

# Distributed Periodic Steady State Kalman Filter

Nicholas Assimakis  
General Department, National and Kapodistrian University of Athens,  
Psachna Evias, Greece

Maria Adam  
Department of Computer Science and Biomedical Informatics, University of Thessaly  
2-4 Papasiopoulou Str., Lamia, Greece

Christos Massouros  
General Department, National and Kapodistrian University of Athens  
Psachna Evias, Greece

Received: June 16, 2021. Revised: December 23, 2021. Accepted: January 15, 2022. Published: January 17, 2022

**Abstract:** - In this paper a distributed implementation for the periodic steady state Kalman filter is proposed. The distributed algorithm has parallel structure and can be implemented using processors in parallel without idle time. The number of processors is equal to the model period. The resulting speedup is also derived. The Finite Impulse Response (FIR) form of the periodic steady state Kalman filter is derived.

**Key-Words:** - Kalman filter, steady state, periodic models, distributed algorithms

## I. INTRODUCTION

Kalman filter [1]-[2] is the most well-known and widely used estimation algorithm, since it has been used to successfully solve real time problems in various applications, such as in [2] aerospace industry, chemical process, communication systems design, control, civil engineering, filtering noise from two dimensional images, pollution prediction and power systems.

The estimation problem is associated with time varying systems described by the following state space equations:

$$x(k+1) = F(k)x(k) + w(k) \quad (1)$$

$$z(k) = H(k)x(k) + v(k) \quad (2)$$

where  $x(k)$  is the  $n$ -dimensional state vector,  $z(k)$  is the  $m$ -dimensional measurement vector,  $F(k)$  is the transition matrix,  $H(k)$  is the output matrix,  $w(k)$  is the state noise and  $v(k)$  is the 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 and the measurement noise are zero mean Gaussian processes with known covariances  $Q(k)$  and  $R(k)$ , respectively. The following assumptions hold: (a) the initial value of the state  $x(0)$  is a Gaussian random variable with known mean  $x_0$  and covariance  $P_0$ ; (b) the noise processes and the initial state are independent.

Periodic linear systems arise from continuous linear systems, when multi-rate sampling is performed, with many interesting and practical applications as stated in [3]. In this paper we focus on the case of periodic models and especially on the periodic steady state Kalman filter [4]-[6]. Parallel Kalman Filter implementations are mentioned in [7]. Distributed implementations for the steady state Kalman filter are proposed in [8]. The problem of distributed state estimation is addressed in [9]. The problem of distributed state estimation has been

studied for a Linear Time Invariant (LTI) Gaussian system in [10] where distributed estimators are proposed and it is shown that the average of local estimates from all sensors coincides with the optimal Kalman estimate. The distributed Kalman filtering problem is studied for time-varying DSSs with Gaussian white noises in [11] where a locally optimal distributed estimator is designed in the linear minimum variance sense.

In this paper a distributed implementation for the periodic steady state Kalman filter is proposed. The novelty of this paper concerns: (a) The derivation of a distributed algorithm with parallel structure during each period. The proposed algorithm can be implemented using processors in parallel without idle time. The number of processors is equal to the model period. (b) The derivation of a distributed algorithm for the Finite Impulse Response (FIR) form of the periodic steady state Kalman filter.

The paper is organised as follows: In section 2 the periodic steady state Kalman filter is presented. In section 3 the parallel periodic steady state Kalman filter is proposed. In section 4 the speedup is discussed. In section 5 the FIR form of the periodic steady state Kalman filter is developed. Finally, section 6 summarizes the conclusion.

## II. PERIODIC STEADY STATE KALMAN FILTER

Kalman filter produces iteratively the state estimation  $x(k/k)$  and the corresponding estimation error covariance matrix  $P(k/k)$  as well as the state prediction  $x(k+1/k)$  and the corresponding prediction error covariance matrix  $P(k+1/k)$  using measurements till time  $k$  and the Kalman filter gain  $K(k)$ . For the varying systems described by (1), the Time Varying Kalman Filter is summarized in the following, [2]:

### Time Varying Kalman Filter (TVKF)

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

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

$$P(k/k) = [I - K(k)H(k)]P(k/k - 1) \quad (5)$$

$$x(k+1/k) = F(k)x(k/k) \quad (6)$$

$$P(k+1/k) = Q(k) + F(k)P(k/k)F^T(k) \quad (7)$$

for  $k = 0, 1, \dots$ ,

with initial conditions

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

In the following,  $M^T$  denotes the transpose matrix of matrix  $M$  and  $I$  is used for the identity matrix.

It is known [4]-[6] that in the case of periodic model, the model parameters (matrices) are periodic with period  $p$  ( $p$  is integer:  $p \geq 2$ ):

$$F(k) = F(k \bmod p) \quad (8)$$

$$H(k) = H(k \bmod p) \quad (9)$$

$$Q(k) = Q(k \bmod p) \quad (10)$$

$$R(k) = R(k \bmod p) \quad (11)$$

for  $k = 0, 1, \dots$

The corresponding discrete time periodic Riccati equation results from the Kalman filter equations (3), (5) and (7) and has as follows:

$$P(k+1/k) = Q(k) + F(k)P(k/k - 1)F^T(k) - F(k)P(k/k - 1)H^T(k) [H(k)P(k/k - 1)H^T(k) + R(k)]^{-1} H(k)P(k/k - 1)F^T(k) \quad (12)$$

It is known [4]-[6] for periodic systems that when steady state exists, then the steady state prediction error covariance is periodic with period  $p$ :

$$\bar{P}(k+1/k) = \bar{P}(k \bmod p + 1/k \bmod p) \quad (13)$$

Then, the steady state Kalman filter gain becomes periodic with period  $p$ :

$$\bar{K}(k) = \bar{K}(k \bmod p) \quad (14)$$

and the steady state estimation error covariance is periodic with period  $p$ :

$$\bar{P}(k/k) = \bar{P}(k \bmod p/k \bmod p) \quad (15)$$

Thus, by solving the periodic Riccati equation [3], [12]-[13] the periodic steady state prediction error covariance is computed. Then the periodic steady state Kalman filter gain can be computed using

$$\bar{K}(k) = \bar{P}(k/k - 1)H^T(k) [H(k)\bar{P}(k/k - 1)H^T(k) + R(k)]^{-1} \quad (16)$$

and the periodic estimation error covariance can be computed using

$$\bar{P}(k/k) = [I - \bar{K}(k)H(k)]\bar{P}(k/k - 1) \quad (17)$$

in a period time.

Furthermore, from the Kalman filter equations (4) and (6) and for  $k = 1, 2, \dots$ , the estimation is derived:

$$x(k/k) = A(k)x(k-1/k-1) + K(k)z(k) \quad (18)$$

where

$$A(k) = [I - K(k)H(k)]F(k-1) \quad (19)$$

with initial condition

$$x(0/0) = [I - K(0)H(0)]x(0/-1) + K(0)z(0) \quad (20)$$

and

$$K(0) = P(0/-1)H^T(0) [H(0)P(0/-1)H^T(0) + R(0)]^{-1} \quad (21)$$

Then the Periodic Steady State Kalman Filter is derived:

### Periodic Steady State Kalman Filter

$$x(k/k) = \bar{A}(k \bmod p)x(k-1/k-1) + \bar{K}(k \bmod p)z(k) \quad (22)$$

for  $k = 1, 2, \dots$ , with initial condition  $x(0/0)$  computed by (20) and (21).

Obviously, the coefficients  $\bar{A}(k \bmod p)$  are periodic with period  $p$

$$\bar{A}(k) = \begin{cases} \bar{A}(k \bmod p) & k \neq \lambda p \\ \bar{A}(p) & k = \lambda p \end{cases} \quad (23)$$

for  $k = 1, 2, \dots, \lambda = 1, 2, \dots$ ,

and they are calculated off-line by first solving the corresponding discrete time periodic Riccati equation, then computing the periodic steady state Kalman filter gain using (14) and (16) we take

$$\bar{A}(k) = [I - \bar{K}(k)H(k)]F(k-1) \quad (24)$$

in a period time.

### III. PARALLEL PERIODIC STEADY STATE KALMAN FILTER

Parallel Kalman Filter implementations are mentioned in [7]-[8]. We are going to develop a parallel algorithm for the Periodic Steady State Kalman Filter, taking advantage of the system periodicity. We use the basic equation (22) and (14), (23) to rewrite the estimations in a period.

For example, for period  $p = 3$ , we have:

$$x(0/0) = [I - K(0)H(0)]x(0/-1) + K(0)z(0)$$

$$x(1/1) = \bar{A}(1)x(0/0) + \bar{K}(1)z(1)$$

$$x(2/2) = \bar{A}(2)x(1/1) + \bar{K}(2)z(2) \\ = \bar{A}(2)\bar{A}(1)x(0/0) + \bar{A}(2)\bar{K}(1)z(1) + \bar{K}(2)z(2)$$

$$x(3/3) = \bar{A}(3)x(2/2) + \bar{K}(3)z(3) \\ = \bar{A}(3)\bar{A}(2)\bar{A}(1)x(0/0) \\ + \bar{A}(3)\bar{A}(2)\bar{K}(1)z(1) + \bar{A}(3)\bar{K}(2)z(2) + \bar{K}(3)z(3)$$

Then, we derive:

$$x(4/4) = \bar{A}(4)x(3/3) + \bar{K}(4)z(4) \\ = \bar{A}(4)\bar{A}(3)\bar{A}(2)\bar{A}(1)x(0/0) \\ + \bar{A}(4)\bar{A}(3)\bar{A}(2)\bar{K}(1)z(1) \\ + \bar{A}(4)\bar{A}(3)\bar{K}(2)z(2) \\ + \bar{A}(4)\bar{K}(3)z(3) + \bar{K}(4)z(4) \\ = \bar{A}(4)\bar{A}(3)\bar{A}(2)x(1/1) \\ + \bar{A}(4)\bar{A}(3)\bar{K}(2)z(2) + \bar{A}(4)\bar{K}(3)z(3) + \bar{K}(4)z(4)$$

$$x(5/5) = \bar{A}(5)x(4/4) + \bar{K}(5)z(5) \\ = \bar{A}(5)\bar{A}(4)\bar{A}(3)\bar{A}(2)x(1/1) \\ + \bar{A}(5)\bar{A}(4)\bar{A}(3)\bar{K}(2)z(2) \\ + \bar{A}(5)\bar{A}(4)\bar{K}(3)z(3) + \bar{A}(5)\bar{K}(4)z(4)$$

$$+ \bar{K}(5)z(5) \\ = \bar{A}(5)\bar{A}(4)\bar{A}(3)x(2/2) \\ + \bar{A}(5)\bar{A}(4)\bar{K}(3)z(3) + \bar{A}(5)\bar{K}(4)z(4) + \bar{K}(5)z(5)$$

$$x(6/6) = \bar{A}(6)x(5/5) + \bar{K}(6)z(6) \\ = \bar{A}(6)\bar{A}(5)\bar{A}(4)\bar{A}(3)x(2/2) \\ + \bar{A}(6)\bar{A}(5)\bar{A}(4)\bar{K}(3)z(3) \\ + \bar{A}(6)\bar{A}(5)\bar{K}(4)z(4) + \bar{A}(6)\bar{K}(5)z(5) + \bar{K}(6)z(6) \\ = \bar{A}(6)\bar{A}(5)\bar{A}(4)x(3/3) \\ + \bar{A}(6)\bar{A}(5)\bar{K}(4)z(4) + \bar{A}(6)\bar{K}(5)z(5) + \bar{K}(6)z(6)$$

and

$$x(7/7) = \bar{A}(7)\bar{A}(6)\bar{A}(5)x(4/4) \\ + \bar{A}(7)\bar{A}(6)\bar{K}(5)z(5) + \bar{A}(7)\bar{K}(6)z(6) + \bar{K}(7)z(7)$$

$$x(8/8) = \bar{A}(8)\bar{A}(7)\bar{A}(6)x(5/5) \\ + \bar{A}(8)\bar{A}(7)\bar{K}(6)z(6) + \bar{A}(8)\bar{K}(7)z(7) + \bar{K}(8)z(8)$$

$$x(9/9) = \bar{A}(9)\bar{A}(8)\bar{A}(7)x(6/6) \\ + \bar{A}(9)\bar{A}(8)\bar{K}(7)z(7) + \bar{A}(9)\bar{K}(8)z(8) + \bar{K}(9)z(9)$$

and so on.

It is obvious that after the first period the structure of the equations remains constant per period due to periodicity of  $\bar{K}(k)$  and  $\bar{A}(k)$ .

The Parallel Periodic Steady State Kalman Filter is then derived:

#### Parallel Periodic Steady State Kalman Filter

$$x(ip + j/ip + j) = \alpha_j x((i-1)p + j/(i-1)p + j) + \bar{K}(ip + j)z(ip + j) \quad (25)$$

$$+ \sum_{\ell=1}^{p-1} \beta_{j\ell} \bar{K}((i-1)p + j + \ell)z((i-1)p + j + \ell)$$

for  $i = 1, 2, \dots$  and  $j = 1, \dots, p$ ,

with initial conditions  $x(1/1), \dots, x(p/p)$ ,

where the coefficients  $\alpha_j, \beta_{j\ell}, j = 1, \dots, p$ , and  $\ell = 1, \dots, p-1$  are defined

$$\alpha_j = \prod_{\ell=1}^p \bar{A}(ip + j - \ell + 1) \quad (26)$$

and

$$\beta_{j\ell} = \prod_{r=\ell+1}^p \bar{A}(ip + j + \ell + 1 - r) \quad (27)$$

It is evident that the coefficients  $\alpha_j, \bar{K}(ip + j)$  and  $\beta_{j\ell} \bar{K}((i-1)p + j + \ell)$ , for  $j = 1, \dots, p$ , and  $\ell = 1, \dots, p-1$ , depend on the periodic coefficients

of the Periodic Steady State Kalman Filter and hence they are known and are calculated off-line.

#### IV. SPEEDUP

Due to the fact that the sequential and parallel Periodic Steady State Kalman Filter algorithms (22) and (25) are iterative, in order to compute their computational time, we have to derive their per-iteration calculation burden required for the on-line calculations; the calculation burden of the off-line calculations is not taken into account. The implementation of the Periodic Steady State Kalman Filter algorithms requires matrix operations, which involve scalar additions and scalar multiplications. Let scalar multiplication =  $c \cdot$  scalar addition, where  $c \geq 1$ . Depending on the application, but also on processor technology,  $c \approx 1$ . For instance, [14] reports the same latency but different throughput for the two operations. Then Table 1 summarizes the calculation burden of matrix operations, which are needed for the implementation of the algorithms.

**Table 1. Calculation burden of matrix operations**

Operation and Dimensions	scalar mults	scalar adds	Calculation Burden
$A + B$ $(n \times 1) + (n \times 1)$		$n$	$n$
$A \cdot x$ $(n \times n) \cdot (n \times 1)$	$n^2$	$n^2 - n$	$(c + 1)n^2 - n$
$B \cdot x$ $(n \times m) \cdot (m \times 1)$	$nm - n$	$nm - n$	$(c + 1)nm - n$

The Periodic Steady State Kalman Filter algorithm in (22) does not take advantage of the periodicity of the coefficients. Table 2 summarizes the per-iteration calculation burden of the Periodic Steady State Kalman Filter algorithm in (22).

**Table 2. Periodic Steady State Kalman Filter calculation burden**

Periodic Steady State Kalman Filter (eq. 22)	
Operation and Dimensions	per iteration Calculation Burden
$W_1 = \bar{A}(k \bmod p)x(k - 1/k - 1)$ $(n \times n) \cdot (n \times 1)$	$(c + 1)n^2 - n$
$W_2 = \bar{K}(k \bmod p)z(k)$ $(n \times m) \cdot (m \times 1)$	$(c + 1)nm - n$
$x(k/k) = W_1 + W_2$ $(n \times 1) + (n \times 1)$	$n$
<b>total</b>	$(c + 1)n^2 + (c + 1)nm - n$

The Parallel Periodic Steady State Kalman Filter algorithm in (25) does take advantage of the

periodicity of the coefficients. The basic idea is to compute  $p$  successive iterations in  $p$  ( $p$  is integer:  $p \geq 2$ ) parallel processors, i.e. to use  $p$  processors in a period. Table 3 summarizes the per-iteration calculation burden of the Periodic Steady State Kalman Filter algorithm in (25).

**Table 3. Parallel Periodic Steady State Kalman Filter calculation burden**

Parallel Periodic Steady State Kalman Filter (eq. 25)		
Operation and Dimensions	times	per period Calculation Burden
$W_1 = \alpha_1 x(\xi - p/\xi - p)$ $(n \times n) \cdot (n \times 1)$	1	$(c + 1)n^2 - n$
$W_2 = \bar{K}(\xi)z(\xi)$ $(n \times m) \cdot (m \times 1)$	1	$((c + 1)nm - n)$
$W_\ell = \beta_\ell \bar{K}(\xi - p + \ell)z(\xi - p + \ell)$ $(n \times m) \cdot (m \times 1)$	$p - 1$	$((c + 1)nm - n)(p - 1)$
$x(\xi/\xi) = W_1 + W_2 + \sum_{\ell=1}^{p-1} W_\ell$ $(n \times 1) + (n \times 1)$	$p$	$np$
where $\xi = ip + j$	<b>total</b>	$(c + 1)n^2 + (c + 1)npm - n$

The speedup is then computed for a period time:

$$\text{speedup} = \frac{((c + 1)n^2 + (c + 1)nm - n)p}{(c + 1)n^2 + (c + 1)npm - n}$$

and

$$\text{speedup} = \frac{((c + 1)n + (c + 1)m - 1)p}{(c + 1)n + (c + 1)mp - 1} \quad (28)$$

with

$$1 < \text{speedup} < p \quad (29)$$

(the proof is trivial).

Remarks.

1. The number of processors is equal to the period.
2. No processor is idle: all the processors do the same work.
3. The speedup increases as the period increases (the proof is trivial). In fact the speedup is an increasing function of the period  $p$ .

When  $p = 2$ , the minimum speedup is achieved:

$$\text{speedup}_{\min} = \frac{2((c + 1)n + (c + 1)m - 1)}{(c + 1)n + 2(c + 1)m - 1} \approx \frac{2(n + m)}{n + 2m}$$

When  $p$  tends to infinity, the maximum speedup is achieved:

$$\begin{aligned} \text{speedup}_{\max} &= \frac{(c+1)n + (c+1)m - 1}{(c+1)m} \\ &= 1 + \frac{(c+1)n - 1}{(c+1)m} \\ &\approx 1 + \frac{(c+1)n}{(c+1)m} = 1 + \frac{n}{m} \end{aligned}$$

Example 1.

This example is the Example in section 6 in [15] with dimensions  $n = 2$ ,  $m = 1$  and period  $p = 3$ . Assuming  $c = 1$ , the speedup is  $\text{speedup} = 1.667$ .

Example 2.

This example is the Example 2 in [16] with dimensions  $n = 20$ ,  $m = 1$  and period  $p = 2$ . Assuming  $c = 1$ , the speedup is  $\text{speedup} = 1.907$ .

Example 3.

This example is the Example 8 in [6] with dimensions  $n = 3$ ,  $m = 1$  and period  $p = 3$ . Assuming  $c = 1$ , the speedup is  $\text{speedup} = 1.909$ .

Example 4.

This example is the Example 9 in [6] with dimensions  $n = 4$ ,  $m = 1$  and period  $p = 120$ . Assuming  $c = 1$ , the speedup is  $\text{speedup} = 4.372$ .

#### V. PARALLEL IMPLEMENTATION OF THE FIR FORM OF THE PERIODIC STEADY STATE KALMAN FILTER

We are going to derive the FIR form of the Periodic Steady State Kalman Filter. We use the basic equation (22) to rewrite the estimations:

$$\begin{aligned} x(1/1) &= \bar{A}(1)x(0/0) + \bar{K}(1)z(1) \\ x(2/2) &= \bar{A}(2)x(1/1) + \bar{K}(2)z(2) \\ &= \bar{A}(2)\bar{A}(1)x(0/0) + \bar{A}(2)\bar{K}(1)z(1) + \bar{K}(2)z(2) \end{aligned}$$

...

$$\begin{aligned} x(p/p) &= \bar{A}(p) \cdots \bar{A}(2)\bar{A}(1)x(0/0) \\ &\quad + \bar{A}(p)\bar{A}(p-1) \cdots \bar{A}(2)\bar{K}(1)z(1) \\ &\quad + \bar{A}(p)\bar{A}(p-1) \cdots \bar{A}(3)\bar{K}(2)z(2) \\ &\quad + \dots + \bar{A}(p)\bar{K}(p-1)z(p-1) + \bar{K}(p)z(p) \end{aligned}$$

...

$$\begin{aligned} x(vp/vp) &= \bar{A}(vp)\bar{A}(vp-1) \cdots \bar{A}(2)\bar{A}(1)x(0/0) \\ &\quad + \bar{A}(vp)\bar{A}(vp-1) \cdots \bar{A}(2)\bar{K}(1)z(1) \end{aligned}$$

$$\begin{aligned} &+ \bar{A}(vp)\bar{A}(vp-1) \cdots \bar{A}(3)\bar{K}(2)z(2) \\ &+ \dots + \bar{A}(vp)\bar{K}(vp-1)z(vp-1) \\ &+ \bar{K}(vp)z(vp) \\ &= (\bar{A}(p)\bar{A}(p-1) \cdots \bar{A}(2)\bar{A}(1))^v x(0/0) \\ &\quad + \sum_{\ell=1}^{vp-1} \bar{A}(vp)\bar{A}(vp-1) \cdots \bar{A}(\ell+1)\bar{K}(\ell)z(\ell) \\ &\quad + \bar{K}(vp)z(vp) \end{aligned} \tag{30}$$

Owing to the known property that “if all the eigenvalues of a matrix  $A$  is less than 1, then the computed powers of  $A$  can be expected to converge to zero”, [17]; considering that all eigenvalues of  $\bar{A}(p) \cdots \bar{A}(2)\bar{A}(1)$  lie inside the unit circle, the computed powers of  $\bar{A}(p) \cdots \bar{A}(2)\bar{A}(1)$  can be expected to converge to zero, whereby we conclude that, there exists  $v$ :

$$\begin{aligned} &(\bar{A}(p)\bar{A}(p-1) \cdots \bar{A}(2)\bar{A}(1))^{v-1} \neq 0 \\ &\text{and} \\ &(\bar{A}(p)\bar{A}(p-1) \cdots \bar{A}(2)\bar{A}(1))^v \rightarrow 0 \end{aligned} \tag{31}$$

Now, by (31) the equation (30) can be written

$$\begin{aligned} x(vp/vp) &= \bar{K}(vp)z(vp) \\ &+ \sum_{\ell=1}^{vp-1} \bar{A}(vp)\bar{A}(vp-1) \cdots \bar{A}(\ell+1)\bar{K}(\ell)z(\ell) \end{aligned} \tag{32}$$

and from (32) we have

$$\begin{aligned} x(vp+i/vp+i) &= \bar{K}(vp+i)z(vp+i) \\ &+ \sum_{\ell=1}^{vp-1} \bar{A}(vp+i)\bar{A}(vp+i-1) \cdots \bar{A}(\ell+1)\bar{K}(\ell)z(\ell) \end{aligned}$$

for  $i = 1, 2, \dots$ , which is formulated as

$$\begin{aligned} x(vp+i/vp+i) &= \bar{K}(vp+i)z(vp+i) \\ &\quad + \sum_{\ell=1}^{vp+i-1} \delta_{i\ell} \bar{K}(\ell)z(\ell) \end{aligned} \tag{33}$$

where

$$\delta_{i\ell} = \prod_{r=\ell+1}^{vp+i} \bar{A}(vp+i+\ell+1-r) \tag{34}$$

for  $\ell = 1, 2, \dots, vp+i-1$ .

Working as in [18], owing to the known property that “the eigenvalues of the matrix  $A \cdot B$  are the same as those of the matrix  $B \cdot A$ ”, [2], assuming that all the eigenvalues of  $\prod_{i=1}^p \bar{A}(p+1-i) = \bar{A}(p) \cdots \bar{A}(2)\bar{A}(1)$  lie inside the unit circle, we

conclude that all the eigenvalues of the derived matrices by permutation of  $\bar{A}(1), \bar{A}(2), \dots, \bar{A}(p)$  in the above matrix  $\prod_{i=1}^p \bar{A}(p+1-i)$  lie inside the unit circle and due to (23), (31) we have

$$\begin{aligned} \bar{A}(1)(\bar{A}(p) \cdots \bar{A}(2)\bar{A}(1))^{v-1}\bar{A}(p) \cdots \bar{A}(2) &\rightarrow 0 \\ \bar{A}(2)\bar{A}(1)(\bar{A}(p) \cdots \bar{A}(2)\bar{A}(1))^{v-1}\bar{A}(p) \cdots \bar{A}(3) &\rightarrow 0 \\ \dots & \\ \bar{A}(p-1) \cdots \bar{A}(1)(\bar{A}(p) \cdots \bar{A}(2)\bar{A}(1))^{v-1}\bar{A}(p) &\rightarrow 0 \end{aligned}$$

Hence, from (33) we obtain

$$\begin{aligned} x(vp+i/vp+i) &= \bar{K}(vp+i)z(vp+i) \\ &+ \sum_{\ell=i+1}^{vp+i-1} \delta_{i\ell} \bar{K}(\ell)z(\ell) \end{aligned} \quad (35)$$

where  $\delta_{i\ell}$  are given by (34) for  $i = 1, 2, \dots$  and  $\ell = (i+1), (i+2), \dots, (vp+i-1)$ .

In (35) substituting  $\mu = \ell - i$  the FIR form of the Periodic Steady State Kalman Filter is derived:

### FIR Periodic Steady State Kalman Filter

$$x(vp+i/vp+i) = \sum_{\mu=1}^{vp} d_{i\mu} \bar{K}(\mu+i)z(\mu+i) \quad (36)$$

where  $d_{i\mu}$ ,  $i = 1, 2, \dots, \mu = 1, 2, \dots, vp$  are given by

$$d_{i\mu} = \begin{cases} \prod_{\tau=1}^{vp-\mu} \bar{A}(vp+i+1-\tau), & 1 \leq \mu \leq vp-1 \\ I, & \mu = vp \end{cases} \quad (37)$$

and the coefficients  $d_{i\mu}$  depend on the periodic coefficients of the Periodic Steady State Kalman Filter in (22) and hence they are known and are calculated off-line in a period, since they are periodic with period  $p$ .

Remarks.

1. The FIR Periodic Steady State Kalman coefficients are calculated a-priori.
2. No previous estimations are needed.
3. The estimation depends only on a well-defined set of measurements ( $vp$  measurements are needed).
4. We are able to extend this result assuming that  $z(k) = 0, k < 0$  and defining the model parameters periodicity for  $k < 0$ , in order to compute  $x(k/k)$  for  $k = 1, 2, \dots$
5. The FIR form of the periodic Steady State Kalman filter can be implemented in parallel, using the parallel addition algorithm with  $\lceil \log_2(vp) \rceil$  processors.

## VI. CONCLUSION

We focused on the case of periodic models and especially on the periodic steady state Kalman filter.

We proposed a distributed implementation for the periodic steady state Kalman filter. We derived a distributed algorithm with parallel structure during each period. The proposed algorithm can be implemented using processors in parallel without idle time. The number of processors is equal to the model period. The resulting speedup increases as the period increases.

The FIR form of the periodic steady state Kalman filter is developed. The coefficients of the filter are calculated a-priori. No previous estimations are needed. The estimation depends only on a well-defined set of measurements. The FIR form of the periodic steady state Kalman filter can be implemented in parallel.

## ACKNOWLEDGMENTS

We thank the NAUN reviewers (really international experts on their field of research) that NAUN used for our paper.

We acknowledge support of this work by the project "ParICT\\_CENG: Enhancing ICT research infrastructure in Central Greece to enable processing of Big data from sensor stream, multimedia content, and complex mathematical modeling and simulations" (MIS 5047244) which is implemented under the Action "Reinforcement of the Research and Innovation Infrastructure", funded by the Operational Programme "Competitiveness, Entrepreneurship and Innovation" (NSRF 2014-2020) and co-financed by Greece and the European Union (European Regional Development Fund).

## REFERENCES

- [1] R.E. Kalman, A New Approach to Linear Filtering and Prediction Problems, *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, pp. 35-45, 1960.
- [2] Anderson B.D.O., and Moore J.B., *Optimal Filtering*, Dover Publications, New York, 2005.
- [3] Adam M., Assimakis N., Periodic Kalman filter: Steady state from the beginning, *Journal of Mathematical Sciences: Advances and Applications*, vol. 1, no. 3, pp. 505-520, 2008.
- [4] Assimakis N., Adam M., Steady state Kalman filter for periodic models: A new approach, *International Journal of Contemporary Mathematical Sciences*, vol. 4, no. 5, pp. 201-218, 2009.
- [5] Bucy R.S. and Campell L.A., Determination of steady state behavior for periodic discrete filtering problems, *Comput. Math. Applic.*, vol. 15, no. 2, pp. 131-140, 1988.
- [6] Chu, E.K.W., Fan H.Y., Lin W.W. and Wang C.S., Structure-preserving algorithm for periodic discrete-time algebraic Riccati equations, *Int. J. Control*, vol. 77, i.s. 8, pp. 767-788, 2004.
- [7] Assimakis N., Adam M., Koziri M., Voliotis S., Assimakis K., Optimal decentralized Kalman filter and Lainiotis filter, *Digital Signal Processing*, vol. 23, no. 1, pp. 44-52, DOI: 10.1016/j.dsp.2012.08.005, 2013.
- [8] Koziri M., Loukopoulos T., Adam M., Assimakis N., Tziallas G. On the Optimal Processor Assignment for Computing the Steady State Kalman Filter in Parallel and Distributed Systems, *Proceedings of 5th World Conference on Information Systems and Technologies*, pp. 427-437, 2017.
- [9] Viegas D., Batista P., Oliveira P., Silvestre C., Discrete-time distributed Kalman filter design for formations of autonomous vehicles, *Control Engineering Practice*, vol. 75, pp. 55–68, 2018.
- [10] Yan J., Yang X., Mo Y., You K., A Distributed Implementation of Steady-State Kalman Filter, arXiv: 2101.10689v2 [eess.SY] 17 Mar 2021, 2021.
- [11] Chen B., Hu G., Ho D. W., Yu L., Distributed Kalman filtering for time-varying discrete sequential systems, *Automatica*, vol. 99, pp. 228–236, 2019.
- [12] Lainiotis D.G., Assimakis N.D., Katsikas S.K., New doubling algorithm for the discrete periodic Riccati equation, *Applied Mathematics and Computation*, vol. 60, no. 2-3, pp. 265-283, 1994.
- [13] Varga A., On solving Discrete-Time Periodic Riccati Equations, *Proc. Of 16th IFAC World Congress 2005*, Prague, July 2005.
- [14] Intel®64 and IA-32 architectures optimization reference manual. <http://www.intel.com/content/www/us/en/architecture-and-technology/64-ia-32-architectures-optimization-manual.html>
- [15] Varga A., Computation of Kalman Decompositions of Periodic Systems, *European Journal of Control*, vol. 10, issue 1, pp. 1-8, 2004.
- [16] Larin V.B., High-accuracy algorithms for solving the discrete-time periodic Riccati equation, *Int Appl Mech*, vol.43, pp. 1028-1034, 2007.
- [17] Higham N.J. and Knight, P.A. Matrix powers in finite precision arithmetic, *SIAM J. Matrix Anal. Appl.*, vol. 16, no. 2, pp.343–358, 1995.
- [18] Assimakis N., Adam M., FIR implementation of the steady state Kalman filter, *International Journal of Signal and Imaging Systems Engineering (IJSISE)*, vol. 1, no s. 3/4, pp. 279-286, 2008.

### Contribution of individual authors to the creation of a scientific article (ghostwriting policy)

Nicholas Assimakis: Conceptualization, Methodology.

Maria Adam: Formal analysis, Investigation, Writing - original draft preparation.

Christos Massourous: Validation, Writing - review and editing and Visualization.

### Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)

This article is published under the terms of the Creative Commons Attribution License 4.0