bruce, A. Goel, and D. S. Bernstein, "Convergence and consistency of recursive least squares with variable-rate forgetting," *Automatica*, vol. 119, p. 109052, 2020.
[8] L. Cao and H. Schwartz, "A directional forgetting algorithm based on the decomposition of the information matrix," *Automatica*, vol. 36, no. 11, pp. 1725–1731, 2000.
[9] R. Kulhavỳ and M. Kárnỳ, "Tracking of slowly varying parameters by directional forgetting," *IFAC Proceedings Volumes*, vol. 17, no. 2, pp. 687–692, 1984.
[10] S. Bittanti, P. Bolzern, and M. Campi, "Convergence and exponential convergence of identification algorithms with directional forgetting factor," *Automatica*, vol. 26, no. 5, pp. 929–932, 1990.
[11] M. E. Salgado, G. C. Goodwin, and R. H. Middleton, "Modified least squares algorithm incorporating exponential resetting and forgetting," *International Journal of Control*, vol. 47, no. 2, pp. 477–491, 1988.
[12] B. Lai and D. S. Bernstein, "Exponential resetting and cyclic resetting recursive least squares," *IEEE Ctrl. Sys. Lets.*, vol. 7, pp. 985–990, 2023.
[13] A. Vahidi *et al.*, "Recursive least squares with forgetting for online estimation of vehicle mass and road grade: theory and experiments," *Vehicle Sys. Dynms.*, vol. 43, no. 1, pp. 31–55, 2005.
[14] C. Paleologu, J. Benesty, and S. Ciochina, "A robust variable forgetting factor recursive least-squares algorithm for system identification," *IEEE Signal Processing Letters*, vol. 15, pp. 597–600, 2008.
[15] R. M. Johnstone, C. R. Johnson Jr, R. R. Bitmead, and B. D. Anderson, "Exponential convergence of recursive least squares with exponential forgetting factor," *Sys. & Ctrl. Letters*, vol. 2, no. 2, pp. 77–82, 1982.
[16] S. Dasgupta and Y.-F. Huang, "Asymptotically convergent modified recursive least-squares with data-dependent updating and forgetting factor for systems with bounded noise," *IEEE Transactions on Information Theory*, vol. 33, no. 3, pp. 383–392, 1987.
[17] J. Humpherys and J. West, "Kalman filtering with newton's method," *IEEE Control Systems Magazine*, vol. 30, no. 6, pp. 101–106, 2010.
[18] J. Humpherys, P. Redd, and J. West, "A fresh look at the kalman filter," *SIAM review*, vol. 54, no. 4, pp. 801–823, 2012.
[19] B. Lai and D. S. Bernstein, "Generalized forgetting recursive least squares: Stability and robustness guarantees," 2023.
[20] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.
[21] A. Goel, A. L. Bruce, and D. S. Bernstein, "Recursive least squares with variable-direction forgetting: Compensating for the loss of persistency," *IEEE Ctrl. Sys. Mag.*, vol. 40, no. 4, pp. 80–102, 2020.
[22] G. Goodwin, E. Teoh, and H. Elliott, "Deterministic convergence of a self-tuning regulator with covariance resetting," in *IEE Proceedings D-Control Theory and Applications*, vol. 1, no. 130, 1983, pp. 6–8.

Appendix

Lemma 1. *Let $A \in \mathbb{R}^{n\times n}$, $U \in \mathbb{R}^{n\times p}$, $C \in \mathbb{R}^{p\times p}$, $V \in \mathbb{R}^{p\times n}$. Assume A, C, and $A+UCV$ are nonsingular. Then, $(A+UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1}+VA^{-1}U)^{-1}VA^{-1}$.*

Lemma 2. *Let $A \in \mathbb{R}^{n\times n}$ be positive definite, let $b \in \mathbb{R}^n$ and $c \in \mathbb{R}$, and define $f\colon \mathbb{R}^n \to \mathbb{R}$ by $f(x) \triangleq x^{\mathrm{T}}Ax + 2b^{\mathrm{T}}x + c$. Then, f has a unique stationary point, which is the global minimizer given by $\arg\min_{x\in\mathbb{R}^n} f(x) = -A^{-1}b$.*

Proof of Theorem 1. We write $J_0(\hat{x})$, given by (17), as

$$J_0(\hat{x}) = \|y_0 - C_0\hat{x}\|^2_{\Gamma_0^{-1}} + \|A_0^{-1}(\hat{x} - B_0u_0) - \hat{x}_0\|^2_{P_0^{-1}-F_0},$$

$$= \|y_0 - C_0\hat{x}\|^2_{\Gamma_0^{-1}} + \|\hat{x} - (A_0\hat{x}_0 + B_0u_0)\|^2_{A_0^{-\mathrm{T}}(P_0^{-1}-F_0)A_0^{-1}}$$

Next, we expand $J_0(\hat{x}) = \hat{x}^{\mathrm{T}}H_0\hat{x} + 2b_0^{\mathrm{T}}\hat{x} + c_0$, where

$$H_0 \triangleq C_0^{\mathrm{T}}\Gamma_0^{-1}C_0 + A_0^{-\mathrm{T}}(P_0^{-1} - F_0)A_0^{-1} \tag{41}$$

$$b_0 \triangleq -C_0^{\mathrm{T}}\Gamma_0^{-1}y_0 - A_0^{-\mathrm{T}}(P_0^{-1} - F_0)A_0^{-1}(A_0\hat{x}_0 + B_0u_0)$$

$$c_0 \triangleq y_0^{\mathrm{T}}\Gamma_0^{-1}y_0 + \|A_0^{-1}(A_0\hat{x}_0 + B_0u_0)\|^2_{P_0^{-1}-F_0}$$

Defining $P_1 \triangleq H_0^{-1}$, it follows from the expression of H_0 that (21) is satisfied for $k = 0$. Furthermore, if $k = 0$, (16) simplifies to $F_0 \prec P_0^{-1}$ and hence $P_0^{-1} - F_0 \succ 0$. Furthermore, since Γ_0 is positive definite, it follows from definition (41) that H_0 is positive definite. Therefore, from Lemma 2, it follows that the unique minimizer $\hat{x}_1$ of J_0 is given by $\hat{x}_1 = -H_0^{-1}b_0$, which can be expanded as

$$\hat{x}_1 = P_1\big[C_0^{\mathrm{T}}\Gamma_0^{-1}y_0 + A_0^{-\mathrm{T}}(P_0^{-1} - F_0)A_0^{-1}(A_0\hat{x}_0 + B_0u_0)\big]$$

$$
\begin{aligned}
=&P_1\big[C_0^{\mathrm T}\Gamma_0^{-1}y_0 - C_0^{\mathrm T}\Gamma_0^{-1}C_0(A_0\hat x_0 + B_0u_0)\\
&+ (A_0^{-\mathrm T}(P_0^{-1} - F_0)A_0^{-1} + C_0^{\mathrm T}\Gamma_0^{-1}C_0)(A_0\hat x_0 + B_0u_0)\big]\\
=&P_1\big[C_0^{\mathrm T}\Gamma_0^{-1}y_0 - C_0^{\mathrm T}\Gamma_0^{-1}C_0(A_0\hat x_0 + B_0u_0)\\
&+ P_1^{-1}(A_0\hat x_0 + B_0u_0)\big]\\
=&A_0\hat x_0 + B_0u_0 + P_1C_0^{\mathrm T}\Gamma_0^{-1}(y_0 - C_0(A_0\hat x_0 + B_0u_0)).
\end{aligned}
$$

Hence, (22) holds for $k = 0$. Moreover, since H_0 is positive definite, it follows that $P_1 = H_0^{-1}$ is also positive definite.

Now, let $k \geq 1$. Note that for all $0 \leq i \leq k$,

$$
\begin{aligned}
(\mathcal T_{i,k+1}(\hat x) - \hat x_i) &= \Phi_{i,k+1}(\hat x - \mathcal B_{k+1,i}\mathcal U_{k+1,i}) - \hat x_i\\
&= \Phi_{i,k+1}(\hat x - (\Phi_{k+1,i}\hat x_i + \mathcal B_{k+1,i}\mathcal U_{k+1,i}))\\
&= \Phi_{i,k+1}(\hat x - \mathcal T_{k+1,i}(\hat x_i)).
\end{aligned}
$$

Hence, $J_k(\hat x)$, given by (17), can be written as

$$
J_k(\hat x) = \|\Phi_{0,k+1}(\hat x - \mathcal T_{k+1,0}(\hat x_0))\|^2_{P_0^{-1}} +
$$

$$
\sum_{i=0}^{k} \|y_i - C_i\mathcal T_{i+1,k+1}(\hat x)\|^2_{\Gamma_i^{-1}} - \|\Phi_{i,k+1}(\hat x - \mathcal T_{k+1,i}(\hat x_i))\|_{F_i}.
$$

Next, we expand $J_k(\hat x) = \hat x^{\mathrm T}H_k\hat x + 2b_k^{\mathrm T}\hat x + c_k$, where

$$
\begin{aligned}
H_k \triangleq& \sum_{i=0}^{k}\Big[\Phi_{i+1,k+1}^{\mathrm T}C_i^{\mathrm T}\Gamma_i^{-1}C_i\Phi_{i+1,k+1} - \Phi_{i,k+1}^{\mathrm T}F_i\Phi_{i,k+1}\Big]\\
&+ \Phi_{0,k+1}^{\mathrm T}P_0^{-1}\Phi_{0,k+1}, \qquad (42)\\
b_k \triangleq& \sum_{i=0}^{k}\Big[\Phi_{i,k+1}^{\mathrm T}F_i\Phi_{i,k+1}\mathcal T_{k+1,i}(\hat x_i)\\
&- \Phi_{i+1,k+1}^{\mathrm T}C_i^{\mathrm T}\Gamma_i^{-1}(y_i + C_i\Phi_{i+1,k+1}\mathcal B_{k+1,i+1}\mathcal U_{k+1,i+1})\Big]\\
&- \Phi_{0,k+1}^{\mathrm T}P_0^{-1}\Phi_{0,k+1}\mathcal T_{k+1,0}(\hat x_0), \qquad (43)\\
c_k \triangleq& \sum_{i=0}^{k}\Big[\|y_i + C_i\Phi_{i+1,k+1}\mathcal B_{k+1,i+1}\mathcal U_{k+1,i+1}\|^2_{\Gamma_i}\\
&- \|\Phi_{i,k+1}\mathcal T_{k+1,i}(\hat x_i)\|^2_{F_i}\Big] + \|\Phi_{0,k+1}\mathcal T_{k+1,0}(\hat x_i)\|^2_{P_0^{-1}}.
\end{aligned}
$$

Note that H_k can be written recursively as

$$
\begin{aligned}
H_k =& \sum_{i=0}^{k-1}\big[\Phi_{i+1,k+1}^{\mathrm T}C_i^{\mathrm T}\Gamma_i^{-1}C_i\Phi_{i+1,k+1} - \Phi_{i,k+1}^{\mathrm T}F_i\Phi_{i,k+1}\big]\\
&+ \Phi_{0,k+1}^{\mathrm T}P_0^{-1}\Phi_{0,k+1} + C_k^{\mathrm T}\Gamma_k^{-1}C_k - A_k^{-\mathrm T}F_kA^{-1}\\
=&A_k^{-\mathrm T}\Big[\sum_{i=0}^{k-1}\big(\Phi_{i+1,k}^{\mathrm T}C_i^{\mathrm T}\Gamma_i^{-1}C_i\Phi_{i+1,k} - \Phi_{i,k}^{\mathrm T}F_i\Phi_{i,k}\big)\\
&+ \Phi_{0,k}^{\mathrm T}P_0^{-1}\Phi_{0,k}\Big]A_k^{-1} + C_k^{\mathrm T}\Gamma_k^{-1}C_k - A_k^{-\mathrm T}F_kA^{-1}\\
=&A_k^{-\mathrm T}(H_{k-1} - F_k)A_k^{-1} + C_k^{\mathrm T}\Gamma_k^{-1}C_k. \qquad (44)
\end{aligned}
$$

Defining $P_{k+1} \triangleq H_k^{-1}$, it follows that (21) is satisfied.

Next, to write a recursive update for b_k, we first write b_k as $b_k = b_{k,1} + b_{k,2} + b_{k,3}$, where

$$
\begin{aligned}
b_{k,1} \triangleq& \sum_{i=0}^{k-1}\Big[\Phi_{i,k+1}^{\mathrm T}F_i\Phi_{i,k+1}\mathcal T_{k+1,i}(\hat x_i)\\
&- \Phi_{i+1,k+1}^{\mathrm T}C_i^{\mathrm T}\Gamma_i^{-1}(y_i + C_i\Phi_{i+1,k+1}\mathcal B_{k+1,i+1}\mathcal U_{k+1,i+1})\Big],\\
b_{k,2} \triangleq& -\Phi_{0,k+1}^{\mathrm T}P_0^{-1}\Phi_{0,k+1}\mathcal T_{k+1,0}(\hat x_0)\\
b_{k,3} \triangleq& A_k^{-\mathrm T}F_kA_k^{-1}(A_k\hat x_k + B_ku_k) - C_k^{\mathrm T}\Gamma_k^{-1}y_k.
\end{aligned}
$$

Note that $b_{k,1}$ is the sum of first $k-1$ terms of summation in (43), $b_{k,3}$ is the last term of summation in (43), and $b_{k,2}$ is the remaining term outside the summation. Next, note, for all $0 \leq i \leq k$, the identities

$$
\begin{aligned}
&\Phi_{i,k+1}\mathcal T_{k+1,i}(\hat x_i) = \Phi_{i,k}\mathcal T_{k,i}(\hat x_i) + \Phi_{i,k+1}B_ku_k, \qquad (45)\\
&\Phi_{i+1,k+1}\mathcal B_{k+1,i+1}\mathcal U_{k+1,i+1}\\
&\qquad = \Phi_{i+1,k}\mathcal B_{k,i+1}\mathcal U_{k,i+1} + \Phi_{i+1,k+1}B_ku_k. \qquad (46)
\end{aligned}
$$

Using (45) and (46), we can write $b_{k,1} = b_{k,1,1} + b_{k,1,2}$, where

$$
\begin{aligned}
b_{k,1,1} \triangleq& \sum_{i=0}^{k-1}\big[\Phi_{i,k+1}^{\mathrm T}F_i\Phi_{i,k}\mathcal T_{k,i}(\hat x_i)\\
&- \Phi_{i+1,k+1}^{\mathrm T}C_i^{\mathrm T}\Gamma_i^{-1}(y_i + C_i\Phi_{i+1,k}\mathcal B_{k,i+1}\mathcal U_{k,i+1})\big],\\
b_{k,1,2} \triangleq& \sum_{i=0}^{k-1}\big[\Phi_{i,k+1}^{\mathrm T}F_i\Phi_{i,k+1}B_ku_k\\
&- \Phi_{i+1,k+1}^{\mathrm T}C_i^{\mathrm T}\Gamma_i^{-1}C_i\Phi_{i+1,k+1}B_ku_k\big].
\end{aligned}
$$

Similarly, (45) implies that $b_{k,2} = b_{k,2,1} + b_{k,2,2}$, where

$$
\begin{aligned}
b_{k,2,1} &\triangleq -\Phi_{0,k+1}^{\mathrm T}P_0^{-1}\Phi_{0,k}\mathcal T_{k,0}(\hat x_0),\\
b_{k,2,2} &\triangleq -\Phi_{0,k+1}^{\mathrm T}P_0^{-1}\Phi_{0,k+1}B_ku_k.
\end{aligned}
$$

It then follows from (43) and (42), respectively, that

$$
\begin{aligned}
b_{k,1,1} + b_{k,2,1} &= A_k^{-\mathrm T}b_{k-1},\\
b_{k,1,2} + b_{k,2,2} &= -A_k^{-\mathrm T}H_{k-1}A_k^{-1}B_ku_k.
\end{aligned}
$$

Hence, we obtain the recursive update

$$
\begin{aligned}
b_k =& A_k^{-\mathrm T}b_{k-1} - A_k^{-\mathrm T}H_{k-1}A_k^{-1}B_ku_k\\
&+ A_k^{-\mathrm T}F_kA_k^{-1}(A_k\hat x_k + B_ku_k) - C_k^{\mathrm T}\Gamma_k^{-1}y_k.
\end{aligned}
$$

Finally, note that (42) can be used to rewrite (16) as $F_k \prec H_{k-1}$, and hence $H_{k-1} - F_k \succ 0$. Furthermore, since Γ_k is positive definite, it follows from (44) that H_k is positive definite. Therefore, by Lemma 2, the unique minimizer $\hat x_{k+1}$ of J_k is given by $\hat x_{k+1} = -H_k^{-1}b_k$, which simplifies to

$$
\begin{aligned}
\hat x_{k+1} =& -H_k^{-1}b_k = -P_{k+1}b_k\\
=& P_{k+1}\big[-A_k^{-\mathrm T}b_{k-1} + A_k^{-\mathrm T}H_{k-1}A_k^{-1}B_ku_k\\
&- A_k^{-\mathrm T}F_kA_k^{-1}(A_k\hat x_k + B_ku_k) + C_k^{\mathrm T}\Gamma_k^{-1}y_k\big]\\
=& P_{k+1}\big[A_k^{-\mathrm T}P_k^{-1}\hat x_k + A_k^{-\mathrm T}P_k^{-1}A_k^{-1}B_ku_k\\
&- A_k^{-\mathrm T}F_kA_k^{-1}(A_k\hat x_k + B_ku_k) + C_k^{\mathrm T}\Gamma_k^{-1}y_k\big]\\
=& P_{k+1}\big[A_k^{-\mathrm T}(P_k^{-1} - F_k)A_k^{-1}(A_k\hat x_k + B_ku_k) + C_k^{\mathrm T}\Gamma_k^{-1}y_k\big]\\
=& P_{k+1}\big[C_k^{\mathrm T}\Gamma_k^{-1}y_k - C_k^{\mathrm T}\Gamma_k^{-1}C_k(A_k\hat x_k + B_ku_k)\\
&+ \big(A_k^{-\mathrm T}(P_k^{-1} - F_k)A_k^{-1} + C_k^{\mathrm T}\Gamma_k^{-1}C_k\big)(A_k\hat x_k + B_ku_k)\big]\\
=& P_{k+1}\big[C_k^{\mathrm T}\Gamma_k^{-1}(y_k - C_k(A_k\hat x_k + B_ku_k))\\
&+ P_{k+1}^{-1}(A_k\hat x_k + B_ku_k)\big]\\
=& A_k\hat x_k + B_ku_k + P_{k+1}C_k^{\mathrm T}\Gamma_k^{-1}(y_k - C_k(A_k\hat x_k + B_ku_k)).
\end{aligned}
$$

Hence, (22) is satisfied. Finally, since H_k is positive definite, it follows that $P_{k+1} = H_k^{-1}$ is also positive definite. □