FIR implementation of the Steady-State Kalman Filter

Nicholas Assimakis*

Department of Electronics,
Technological Educational Institute of Lamia,
35100 Lamia, Greece
E-mail: assimakis@teilam.gr
*Corresponding author

Maria Adam

Department of Computer Science and Biomedical Informatics,
University of Central Greece,
35100 Lamia, Greece
E-mail: madam@ucg.gr

Abstract: In this paper, an FIR implementation for the Steady-State Kalman Filter (SSKF) is proposed. The method requires the knowledge of a subset of previous time measurements. The proposed algorithm is faster than the classical one, especially for large estimation time. The proposed implementation is also applicable to periodic models.

Keywords: KF; Kalman filter; steady state; periodic model; FIR-filters; matrix theory.

Reference to this paper should be made as follows: Assimakis, N. and Adam, M. (2008) 'FIR implementation of the Steady-State Kalman Filter', *Int. J. Signal and Imaging Systems Engineering*, Vol. 1, Nos. 3/4, pp.279–286.

Biographical notes: N. Assimakis received his PhD in Computer Engineering and Informatics from University of Patras in 1993. His academic position is Assistant Professor at the Department of Electronics, Technological Educational Institute of Lamia, Greece. His research interests include estimation theory, Kalman filter, Riccati equation, digital signal processing, algorithms, numerical analysis.

M. Adam received her PhD in Mathematics from National Technical University of Athens in 2001. Her current position is Visiting Assistant Professor at the University of Central Greece. Her research interests include linear algebra, matrix analysis, numerical range, graphs and statistic models associated with matrices, numerical linear algebra, applications in dynamical systems and control theory.

1 Introduction

Estimation plays an important role in many fields of science. The discrete-time Kalman Filter (KF) (Kalman, 1960) is the most well-known algorithm that solves the estimation/filtering problem. Many real-world problems have been successfully solved using the Kalman filter ideas; filter applications to aerospace industry, chemical process, communication systems design, control, civil engineering, filtering noise from two-dimensional images, pollution prediction and power systems are mentioned in Anderson and Moore (1979). The SSKF is briefly presented and an FIR implementation is proposed and applied to periodic models. The proposed algorithm is faster than the classical implementation.

2 Steady-State Kalman Filter

The estimation problem arises in linear estimation and is associated with time-varying systems described by the following state-space equations:

$$x(k+1) = F(k+1, k)x(k) + w(k) \tag{1}$$

$$z(k+1) = H(k+1)x(k+1) + v(k+1) \tag{2}$$

where $x(k)$ is the n-dimensional state vector at time k, $z(k)$ is the m-dimensional measurement vector, $F(k+1, k)$ is the $n \times n$ system transition matrix, $H(k+1)$ is the $m \times n$ output matrix, $\{w(k)\}$ and $\{v(k)\}$ are the plant noise and the measurement noise processes; they are assumed to be Gaussian, zero-mean white and uncorrelated random processes and $Q(k)$, $R(k+1)$ are the plant noise and measurement noise covariance matrices, respectively. The vector $x(0)$ is a Gaussian random process with mean x_0 and covariance P_0 and $x(0)$, $\{w(k)\}$ and $\{v(k)\}$ are independent.

The filtering/estimation problem is to produce an estimate at time L of the state vector using measurements till time L, i.e. the aim is to use the measurements set $\{z(1), \ldots, z(L)\}$ to calculate an estimate value $x(L/L)$ of the state vector $x(L)$. The discrete-time KF is summarised in the following:

2.1 Kalman Filter (KF)

$$K(k) = P(k/k-1)H^T(k) \cdot [H(k)P(k/k-1)H^T(k) + R(k)]^{-1} \quad (3)$$

$$P(k/k) = [I - K(k)H(k)]P(k/k-1) \quad (4)$$

$$P(k+1/k) = Q(k) + F(k+1, k)P(k/k)F^T(k+1, k) \quad (5)$$

$$x(k/k) = [I - K(k)H(k)]\, x(k/k-1) + K(k)z(k) \quad (6)$$

$$x(k+1/k) = F(k+1, k)x(k/k) \quad (7)$$

for $k = 0, 1, \ldots,$ with initial conditions $x(0/-1) = x_0$ and $P(0/-1) = P_0$, where $x(k/k)$ is the estimate value of the state vector at time k, $P(k/k)$ is the corresponding estimation error covariance matrix, $K(k)$ is the KF gain, $x(k + 1/k)$ is the prediction value of the state vector at time k, $P(k + 1/k)$ is the corresponding prediction error covariance matrix. All basic terms involved in the KF equations are defined in Appendix A.

For *time-invariant systems* where the system transition matrix, the output matrix, the plant and measurement noise covariance matrices are constant, the resulting Time-Invariant Kalman Filter (TIKF) takes the following form:

2.2 Time-Invariant Kalman Filter (TIKF)

$$x(k+1/k) = Fx(k/k) \quad (8)$$

$$P(k+1/k) = FP(k/k)F^T + Q \quad (9)$$

$$K(k+1) = P(k+1/k)H^T[HP(k+1/k)H^T + R]^{-1} \quad (10)$$

$$x(k+1/k+1) = x(k+1/k) + K(k+1) \cdot [z(k+1) - Hx(k+1/k)] \quad (11)$$

$$P(k+1/k+1) = P(k+1/k) - K(k+1)HP(k+1/k). \quad (12)$$

For time-invariant systems, it is well known (Kalman, 1960) that if the signal process model is asymptotically stable, then there exists a steady-state value $\overline{P}$ of the prediction error covariance matrix, which is reached at time $k = s$.

In this case, the resulting discrete-time SSKF takes the following form:

2.3 Steady-State Kalman Filter (SSKF)

$$x(s+k+1/s+k+1) = Ax(s+k/s+k) + Bz(s+k+1) \quad (13)$$

where

$$A = [I - \overline{K}H]F \quad (14)$$

$$B = \overline{K} \quad (15)$$

Remarks

1 SSKF requires the implementation of TIKF for s recursions, i.e., until the steady-state time $k = s$ is reached.

2 SSKF is a recursive algorithm: the calculation of the estimate at time $s + k + 1$ requires the estimate at the previous time $s + k$.

3 SSKF has a structure of an IIR filter.

4 The matrices A and B can be calculated off-line: first the corresponding discrete-time Riccati equation (Anderson and Moore, 1979)

$$P(k+1/k) = FP(k/k-1)F^T + Q - FP(k/k-1)H^T \cdot [HP(k/k-1)H^T + R]^{-1} HP(k/k-1)F^T \quad (16)$$

is solved and then the steady-state gain $\overline{K}$ is

$$\overline{K} = \overline{P}H^T[H\overline{P}H^T + R]^{-1}. \quad (17)$$

3 FIR Steady-State Kalman Filter

In Higham and Knight (1995), it is established that "if the spectral radius of a matrix A is less than 1, then the computed powers of A can be expected to converge to zero"; thus, considering that all eigenvalues of F lie inside the unit circle, we have (Assimakis et al., 2003) that the matrix A in equation (14) has the following important property:

$$A^k \xrightarrow[k\to\infty]{} 0. \quad (18)$$

Owing to the computer accuracy, this property of matrix A leads to the conclusion (Assimakis et al., 2003) that there exists some v, such that:

$$A^{\nu} \neq 0 \text{ and } A^{\nu+i} = 0, \quad i = 1, 2, \ldots \quad (19)$$

Thus, the following implementation of the SSKF is derived (Assimakis et al., 2003).

3.1 FIR Steady-State Kalman Filter (FIR-SSKF)

$$x(s+\nu+k/s+\nu+k) = \sum_{j=k}^{\nu+k} c(j)z(s+j) \quad (20)$$

where there exists some v as in equation (19) and

$$c(j) = A^{\ell+k-j}\overline{K}, \quad j = k, \ldots, \nu+k. \quad (21)$$

Remarks

1 FIR-SSKF requires first the implementation of TIKF until the steady-state time is reached and subsequently the implementation of SSKF for v recursions.

2 FIR-SSKF is not a recursive algorithm: there is no need for any previous estimates calculation. The method requires the knowledge of a subset of $v + 1$ previous time measurements to calculate the state estimate. The number of the needed previous time measurements is a-priori known (i.e., before the implementation of the filter): v can be determined off-line.

3 FIR-SSKF has a structure of an FIR filter.

4 The steady-state prediction error covariance matrix is calculated by off-line solving the corresponding discrete-time Riccati equation (16) using suitable techniques (Assimakis et al., 1997; Lainiotis, 1975; Lainiotis et al., 1994). Then, the steady-state gain and the matrix A are calculated off-line using equations (17) and (14). Finally, v in equation (19) and the coefficients $c(j)$ in equation (21) are calculated off-line.

It is obvious that v in equation (19) can be determined with respect to the desired accuracy. In fact, instead of equation (19), we are able to use:

$$A^{v} > \varepsilon \quad \text{and} \quad A^{v+i} \le \varepsilon, \quad i = 1, 2, ... \tag{22}$$

where ε is a small positive real number.

The behaviour of the proposed implementation is presented in the following examples.

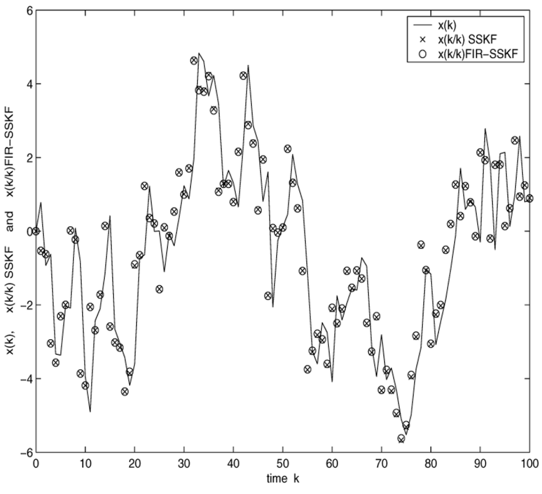
Example 1: Let us assume a simple scalar ($n = 1$ and $m = 1$) time-invariant model: $F = 0.8$, $H = 1$, $Q = 2$ and $R = 0.1$ with initial conditions

$$x(0/-1) = x_0 = 0 \ \text{ and } \ P(0/-1) = P_0 = 0.$$

The steady-state time is reached after the implementation of TIKF for $s = 7$ recursions.

The state $x(k)$ and the calculated estimates $x(k/k)$ are plotted in Figure 1, using SSKF and FIR-SSKF for $v = 5$. The calculated estimates using SSKF and FIR-SSKF are very close to each other.

Figure 1 State $x(k)$ and estimates $x(k/k)$ using SSKF and FIR-SSKF



Example 2: Using Monte Carlo simulation, the estimates are calculated for time $k = 1, \ldots, 100$ by implementing SSKF and FIR-SSKF for various values of ε. For 100 Monte Carlo runs, the mean absolute error decreases as the accuracy increases, as shown in Table 1.

Table 1 FIR-SSKF for different desired accuracy: mean absolute error

Accuracy		
ε	v	*Mean absolute error*
10^{-6}	5	0.0160
10^{-8}	6	0.0159
10^{-12}	9	0.0151
10^{-16}	12	0.0146

4 FIR Steady-State Periodic Kalman Filter

The proposed implementation is also applicable to periodic models.

In the case of *periodic model*, the matrices $F(k + 1, k)$, $H(k + 1)$, $Q(k)$ and $R(k + 1)$ are periodic with period p, i.e.:

$$F(k+1+ip, k+ip) = F(k+1+(i+1)p, k+(i+1)p) \tag{23}$$

$$H(k+1+ip) = H(k+1+(i+1)p) \tag{24}$$

$$Q(k+ip) = Q(k+(i+1)p) \tag{25}$$

$$R(k+1+ip) = R(k+1+(i+1)p) \tag{26}$$

for $k = 0, 1, \ldots, p - 1$ and $i = 0, 1, \ldots$

The corresponding discrete-time periodic Riccati equation has as follows:

$$\begin{aligned} P(k+1/k) = {} & Q(k) + F(k+1, k)P(k/k-1)F^T(k+1, k) \\ & - F(k+1, k)P(k/k-1)H^T(k) \\ & \cdot [H(k)P(k/k-1)H^T(k) + R(k)]^{-1} \\ & \cdot H(k)P(k/k-1)F^T(k+1, k). \end{aligned} \tag{27}$$

It is known (Varga, 2005) for periodic systems that the discrete-time periodic Riccati equation has a steady-state periodic stabilising solution with period p:

$$\bar{P}(k+1+ip/k+ip) = \bar{P}(k+1+(i+1)p/k+(i+1)p). \tag{28}$$

Then, it is obvious that the Kalman filter gain matrix $K(k)$ becomes periodic with period p:

$$\bar{K}(k+1+ip) = \bar{K}(k+1+(i+1)p) \tag{29}$$

where

$$\bar{K}(k) = \bar{P}(k/k-1)H^T(k)[H(k)\bar{P}(k/k-1)H^T(k) + R(k)]^{-1}. \tag{30}$$

Combining equations (6) and (7), we are able to write:

$$x(k+1/k+1) = A(k+1, k)x(k/k) + K(k+1)z(k+1) \tag{31}$$

where

$$A(k+1,k)=[I-K(k+1)H(k+1)]F(k+1,k). \qquad (32)$$

We observe that the matrix $A(k+1, k)$ becomes periodic with period p:

$$\bar{A}(k+1+ip,k+ip)=\bar{A}(k+1+(i+1)p,k+(i+1)p) \qquad (33)$$

where

$$\bar{A}(k+1,k)=F(k+1,k)-\bar{K}(k+1)H(k+1)F(k+1,k). \qquad (34)$$

Thus, in the steady-state case, after the steady-state time is reached in s periods, the resulting discrete-time Steady-State Periodic Kalman Filter (SSPKF) has as follows:

4.1 Steady-State Periodic Kalman Filter (SSPKF)

$$\begin{aligned} x(sp+k+1/sp+k+1) &= \bar{A}(k \bmod p+1, k \bmod p)x(sp+k/sp+k) \\ &\quad +\bar{K}(k \bmod p+1)z(sp+k+1) \end{aligned} \qquad (35)$$

for $k = 0, 1, \ldots$

Remarks

1. SSPKF implementation requires the KF implementation for $k = 0, \ldots, sp$ to calculate the estimate $x(sp/sp)$.
2. The steady-state periodic prediction error covariance matrix $\bar{P}(k+1,k)$ is calculated by off-line solving the corresponding discrete-time periodic Riccati equation. Then, the steady-state periodic gain matrix $\bar{K}(k+1)$ and the corresponding matrix $\bar{A}(k+1,k)$ are calculated off-line using equations (30) and (34), respectively.

We define the matrix

$$\bar{A}_r^i=\begin{cases}\bar{A}(i,i-1)\cdot\bar{A}(i-1,i-2)\cdot\ldots\cdot\bar{A}(r,r-1), & i\ge r\\ I, & i<r\end{cases}$$

Note that if all the singular values of $\bar{A}(1,0),\bar{A}(2,1),\ldots,\bar{A}(p,p-1)$ lie inside the unit circle, then it follows that all eigenvalues of the product matrix $\bar{A}_1^P=\bar{A}(p,p-1)\cdot\ldots\cdot\bar{A}(2,1)\cdot\bar{A}(1,0)$ lie also inside the unit circle; thus, considering that $\bar{A}_1^P$ has spectral radius less than 1 (i.e., all eigenvalues of $\bar{A}_1^P$ lie inside the unit circle), we have that the computed powers of $\bar{A}_1^P$ can be expected to converge to zero, working as in Assimakis et al. (2003), concluding that there exists some v, such that:

$$[\bar{A}_1^p]^{\nu}\neq 0 \text{ and } [\bar{A}_1^p]^{\nu+i}=0,\quad i=1,2,\ldots \qquad (36)$$

Using equation (35) after the steady-state time is reached in s periods and taking advantage of the periodicity of the gain matrix $\bar{K}(k+1)$ and the corresponding matrix $\bar{A}(k+1,k)$, we obtain the following estimates per period time (p lags) using a double sum:

$$\begin{aligned} &x(sp+\nu p+\mu p/sp+\nu p+\mu p)\\ &\quad=\sum_{i=1}^{p}\sum_{j=1}^{\nu+1}c(i,j)z(sp+jp+\mu p-2p+i) \end{aligned} \qquad (37)$$

where

$$c(i,j)=[\bar{A}_1^p]^{\nu-j+1}\bar{A}_{i+1}^p\bar{K}(i), \qquad (38)$$

for $i = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, v + 1$, and some v such that as in equation (36).

Moreover, analogous estimates concerning each time φ in a period time ($\varphi = 1, 2, \ldots, p$) can be written, leading to a generalisation of the expression in equation (37):

$$\begin{aligned} &x(sp+\nu p+\mu p+\varphi/sp+\nu p+\mu p+\varphi)\\ &\quad=\sum_{i=1}^{p}\sum_{j=1}^{\nu+1}\hat{c}(i,j)z(sp+\mu p+jp-p+i) \end{aligned} \qquad (39)$$

for $1 \le \varphi \le p$, where

$$\hat{c}(i,j)=\begin{cases}\bar{A}_1^{\varphi}[\bar{A}_1^p]^{\nu-j}\bar{A}_{i+1}^p\bar{K}(i), & 1\le i\le p,\quad 1\le j\le\nu\\ \bar{A}_{i+1}^{\varphi}\bar{K}(i), & 1\le i\le\varphi,\quad j=\nu+1\\ 0, & i>\varphi,\quad j=\nu+1.\end{cases} \qquad (40)$$

Remarks

1. There is no need for any previous estimates calculation. The calculation of the steady-state estimate $x(L)$ at some time L requires the use of the subset of $p(v + 1)$ previous time measurements.
2. The steady-state periodic prediction error covariance matrix $\bar{P}(k+1,k)$ is calculated by off-line solving the corresponding discrete-time periodic Riccati equation. Then, the steady-state periodic gain matrix $\bar{K}(k+1)$ and the corresponding matrix $\bar{A}(k+1,k)$ are calculated off-line using equations (30) and (34), respectively. Finally, v in equation (36) and the coefficients $\hat{c}(i,j)$ in equation (40) are calculated off-line.
3. This algorithm is a generalisation of the algorithm concerning time-invariant systems. In fact, in the special case for $p = 1$ and $\varphi = 1$, we derive the FIR-SSKF, i.e., the equations (20)–(21).
4. This algorithm uses a double sum.

Using a single sum instead of a double sum, we are able to write equation (37) and the corresponding coefficients in equation (38) as:

$$\begin{aligned} &x(sp+\nu p+\mu p/sp+\nu p+\mu p)\\ &\quad=\sum_{k=1}^{p(\nu+1)}c(k)z(sp+\mu p-p+k) \end{aligned} \qquad (41)$$

where

$$c(k) = [\bar{A}_1^p]^{\nu-((k-1)\mathrm{div}p)} \bar{A}^p_{((k-1)\bmod p)+2}\bar{K}((k-1)\bmod p+1) \quad (42)$$

obtaining estimates per period time (p lags). The proof is given in detail in Appendix B.

Moreover, we prove in Appendix C that analogous estimates concerning each time φ in a period time ($\varphi = 1, 2, \ldots, p$), which are formed in the following:

$$x(sp+\nu p+\mu p+\varphi / sp+\nu p+\mu p+\varphi) = \sum_{k=1}^{p(\nu+1)+\varphi} \hat{c}(k) z(sp+\mu p-p+k) \quad (43)$$

where

$$\hat{c}(k) = \begin{cases} \bar{A}_1^\varphi c(k) = \bar{A}_1^\varphi [\bar{A}_1^p]^{\nu-((k-1)\mathrm{div}p)} \bar{A}^p_{((k-1)\bmod p)+2} \\ \quad \cdot \bar{K}((k-1)\bmod p+1), 1 \le k \le p(\nu+1) \\ \bar{A}^\varphi_{((k-1)\bmod p)+2}\bar{K}((k-1)\bmod p+1), \\ \quad p(\nu+1)+1 \le k \le \varphi + p(\nu+1). \end{cases} \quad (44)$$

Note that, for $1 \le k \le \varphi$ yields $(k-1)\mathrm{div}\, p = 0$, and $1 \le (k-1) \bmod p + \le \varphi$, and denoting $\tau = (k-1)\bmod p + 1$, the corresponding coefficients, $\hat{c}(k)$, are written as:

$$\begin{aligned} \hat{c}(k) &= \bar{A}_1^\varphi [\bar{A}_1^p]^{\nu-((k-1)\mathrm{div}\, p)} \bar{A}^p_{((k-1)\bmod p)+2}\bar{K}((k-1)\bmod p+1) \\ &= \bar{A}_1^\varphi [\bar{A}_1^p]^\nu \bar{A}^p_{\tau+1}\bar{K}(\tau) \\ &= \bar{A}^\varphi_{\tau+1}\left[\bar{A}_1^\tau \bar{A}^p_{\tau+1}\right]^{\nu+1} \bar{K}(\tau). \end{aligned}$$

Owing to the known property that "the eigenvalues of the matrix $A \cdot B$ are the same as those of the matrix $B \cdot A$", the spectral radius of $\left[\bar{A}_1^\tau \bar{A}^p_{\tau+1}\right], \tau = 1,\ldots,\varphi$ is less than 1, thus, as in equation (36), the computed powers of $\left[\bar{A}_1^\tau \bar{A}^p_{\tau+1}\right], \tau = 1,\ldots,\varphi$ can be expected to converge to zero; whereby we conclude that $\hat{c}(k) \to 0$, $k = 1, 2, \ldots, \varphi$, $1 \le \varphi \le p$.

Moreover, for $p(\nu+1)+1 \le k \le p(\nu+1)+\varphi$, we take $(k-1)\mathrm{div}\, p = \nu+1$ and $0 \le (k-1)\bmod p \le \varphi - 1$, thus

$$\bar{A}_1^\varphi [\bar{A}_1^p]^{\nu-((k-1)\mathrm{div}\, p)} \bar{A}^p_{((k-1)\bmod p)+2} = \bar{A}^\varphi_{((k-1)\bmod p)+2}.$$

Consequently, substituting in equations (43) and (44), we derive the following implementation of the SSPKF.

4.2 *FIR Steady-State Periodic Kalman Filter (FIR-SSPKF)*

$$x(sp+\nu p+\mu p+\varphi / sp+\nu p+\mu p+\varphi) = \sum_{k=\varphi+1}^{p(\nu+1)+\varphi} \hat{c}(k) z(sp+\mu p-p+k) \quad (45)$$

where

$$\begin{aligned} \hat{c}(k) &= [\bar{A}_1^\varphi] c(k) \\ &= [\bar{A}_1^\varphi][\bar{A}_1^p]^{\nu-((k-1)\mathrm{div}\, p)} \bar{A}^p_{((k-1)\bmod p)+2} \\ &\quad \bar{K}((k-1)\bmod p+1). \end{aligned} \quad (46)$$

Remarks

1 This algorithm uses a single sum.

2 This implementation of the SSPKF has a structure of an FIR filter.

5 Computational comparison

The two SSKF algorithms presented earlier calculate estimates of the state vector very close to each other; they calculate theoretically the same estimates for large enough v. Thus, to compare the algorithms with respect to their computational time, we have to compare their calculation burdens required for the online calculations; the calculation burden of the off-line calculations (initialisation process) is not taken into account.

Scalar operations are involved in matrix manipulation operations, which are needed for the implementation of the SSKF algorithms. Table 2 summarises the calculation burden of needed matrix operations.

Table 2 Calculation burden of matrix operations

Matrix operation	*Multiplications*	*Additions*	*Calculation burden*
$(n \times m) \cdot (m \times k)$	nmk	$n(m-1)k$	$2nmk - nk$
$(n \times 1) + (n \times 1)$	–	n	n

The calculation burdens of both algorithms are equal to each other till time $s + v$, due to the facts that, both SSKF and FIR-SSKF require the implementation of TIKF for s recursions until the steady-state time is reached and that FIR-SSKF requires the implementation of SSKF for v more recursions. Thus, to compare the algorithms, we compute the estimate value $x(L/L)$ of the state vector $x(L)$ at some time $L = s + v + \mu$, with $\mu \ge 1$.

We make the following basic *remarks*

a SSKF is a recursive algorithm. Thus, the calculation burden of SSKF depends on the estimation time. It depends on the number of recursions: SSKF executes μ recursions after time $s + v$. In fact, the calculation burden increases as the number of recursions increases.

b FIR-SSKF is not a recursive algorithm. Thus, it calculates each estimate in the same time. The calculation burden of FIR-SSKF does not depend on the estimation time; it remains constant.

The calculation burdens of the SSKF algorithms are summarised in Table 3.

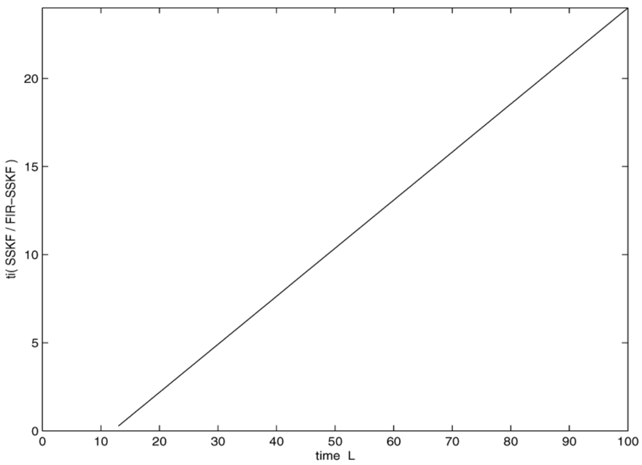
Table 3 Calculation burdens of the Steady-State Kalman Filter algorithms

Algorithm	*Calculation burden*
SSKF	$(2n^2 + 2nm - n)\mu$
FIR-SSKF	$(2nm - n)(v + 1) + nv$

The basic conclusion is that the calculation burden of SSKF increases as L increases, while the calculation burden of FIR-SSKF remains constant for all L. This conclusion is verified through the following simulation example.

Example 3: For the scalar ($n = 1$ and $m = 1$) time-invariant model of Example 1 in Section 3, the time improvement from SSKF to FIR-SSKF using $s = 7$, $v = 5$, $L = 13, 14, \ldots, 100$ is plotted in Figure 2. The proposed algorithm FIR-SSKF is faster than the classical one SSPKF, especially for large estimation time.

Figure 2 Time improvement from SSKF to FIR-SSKF



Analogous results are derived for the SSPKF algorithms.

To compare the algorithms, we compute the estimate value $x(L/L)$ of the state vector $x(L)$ at some time $L = sp + vp + \mu p + \varphi$.

The calculation burdens of both algorithms SSPKF and FIR-SSPKF are equal to each other till time $sp + vp$, due to the facts that they both require the implementation of KF until the steady-state time is reached and that the FIR-SSPKF requires the implementation of the SSPKF for v more recursions.

The calculation burdens of the SSPKF algorithms are summarised in Table 4.

Table 4 Calculation burdens of the Steady-State Periodic Kalman Filter algorithms

Algorithm	*Calculation burden*
SSPKF	$(2n^2 + 2nm - n)\, p(\mu + \varphi)$
FIR-SSPKF	$(2nm - n)(v + 1)p + n(pv + p - 1)$

Note that the calculation burden of SSPKF is an increasing function of μ and φ, while the calculation burden of FIR-SSPKF is constant (it depends on the state vector dimension n, on the measurement vector dimension m, on the period p and on the off-line calculated v). Thus, the proposed algorithm FIR-SSPKF is faster than the classical one SSPKF, especially for large estimation time.

6 Conclusions

A new approach for the SSKF is presented in this paper. The method is based on implementing the SSKF equations in a different way than the classical algorithm does and taking advantage of the finite computer precision. The method requires the knowledge of a subset of previous time measurements to calculate the state estimate; there is no need of any previous estimates calculation. The estimates provided by the proposed algorithm are very close to those provided by the classical algorithm. The implementation of the proposed algorithm is very attractive for large estimation time: the proposed algorithm is faster than the classical one; this is very important due to the fact that, in most real-time applications, it is essential to obtain the estimate in the shortest possible time. The proposed method is also applicable to periodic models.

References

Anderson, B.D.O. and Moore, J.B. (1979) *Optimal Filtering*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, USA.

Assimakis, N.D., Lainiotis, D.G., Katsikas, S.K. and Sanida, F.L. (1997) 'A survey of recursive algorithms for the solution of the discrete time Riccati equation', *Nonlinear Analysis, Theory, Methods and Applications*, Vol. 30, No. 4, pp.2409–2420.

Assimakis, N.D., Psarakis, E.Z. and Lainiotis, D.G. (2003) 'Steady state kalman filter: a new approach', *Neural, Parallel and Scientific Computations*, Vol. 11, pp.485–490.

Higham, N.J. and Knight, P.A. (1995) 'Matrix powers in finite precision arithmetic', *SIAM J. Matrix Anal. Appl.*, Vol. 16, No. 2, pp.343–358.

Kalman, R.E. (1960) 'A new approach to linear filtering and prediction problems', *J. Bas. Eng., Trans. ASME, Ser. D*, Vol. 82, No. 1, pp.34–45.

Lainiotis, D.G. (1975) 'Discrete Riccati equation solutions: partitioned algorithms', *IEEE Transactions on AC*, Vol. AC–20, August, pp.555–556.

Lainiotis, D.G., Assimakis, N.D. and Katsikas, S.K. (1994) 'A new computationally effective algorithm for solving the discrete Riccati equation', *Journal of Mathematical Analysis and Applications*, Vol. 186, No. 3, September, pp.868–895.

Varga, A. (2005) 'On solving discrete-time periodic Riccati Equations', *Proceedings of 16th IFAC World Congress 2005*, Prague, July, Th-E01-TP/9, http://www.nt.ntnu.no/users/skoge/prost/prooceedings/ifac2005/Topics/Topic6.html

Appendix A

Basic terms	
Term	*Definition*
$x(k)$	State vector
$z(k)$	Measurement vector
$F(k+1, k)$	Transition matrix
$H(k+1)$	Output matrix
$w(k)$	Plant noise
$v(k)$	Measurement noise
$Q(k)$	Plant noise covariance matrix
$R(k+1)$	Measurement noise covariance matrix
$x(k+1/k)$	State prediction
$P(k+1/k)$	Prediction error covariance matrix
$x(k/k)$	State estimate
$P(k/k)$	Estimation error covariance matrix
$K(k)$	Kalman filter gain
$\bar{P}$	Steady-state prediction error covariance matrix
$\bar{K}$	Steady-state Kalman filter gain

Appendix B

Proof of equations (41) and (42)

Developing the double sum in equation (37) we have:

$$\begin{aligned}
&x(sp+\nu p+\mu p/sp+\nu p+\mu p)\\
&=\sum_{i=1}^{p}\sum_{j=1}^{\nu+1}c(i,j)z(sp+jp+\mu p-2p+i)\\
&=\sum_{i=1}^{p}\{c(i,1)z(sp+\mu p+p-2p+i)\\
&\quad+c(i,2)z(sp+\mu p+2p-2p+i)\\
&\quad+c(i,3)z(sp+\mu p+3p-2p+i)\\
&\quad+\cdots+c(i,\nu)z(sp+\mu p+\nu p-2p+i)\\
&\quad+c(i,\nu+1)z(sp+\mu p+(\nu+1)p-2p+i)\}.
\end{aligned}$$

Substituting the coefficients $c(i, j)$ in equation (38), we take:

$$\begin{aligned}
&x(sp+\nu p+\mu p/sp+\nu p+\mu p)\\
&=\sum_{i=1}^{p}[\bar{A}_1^p]^{\nu}\,\bar{A}_{i+1}^p\bar{K}(i)z(sp+\mu p-p+i)\\
&\quad+\sum_{i=1}^{p}[\bar{A}_1^p]^{\nu-1}\,\bar{A}_{i+1}^p\bar{K}(i)z(sp+\mu p+i)\\
&\quad+\sum_{i=1}^{p}[\bar{A}_1^p]^{\nu-2}\,\bar{A}_{i+1}^p\bar{K}(i)z(sp+\mu p+p+i)\\
&\quad+\cdots+\sum_{i=1}^{p}[\bar{A}_1^p]\bar{A}_{i+1}^p\bar{K}(i)z(sp+\mu p+(\nu-2)p+i)\\
&\quad+\sum_{i=1}^{p}\underbrace{[\bar{A}_1^p]^0}_{I}\,\bar{A}_{i+1}^p\bar{K}(i)z(sp+\mu p+(\nu-1)p+i).
\end{aligned}$$

Developing the above sums we obtain:

$$\begin{aligned}
&x(sp+\nu p+\mu p/sp+\nu p+\mu p)\\
&=[\bar{A}_1^p]^{\nu}\left\{\bar{A}_{1+1}^p\bar{K}(1)z(sp+\mu p-p+1)\right.\\
&\qquad+\bar{A}_{2+1}^p\bar{K}(2)z(sp+\mu p-p+2)\\
&\qquad\left.+\cdots+\bar{A}_{p+1}^p\bar{K}(p)z(sp+\mu p-p+p)\right\}\\
&\quad+[\bar{A}_1^p]^{\nu-1}\left\{\bar{A}_{1+1}^p\bar{K}(1)z(sp+\mu p+1)\right.\\
&\qquad+\bar{A}_{2+1}^p\bar{K}(2)z(sp+\mu p+2)\\
&\qquad\left.+\cdots+\bar{A}_{p+1}^p\bar{K}(p)z(sp+\mu p+p)\right\}\\
&\quad+[\bar{A}_1^p]^{\nu-2}\left\{\bar{A}_{1+1}^p\bar{K}(1)z(sp+\mu p+p+1)\right.\\
&\qquad+\bar{A}_{2+1}^p\bar{K}(2)z(sp+\mu p+p+2)\\
&\qquad\left.+\cdots+\bar{A}_{p+1}^p\bar{K}(p)z(sp+\mu p+p+p)\right\}\\
&\quad+\cdots+\\
&\quad+\bar{A}_1^p\left\{\bar{A}_{1+1}^p\bar{K}(1)z(sp+\mu p+(\nu-2)p+1)\right.\\
&\qquad\left.+\cdots+\bar{A}_{p+1}^p\bar{K}(p)z(sp+\mu p+(\nu-2)p+p)\right\}\\
&\quad+\left\{\bar{A}_{1+1}^p\bar{K}(1)z(sp+\mu p+(\nu-1)p+1)\right.\\
&\qquad\left.+\cdots+\bar{A}_{p+1}^p\bar{K}(p)z(sp+\mu p+(\nu-1)p+p)\right\}
\end{aligned}$$

$$\begin{aligned}
&=\left\{\begin{aligned}
&[\bar{A}_1^p]^{\nu-(1-1)\operatorname{div}p}\,\bar{A}_{(1-1)\bmod p+2}^p\\
&\qquad\cdot\bar{K}((1-1)\bmod p+1)z(sp+\mu p-p+1)\\
&+[\bar{A}_1^p]^{\nu-(2-1)\operatorname{div}p}\,\bar{A}_{(2-1)\bmod p+2}^p\\
&\qquad\cdot\bar{K}((2-1)\bmod p+1)z(sp+\mu p-p+2)\\
&+\cdots+\\
&+[\bar{A}_1^p]^{\nu-(p-1)\operatorname{div}p}\,\bar{A}_{(p-1)\bmod p+2}^p\\
&\qquad\cdot\bar{K}((p-1)\bmod p+1)z(sp+\mu p-p+p)
\end{aligned}\right\}\\
&\quad+\left\{\begin{aligned}
&[\bar{A}_1^p]^{\nu-(p+1-1)\operatorname{div}p}\,\bar{A}_{(p+1-1)\bmod p+2}^p\\
&\quad\cdot\bar{K}((p+1-1)\bmod p+1)z(sp+\mu p-p+(p+1))\\
&+[\bar{A}_1^p]^{\nu-(p+2-1)\operatorname{div}p}\,\bar{A}_{(p+2-1)\bmod p+2}^p\\
&\quad\cdot\bar{K}((p+2-1)\bmod p+1)z(sp+\mu p-p+(p+2))\\
&+\cdots+\\
&+[\bar{A}_1^p]^{\nu-(2p-1)\operatorname{div}p}\,\bar{A}_{(2p-1)\bmod p+2}^p\\
&\quad\cdot\bar{K}((2p-1)\bmod p+1)z(sp+\mu p-p+2p)
\end{aligned}\right\}\\
&\quad+\left\{\begin{aligned}
&[\bar{A}_1^p]^{\nu-(2p+1-1)\operatorname{div}p}\,\bar{A}_{(2p+1-1)\bmod p+2}^p\\
&\quad\cdot\bar{K}((2p+1-1)\bmod p+1)z(sp+\mu p-p+(2p+1))\\
&+[\bar{A}_1^p]^{\nu-(2p+2-1)\operatorname{div}p}\,\bar{A}_{(2p+2-1)\bmod p+2}^p\\
&\quad\cdot\bar{K}((2p+2-1)\bmod p+1)z(sp+\mu p-p+(2p+2))\\
&+\cdots+\\
&+[\bar{A}_1^p]^{\nu-(3p-1)\operatorname{div}p}\,\bar{A}_{(3p-1)\bmod p+2}^p\\
&\quad\cdot\bar{K}((3p-1)\bmod p+1)z(sp+\mu p-p+3p)
\end{aligned}\right\}\\
&\quad+\cdots+
\end{aligned}$$

$$+\left\{\begin{array}{l}[\overline{A}_1^p]^{\nu-((\nu-1)p+1-1)\operatorname{div}p}\,\overline{A}^p_{((\nu-1)p+1-1)\bmod p+2}\\ \cdot\overline{K}(((\nu-1)p+1-1)\bmod p+1)z(sp+\mu p-p+((\nu-1)p+1))\\ +[\overline{A}_1^p]^{\nu-((\nu-1)p+2-1)\operatorname{div}p}\,\overline{A}^p_{((\nu-1)p+2-1)\bmod p+2}\\ \cdot\overline{K}(((\nu-1)p+2-1)\bmod p+1)z(sp+\mu p-p+((\nu-1)p+2))\\ +\cdots+\\ +[\overline{A}_1^p]^{\nu-(\nu p-1)\operatorname{div}p}\,\overline{A}^p_{(\nu p-1)\bmod p+2}\\ \cdot\overline{K}((\nu p-1)\bmod p+1)z(sp+\mu p-p+\nu p)\end{array}\right\}$$

$$+\left\{\begin{array}{l}[\overline{A}_1^p]^{\nu-(\nu p+1-1)\operatorname{div}p}\,\overline{A}^p_{(\nu p+1-1)\bmod p+2}\\ \cdot\overline{K}((\nu p+1-1)\bmod p+1)z(sp+\mu p-p+(\nu p+1))\\ +[\overline{A}_1^p]^{\nu-(\nu p+2-1)\operatorname{div}p}\,\overline{A}^p_{(\nu p+2-1)\bmod p+2}\\ \cdot\overline{K}((\nu p+2-1)\bmod p+1)z(sp+\mu p-p+(\nu p+2))\\ +\cdots+\\ +[\overline{A}_1^p]^{\nu-((\nu+1)p-1)\operatorname{div}p}\,\overline{A}^p_{((\nu+1)p-1)\bmod p+2}\\ \cdot\overline{K}(((\nu+1)p-1)\bmod p+1)z(sp+\mu p-p+(\nu+1)p)\end{array}\right\}$$

$$=\sum_{k=1}^{p(\nu+1)}[\overline{A}_1^p]^{\nu-((k-1)\operatorname{div}p)}\,\overline{A}^p_{((k-1)\bmod p)+2}\overline{K}((k-1)\bmod p+1)\cdot z(sp+\mu p-p+k).$$

This expression completes the proof of equation (41) with the corresponding coefficients in equation (42):

$$c(k)=[\overline{A}_1^p]^{\nu-((k-1)\operatorname{div}p)}\,\overline{A}^p_{((k-1)\bmod p)+2}\overline{K}((k-1)\bmod p+1).$$

Note that, since $k-1=p((k-1)\operatorname{div}p)+(k-1)\bmod p$, substituting in equation (41) we take the following equivalent expression

$$\begin{aligned}&x(sp+\nu p+\mu p/sp+\nu p+\mu p)\\&=\sum_{k=1}^{p(\nu+1)}c(k)z(sp+\mu p-p+k)\\&=\sum_{k=1}^{p(\nu+1)}c(k)z(sp+\mu p-p+p((k-1)\operatorname{div}p)+(k-1)\bmod p+1)\end{aligned}$$

with coefficients $c(k)$, $k=1,2,\dots,p(\nu+1)$ as in equation (42).

Appendix C

Proof of equations (43) and (44)

We denote $r=sp+\nu p+\mu p$ and by equation (35) for $k=0,1,\dots,\varphi-1$ we have successively:

$$\begin{aligned}&x(r+1/r+1)=\overline{A}(1,0)x(r/r)+\overline{K}(1)z(r+1)\\&x(r+2/r+2)\\&\quad=\overline{A}(2,1)x(r+1/r+1)+\overline{K}(2)z(r+2)\\&\quad=\overline{A}(2,1)\overline{A}(1,0)x(r/r)+\overline{A}(2,1)\overline{K}(1)z(r+1)+\overline{K}(2)z(r+2)\\&\cdots\\&x(r+\varphi/r+\varphi)\\&\quad=\overline{A}(\varphi,\varphi-1)\overline{A}(\varphi-1,\varphi-2)\cdots\overline{A}(2,1)\overline{A}(1,0)x(r/r)\\&\qquad+\overline{A}(\varphi,\varphi-1)\overline{A}(\varphi-1,\varphi-2)\cdots\overline{A}(2,1)\overline{K}(1)z(r+1)+\cdots+\\&\qquad+\overline{A}(\varphi,\varphi-1)\overline{K}(\varphi-1)z(r+\varphi-1)+\overline{K}(\varphi)z(r+\varphi)\\&\quad=\overline{A}_1^{\varphi}x(r/r)+\sum_{i=1}^{\varphi}\overline{A}^{\varphi}_{i+1}\overline{K}(i)z(r+i).\end{aligned}$$

Thus, we take:

$$x(r+\varphi/r+\varphi)=\overline{A}_1^{\varphi}x(r/r)+\sum_{i=1}^{\varphi}\overline{A}^{\varphi}_{i+1}\overline{K}(i)z(r+i)$$

for $1\le\varphi\le p$, and substituting equation (41) in the last expression we have:

$$\begin{aligned}&x(r+\varphi/r+\varphi)=\overline{A}_1^{\varphi}x(r/r)+\sum_{i=1}^{\varphi}\overline{A}^{\varphi}_{i+1}\overline{K}(i)z(r+i)\\&=\overline{A}_1^{\varphi}\sum_{k=1}^{p(\nu+1)}c(k)z(sp+\mu p-p+k)\\&\quad+\sum_{i=1}^{\varphi}\overline{A}^{\varphi}_{i+1}\overline{K}(i)z(sp+\mu p+\nu p+i)\\&=\sum_{k=1}^{p(\nu+1)}\overline{A}_1^{\varphi}c(k)z(sp+\mu p-p+k)\\&\quad+\sum_{i=1}^{\varphi}\overline{A}^{\varphi}_{i+1}\overline{K}(i)z(sp+\mu p-p+p(\nu+1)+i)\\&=\sum_{k=1}^{p(\nu+1)+\varphi}\hat{c}(k)z(sp+\mu p-p+k).\end{aligned}$$

This expression completes the proof of equation (43) with the corresponding coefficients in equation (44):

$$\hat{c}(k)=\begin{cases}\overline{A}_1^{\varphi}c(k), & 1\le k\le p(\nu+1)\\ \overline{A}^{\varphi}_{((k-1)\bmod p)+2}\overline{K}((k-1)\bmod p+1), & p(\nu+1)+1\le k\le\varphi+p(\nu+1)\end{cases}$$

for $1\le\varphi\le p$, with coefficients $c(k)$, $k=1,2,\dots,p(\nu+1)$ as in equation (42).