SVD-based State Estimation of Pairwise Markov Models with Application in Econometrics

Maria V. Kulikova
CEMAT, Instituto Superior Técnico
Universidade de Lisboa
Lisbon, Portugal
maria.kulikova at ist.utl.pt

Julia V. Tsyganova
Department of Mathematics, Information and Aviation Technology, Ulyanovsk State University
Ulyanovsk, Russian Federation
TsyganovaJV at gmail.com

Gennady Yu. Kulikov
CEMAT, Instituto Superior Técnico
Universidade de Lisboa
Lisbon, Portugal
gkulikov at math.ist.utl.pt

***Abstract*—In this paper, several general forms of linear time-invariant (LTI) state-space models are explored. In particular, a numerical robustness of the so-called pairwise Kalman filter (PKF) is investigated. We propose stable singular value decomposition (SVD) factorization-based algorithms for implementing the PKF and LTI MIMO estimator, respectively, and explain their practical applicability in econometrics discipline. More precisely, the test for evolving efficiency is expressed in the LTI MIMO form and the newly-derived SVD-based estimator is applied for recovering the Russian market weak-form efficiency process in the last 20 years.**

***Index Terms*—Pairwise Markov model, pairwise Kalman filter, singular value decomposition, test for evolving market efficiency**

I. Introduction

In the past few years, hidden Markov models (HMMs) have been generalized in various ways; e.g. see linear time-invariant multiple-input, multiple-output systems (LTI MIMO) examined in [1], [2], bilinear systems studied in [3] and pairwise Markov models (PMMs) with the related linear estimator named as the pairwise Kalman filter (PKF) and discussed in [4]–[7]. We focus on linear Gaussian PMMs and the closely related LTI MIMO class of models. It is worth noting here that in the PMMs framework, the HMMs processing is also available as explained in [8]. More precisely, the PMMs imply that the pair consisting of the hidden and observable processes, $T = (X, Y)$, is Markovian, in contrast to the HMMs methodology where the unknown process X is assumed to be Markov [9]. We stress that both the LTI MIMO and linear PMMs generalize the classical state-space model structure, for which the classical Kalman filter (KF) has been derived. Thus, these model specifications allow an optimal filtering method for estimating a hidden state process similar to the classical KF. Consequently, the resulted KF-like estimators possess numerical instability problems intrinsic to the classical KF methodology [10].

The *factored-form* (square-root) algorithms are recognized to be preferred implementations for treating a numerical instability problem of the classical KF [11]. The key idea of square-root (SR) algorithms is to ensure the symmetric form and positive semi-definiteness of error covariance matrix P by decomposing it in the form $P = SS^T$ and, next, reformulating the filter equations in terms of the resulted factors, only [12, Chapter 7]. The square-root (SR) methodology yields a wide variety of the KF implementation methods, among which the most popular are the Cholesky and UD factorization-based methods; see [13]–[15] and many others. The most recent development in this realm is the most stable SVD-based KF implementation proposed in [16].

Although considerable research has been devoted to the factored-form KFs design, rather less attention has been paid to derivation of robust PKF and LTI MIMO practical methods. So far, investigations have been confined to the Cholesky-based PKF algorithms in [6], [7] and LTI MIMO KF-like methods in [1], [2]. The most recent development in this realm is the new UD-based PKF strategy proposed in [17]. Despite the newly revealed benefits of the SVD-based KF strategy, such implementations do not exist neither for the PKF nor for the LTI MIMO systems, yet. In this paper we are going to fill in this gap by deriving the SVD-based KF-like estimators for the examined models. Additionally, the considered systems are shown to be better suited for some structural econometric models than the classical state-space representation. More precisely, the test for evolving efficiency proposed in [18] is discussed. Under this methodology, the newly developed SVD-based technique is applied to the Russia Trading System Index (RTSI) for recovering the Russian market weak-form efficiency process in the last 20 years.

II. State-Space Representation: PMMs and LTI MIMO Systems

Consider the classical state-space model representation for linear Gaussian HMMs

$$x_{k+1} = Fx_k + Bu_k + w_k, \qquad w_k \sim \mathcal{N}(0, \Theta), \quad (1)$$
$$y_k = Hx_k + v_k \qquad v_k \sim \mathcal{N}(0, R) \quad (2)$$

where $F \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times d}$ and $H \in \mathbb{R}^{m\times n}$ are known at each time instance t_k. The vectors $x_k \in \mathbb{R}^n$, $u_k \in \mathbb{R}^d$ and $y_k \in \mathbb{R}^m$ are unknown dynamic state, known deterministic input and available measurement vector, respectively. Random variables x_0, w_k and v_k are assumed to be normally distributed

The first and third authors acknowledge the support from the Portuguese National Fund (*Fundação para a Ciência e a Tecnologia*) within projects UID/Multi/04621/2013 and the *Investigador FCT 2013* programme.

and satisfy the following properties:

$$
\begin{aligned}
&\mathbf{E}\{x_0\} = \bar{x}_0, && \mathbf{E}\big\{(x_0-\bar{x}_0)(x_0-\bar{x}_0)^T\big\} = \Pi_0,\\
&\mathbf{E}\{w_k\} = \mathbf{E}\{v_k\} = 0, && \mathbf{E}\big\{w_k x_0^T\big\} = \mathbf{E}\big\{v_k x_0^T\big\} = 0,\\
&\mathbf{E}\big\{w_k v_k^T\big\} = 0, && \mathbf{E}\big\{w_k w_j^T\big\} = \Theta\delta_{kj},\\
& && \mathbf{E}\big\{v_k v_j^T\big\} = R\delta_{kj}
\end{aligned}
$$

where covariance matrices $\Theta \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{m\times m}$ are known. The symbol δ_{kj} is the Kronecker delta function.

The classical KF applied for estimating the hidden dynamic state process $\{x_k\}_{k=1}^N$ from the observed sequence $\{y_k\}_{k=1}^N$, yields the minimum expected mean square error (MSE) estimate, $\{\hat{x}_{k|k}\}_{k=1}^N$, for *linear Gaussian* state-space models. The quantity $\hat{x}_{k|k}$ stands for state estimate at time instance t_k, given the available measurements $\{y_1,\dots,y_k\}$. The classical KF recursion is given as follows [19, Theorem 9.2.1]:

Algorithm 1. KF (*Conventional KF implementation*)

INITIALIZATION: $(k=0)$ $\hat{x}_{0|0} = \bar{x}_0$ and $P_{0|0} = \Pi_0$.
TIME UPDATE: $(k=\overline{1,N})$ $\triangleright$ PRIORI ESTIMATION
$\quad \hat{x}_{k|k-1} = F\hat{x}_{k-1|k-1} + Bu_{k-1}$,
$\quad P_{k|k-1} = FP_{k-1|k-1}F^T + \Theta$.
MEASUREMENT UPDATE: $\triangleright$ POSTERIORI ESTIMATION
$\quad R_{e,k} = HP_{k|k-1}H^T + R$,
$\quad K_k = P_{k|k-1}H^T R_{e,k}^{-1}$,
$\quad \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k e_k$ where $e_k = y_k - H\hat{x}_{k|k-1}$,
$\quad P_{k|k} = (I - K_kH)P_{k|k-1}$.

The important property of the KF for Gaussian state-space models (1), (2) is that $e_k \sim \mathcal{N}(0, R_{e,k})$ where $\{e_k\}$ are the discrete-time KF innovations. It enables to express the log likelihood function (LF) in the following form [20], [21]:

$$
\begin{aligned}
\ln L(\theta|Y_N) &= \sum_{k=1}^N \ln p(y_k|Y_{k-1})\\
&= -\frac{mN}{2}\ln 2\pi - \frac{1}{2}\sum_{k=1}^N \Big\{\ln(\det R_{e,k}) + e_k^T R_{e,k}^{-1} e_k\Big\}. \quad (3)
\end{aligned}
$$

The LTI MIMO systems examined in this paper naturally extend the classical state-space model (1), (2) as follows [1]:

$$
\begin{bmatrix} x_{k+1}\\ y_k \end{bmatrix} = \begin{bmatrix} F & B\\ H & D \end{bmatrix}\begin{bmatrix} x_k\\ u_k \end{bmatrix} + \begin{bmatrix} w_k\\ v_k \end{bmatrix}, \qquad (4)
$$

$$
\begin{bmatrix} w_k\\ v_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0\\ 0 \end{bmatrix}, \begin{bmatrix} \Theta & S\\ S^T & R \end{bmatrix}\right) \qquad (5)
$$

where $\{x_k\}_{k=0}^N$ is the hidden process to be estimated, the sequences $\{u_k\}_{k=0}^N$ and $\{y_k\}_{k=0}^N$ are the deterministic control process and the observed output of the system, respectively.

It is not difficult to see that the LTI MIMO framework allows the classical state-space model representation (1), (2) when $D = 0$ and $S = 0$. The KF-like estimator for the LTI MIMO state-space model (4), (5) can be easily derived at the same manner as the classical KF equations. Here, we present a summary of the LTI MIMO estimator in the form of pseudo-code; see [1], [2], for more details.

Algorithm 2. LTI MIMO KF (*conventional algorithm*)

INITIALIZATION: $(k=0)$ $\hat{x}_{0|0} = \bar{x}_0$ and $P_{0|0} = \Pi_0$.
Set $\overline{F} = F - SR^{-1}H$, $\quad \overline{B} = B - SR^{-1}D$,
$\quad \overline{\Theta} = \Theta - SR^{-1}S^T$.
TIME UPDATE: $(k=\overline{1,N})$ $\triangleright$ PRIORI ESTIMATION
$\quad \hat{x}_{k|k-1} = \overline{F}\hat{x}_{k-1|k-1} + \overline{B}u_{k-1} + SR^{-1}y_{k-1}$,
$\quad P_{k|k-1} = \overline{F}P_{k-1|k-1}\overline{F}^T + \overline{\Theta}$.
MEASUREMENT UPDATE: $\triangleright$ POSTERIORI ESTIMATION
$\quad R_{e,k} = HP_{k|k-1}H^T + R$,
$\quad K_k = P_{k|k-1}H^T R_{e,k}^{-1}$,
$\quad e_k = y_k - H\hat{x}_{k|k-1} - Du_k$,
$\quad \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k e_k$,
$\quad P_{k|k} = (I - K_kH)P_{k|k-1}$.

Next, the linear PMMs discussed in [4] are closely related to the LTI MIMO systems described above. In fact, if one assumes that $u_k = y_{k-1}$ (i.e. instead of the control input sequence, one has previously measured data) in the LTI MIMO representation (4), (5), then the corresponding linear Gaussian PMMs in the state-space form is given as follows [6]:

$$
\begin{bmatrix} x_{k+1}\\ y_k \end{bmatrix} = \begin{bmatrix} F & B\\ H & D \end{bmatrix}\begin{bmatrix} x_k\\ y_{k-1} \end{bmatrix} + \begin{bmatrix} w_k\\ v_k \end{bmatrix}, \qquad (6)
$$

$$
\begin{bmatrix} w_k\\ v_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0\\ 0 \end{bmatrix}, \begin{bmatrix} \Theta & S\\ S^T & R \end{bmatrix}\right). \qquad (7)
$$

Thus, linear Gaussian PMMs allow the classical state-space model representation (1), (2) as well. Hence, in the PMMs framework, a HMM-like processing is available, i.e. the PMMs enlarge the HMMs approach; see the discussion in [8]. The KF-like estimator can be derived for the PMMs as shown in [4, Proposition 1]; see also summary in [6]. The resulted filtering method is called the Pairwise Kalman Filter (PKF).

Algorithm 3. PKF (*conventional implementation*)

INITIALIZATION: $(k=0)$ $\hat{x}_{0|0} = \bar{x}_0$ and $P_{0|0} = \Pi_0$.
Set $\overline{F} = F - SR^{-1}H$, $\quad \overline{B} = B - SR^{-1}D$,
$\quad \overline{\Theta} = \Theta - SR^{-1}S^T$.
TIME UPDATE: $(k=\overline{0,N-1})$ $\triangleright$ PRIORI ESTIMATION
$\quad \hat{x}_{k+1|k} = \overline{F}\hat{x}_{k|k} + \overline{B}y_{k-1} + SR^{-1}y_k$,
$\quad P_{k+1|k} = \overline{F}P_{k|k}\overline{F}^T + \overline{\Theta}$.
MEASUREMENT UPDATE: $\triangleright$ POSTERIORI ESTIMATION
$\quad R_{e,k+1} = HP_{k+1|k}H^T + R$,
$\quad K_{k+1} = P_{k+1|k}H^T R_{e,k+1}^{-1}$,
$\quad e_{k+1} = y_{k+1} - H\hat{x}_{k+1|k} - Dy_k$,
$\quad \hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}e_{k+1}$,
$\quad P_{k+1|k+1} = (I - K_{k+1}H)P_{k+1|k}$.

Finally, we would like to stress that the PMMs and LTI MIMO systems examined in this paper imply an *array* form of their state-space representation. This convenient representation makes them preferable for practical use, because it keeps propagation equations explicit (i.e. it does not require time-consuming simulations) and, hence, more suitable for parallel computations as discussed in [15].

III. MAIN RESULT: THE SVD-BASED FILTERING

Definition 1 (see Theorem 1.1.6 in [22]). *Every matrix $A \in \mathbb{C}^{m\times n}$ of rank r can be written as follows:*

$$A = W\Sigma V^*,\ \Sigma = \begin{bmatrix} S & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{C}^{m\times n},\ S = \mathrm{diag}\{\sigma_1,\ldots,\sigma_r\}$$

where $W \in \mathbb{C}^{m\times m}$, $V \in \mathbb{C}^{n\times n}$ are unitary matrices, V^* is the conjugate transpose of V, and $S \in \mathbb{R}^{r\times r}$ is a real nonnegative diagonal matrix. Here $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$ are called the singular values of A. (Note that if $r = n$ and/or $r = m$, some of the zero submatrices in Σ are empty.)

In the KF community, the SVD factorization-based implementations have been recently designed in [16]. They were shown to outperform the conventional KF as well as numerically stable Cholesky- and UD-based KF methods for estimation accuracy and robustness (with respect to roundoff errors). Motivated by these recent findings, we develop the SVD-based LTI MIMO estimator and PKF algorithm. To the best of authors' knowledge, such methods have never been designed before.

First, we stress that each iterate of SVD-based algorithms has the form of $A = \mathfrak{W}\Sigma\mathfrak{V}^T$ where $A \in \mathbb{R}^{(k+s)\times s}$ is given pre-array and the resulted post-array SVD factors are defined as follows: $\mathfrak{W} \in \mathbb{R}^{(k+s)\times(k+s)}$, $\Sigma \in \mathbb{R}^{(k+s)\times s}$ and $\mathfrak{V} \in \mathbb{R}^{s\times s}$. For the initial value of matrix Π_0 the SVD factorization yields $\Pi_0 = Q_{\Pi_0}D_{\Pi_0}Q_{\Pi_0}^T$ where Q_{Π_0} and D_{Π_0} are an orthogonal and a diagonal matrices, respectively. The matrix D_{Π_0} contains the singular values of Π_0. Next, the filter equations are reformulated in terms of the SVD factors $Q_{P_{k|k}}$ and $D_{P_{k|k}}^{1/2}$ instead of processing the entire matrix $P_{k|k}$. Thus, the resulted SVD-based LTI MIMO KF-like algorithm is given as follows:

Algorithm 4. SVD LTI MIMO KF (*SVD-based algorithm*)

INITIALIZATION: $(k = 0)$

1 Apply SVD to $\Pi_0 = Q_{\Pi_0}D_{\Pi_0}Q_{\Pi_0}^T$.

2 Set $\hat{x}_{0|0} = \bar{x}_0$ and $Q_{P_{0|0}} = Q_{\Pi_0}$, $D_{P_{0|0}}^{1/2} = D_{\Pi_0}^{1/2}$.

3 Set $\overline{F} = F - SR^{-1}H$, $\quad \overline{B} = B - SR^{-1}D$,

4 $\quad \overline{\Theta} = \Theta - SR^{-1}S^T$.

5 Apply SVD to $\overline{\Theta} = Q_{\overline{\Theta}}D_{\overline{\Theta}}Q_{\overline{\Theta}}^T$ and $R = Q_RD_RQ_R^T$.

TIME UPDATE: $(k = \overline{1,N})$ $\triangleright$ PRIORI ESTIMATION

6 $\quad \hat{x}_{k|k-1} = \overline{F}\hat{x}_{k-1|k-1} + \overline{B}u_{k-1} + SR^{-1}y_{k-1}$,

7 $\quad$ Assemble the pre-array and apply SVD as follows:

$$\underbrace{\begin{bmatrix} D_{P_{k-1|k-1}}^{1/2}Q_{P_{k-1|k-1}}^T\overline{F}^T \\ D_{\overline{\Theta}}^{1/2}Q_{\overline{\Theta}}^T \end{bmatrix}}_{\rm Pre-array} = \underbrace{\mathfrak{V}\begin{bmatrix} D_{P_{k|k-1}}^{1/2} \\ 0 \end{bmatrix}Q_{P_{k|k-1}}^T}_{\rm Post-array\ SVD\ factors},$$

8 $\quad$ Read-off from the post-arrays: $Q_{P_{k|k-1}}$, $D_{P_{k|k-1}}^{1/2}$.

MEASUREMENT UPDATE: $\triangleright$ POSTERIORI ESTIMATION

9 $\quad$ Assemble the pre-array and apply SVD as follows:

$$\underbrace{\begin{bmatrix} D_R^{1/2}Q_R^T \\ D_{P_{k|k-1}}^{1/2}Q_{P_{k|k-1}}^TH^T \end{bmatrix}}_{\rm Pre-array} = \underbrace{\mathfrak{W}\begin{bmatrix} D_{R_{e,k}}^{1/2} \\ 0 \end{bmatrix}Q_{R_{e,k}}^T}_{\rm Post-array\ SVD\ factors},$$

10 $\quad$ Read-off from the post-arrays: $Q_{R_{e,k}}$ and $D_{R_{e,k}}^{1/2}$,

11 $\quad \bar{K}_k = (Q_{P_{k|k-1}}D_{P_{k|k-1}}Q_{P_{k|k-1}}^T)H^TQ_{R_{e,k}}$,

12 $\quad e_k = y_k - H\hat{x}_{k|k-1} - Du_k$, $\quad \bar{e}_k = Q_{R_{e,k}}^Te_k$,

13 $\quad \hat{x}_{k|k} = \hat{x}_{k|k-1} + \bar{K}_kD_{R_{e,k}}^{-1}\bar{e}_k$,

14 $\quad$ Assemble the pre-array and apply SVD as follows:

$$\underbrace{\begin{bmatrix} D_{P_{k|k-1}}^{1/2}Q_{P_{k|k-1}}^TA^T \\ D_R^{1/2}Q_R^TK_k^T \end{bmatrix}}_{\rm Pre-array} = \underbrace{\mathfrak{Q}\begin{bmatrix} D_{P_{k|k}}^{1/2} \\ 0 \end{bmatrix}Q_{P_{k|k}}^T}_{\rm Post-array\ SVD\ factors}$$

$\quad$ where $A = (I - K_kH)$ and $K_k = \bar{K}_kD_{R_{e,k}}^{-1}Q_{R_{e,k}}^T$,

15 $\quad$ Read-off from the post-arrays: $Q_{P_{k|k}}$ and $D_{P_{k|k}}^{1/2}$.

As can be seen, instead of conventional LTI MIMO recursion (Algorithm 2) for $P_{k|k}$, Algorithm 4 propagates the SVD factors of this matrix, i.e. the quantities $\{Q_{P_{k|k}}, D_{P_{k|k}}^{1/2}\}$. This strategy improves the estimation quality and numerical robustness when the error covariance matrix is ill-conditioned; see numerical results in [16].

To justify the method in Algorithm 4, one should take into account that $A = \mathfrak{W}\Sigma\mathfrak{V}^T$, where $\mathfrak{W}$ and $\mathfrak{V}$ are orthogonal matrices, and hence $A^TA = (\mathfrak{V}\Sigma\mathfrak{W}^T)(\mathfrak{W}\Sigma\mathfrak{V}^T) = \mathfrak{V}\Sigma^2\mathfrak{V}^T$ for each filtering pre-array to be factorized. Thus, by comparing both sides of the obtained matrix equalities in Algorithm 4, the algebraic equivalence between the new SVD-based equations (Algorithm 4) and the conventional LTI MIMO KF-like formulas (Algorithm 2) is proved. More precisely, from line 7 of Algorithm 4, we get

$$\begin{aligned} P_{k|k-1} &= Q_{P_{k|k-1}}D_{P_{k|k-1}}Q_{P_{k|k-1}}^T \\ &= \overline{F}Q_{P_{k-1|k-1}}D_{P_{k-1|k-1}}Q_{P_{k-1|k-1}}^T\overline{F}^T \\ &\quad + Q_{\overline{\Theta}}D_{\overline{\Theta}}Q_{\overline{\Theta}}^T = \overline{F}P_{k-1|k-1}\overline{F}^T + \overline{\Theta}, \end{aligned}$$

which is exactly formula in line 4 of Algorithm 2.

Next, in line 9 of Algorithm 4 we have

$$\begin{aligned} R_{e,k} &= Q_{R_{e,k}}D_{R_{e,k}}Q_{R_{e,k}}^T \\ &= Q_RD_RQ_R^T + HQ_{P_{k|k-1}}D_{P_{k|k-1}}Q_{P_{k|k-1}}^TH^T \\ &= R + HP_{k|k-1}H^T, \end{aligned}$$

i.e. formula in line 9 of Algorithm 4 implies equation $R_{e,k} = HP_{k|k-1}H^T + R$ in line 5 of Algorithm 2.

Expression in line 14 of Algorithm 4 for calculating the feedback gain K_k and its normalized variant $\bar{K}_k$ in line 11 is derived by taking into account that the matrix $R_{e,k}$ is the SVD factorized. Indeed,

$$\begin{aligned} K_k &= P_{k|k-1}H^T\left(Q_{R_{e,k}}D_{R_{e,k}}Q_{R_{e,k}}^T\right)^{-1} \\ &= P_{k|k-1}H^TQ_{R_{e,k}}D_{R_{e,k}}^{-1}Q_{R_{e,k}}^T = \bar{K}_kD_{R_{e,k}}^{-1}Q_{R_{e,k}}^T \end{aligned}$$

where the following notation is introduced for the normalized feedback gain: $\bar{K}_k = P_{k|k-1}H^TQ_{R_{e,k}}$.

Furthermore, from the SVD factorization in line 14 of Algorithm 4, we obtain

$$\begin{aligned} P_{k|k} &= Q_{P_{k|k}}D_{P_{k|k}}Q_{P_{k|k}}^T \\ &= (I - K_kH)\,Q_{P_{k|k-1}}D_{P_{k|k-1}}Q_{P_{k|k-1}}^T\,(I - K_kH)^T \\ &\quad + K_kQ_RD_RQ_R^TK_k^T, \end{aligned}$$

i.e. $P_{k|k} = (I - K_k H) P_{k|k-1} (I - K_k H)^T + K_k R K_k^T$, which is equivalent to equation in line 9 of Algorithm 2 that is $P_{k|k} = (I - K_k H) P_{k|k-1}$; see the proof in [23, p. 128].

Finally, equation in line 13 of Algorithm 4 for computing *a posteriori* state estimate is derived as follows:

$$\begin{aligned}\hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k e_k = \hat{x}_{k|k-1} + \bar{K}_k D^{-1}_{R_{e,k}} Q^T_{R_{e,k}} e_k \\ &= \hat{x}_{k|k-1} + \bar{K}_k D^{-1}_{R_{e,k}} \bar{e}_k \quad \text{where} \quad \bar{e}_k = Q^T_{R_{e,k}} e_k.\end{aligned}$$

This concludes the proof. Thus, the new SVD-based implementation (Algorithm 4) is shown to be algebraically equivalent to the conventional LTI MIMO KF-like method (Algorithm 2).

Similarly, we derive the SVD-based PKF implementation.

Algorithm 5. SVD-PKF (*SVD-based algorithm*)

INITIALIZATION: $(k = 0)$

Apply SVD to $\Pi_0 = Q_{\Pi_0} D_{\Pi_0} Q^T_{\Pi_0}$.

Set $\hat{x}_{0|0} = \bar{x}_0$ and $Q_{P_{0|0}} = Q_{\Pi_0}$, $D^{1/2}_{P_{0|0}} = D^{1/2}_{\Pi_0}$.

Set $\overline{F} = F - SR^{-1}H$, $\quad \overline{B} = B - SR^{-1}D$,

$\quad \overline{\Theta} = \Theta - SR^{-1}S^T$.

Apply SVD to $\overline{\Theta} = Q_{\overline{\Theta}} D_{\overline{\Theta}} Q^T_{\overline{\Theta}}$ and $R = Q_R D_R Q_R^T$.

TIME UPDATE: $(k = \overline{0, N-1})$ $\triangleright$ PRIORI ESTIMATION

$\quad \hat{x}_{k+1|k} = \overline{F}\hat{x}_{k|k} + \overline{B} y_{k-1} + SR^{-1} y_k$,

$\quad$ Assemble the pre-array and apply SVD as follows:

$$\underbrace{\begin{bmatrix} D^{1/2}_{P_{k|k}} Q^T_{P_{k|k}} \overline{F}^T \\ D^{1/2}_{\overline{\Theta}} Q^T_{\overline{\Theta}} \end{bmatrix}}_{\text{Pre-array}} = \mathfrak{W} \underbrace{\begin{bmatrix} D^{1/2}_{P_{k+1|k}} \\ 0 \end{bmatrix} Q^T_{P_{k+1|k}}}_{\text{Post-array SVD factors}},$$

$\quad$ Read-off from the post-arrays: $Q_{P_{k+1|k}}$, $D^{1/2}_{P_{k+1|k}}$.

MEASUREMENT UPDATE: $\triangleright$ POSTERIORI ESTIMATION

$\quad$ Assemble the pre-array and apply SVD as follows:

$$\underbrace{\begin{bmatrix} D^{1/2}_R Q^T_R \\ D^{1/2}_{P_{k+1|k}} Q^T_{P_{k+1|k}} H^T \end{bmatrix}}_{\text{Pre-array}} = \mathfrak{V} \underbrace{\begin{bmatrix} D^{1/2}_{R_{e,k+1}} \\ 0 \end{bmatrix} Q^T_{R_{e,k+1}}}_{\text{Post-array SVD factors}},$$

$\quad$ Read-off from the post-arrays: $Q_{R_{e,k+1}}$, $D^{1/2}_{R_{e,k+1}}$,

$\quad \bar{K}_{k+1} = (Q_{P_{k+1|k}} D_{P_{k+1|k}} Q^T_{P_{k+1|k}}) H^T Q_{R_{e,k+1}}$,

$\quad e_{k+1} = y_{k+1} - H\hat{x}_{k+1|k} - D y_k$,

$\quad \bar{e}_{k+1} = Q^T_{R_{e,k+1}} e_{k+1}$,

$\quad \hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + \bar{K}_{k+1} D^{-1}_{R_{e,k+1}} \bar{e}_{k+1}$,

$\quad$ Assemble the pre-array and apply SVD as follows:

$$\underbrace{\begin{bmatrix} D^{1/2}_{P_{k+1|k}} Q^T_{P_{k+1|k}} A^T \\ D^{1/2}_R Q^T_R K^T_{k+1} \end{bmatrix}}_{\text{Pre-array}} = \mathfrak{Q} \underbrace{\begin{bmatrix} D^{1/2}_{P_{k+1|k+1}} \\ 0 \end{bmatrix} Q^T_{P_{k+1|k+1}}}_{\text{Post-array SVD factors}}$$

$\quad$ where $A = (I - K_{k+1} H)$

$\quad$ and $K_{k+1} = \bar{K}_{k+1} D^{-1}_{R_{e,k+1}} Q^T_{R_{e,k+1}}$,

$\quad$ Read-off $Q_{P_{k+1|k+1}}$ and $D^{1/2}_{P_{k+1|k+1}}$.

The algebraic equivalence between the new SVD-based PKF (Algorithm 5) and the conventional PKF (Algorithm 3) can be proved at the same manner as shown above for the LTI MIMO KF-like estimator (Algorithms 2 and 4).

Finally, the log LF formula (3) can be expressed in terms of the SVD factors of $R_{e,k}$ as follows [16]:

$$-\ln L(\theta|Y_N) = c_0 + \frac{1}{2}\sum_{k=1}^{N}\left\{\ln\left(\det D_{R_{e,k}}\right) + \bar{e}_k^T D^{-1}_{R_{e,k}} \bar{e}_k\right\}$$

where c_0 is a constant term.

IV. NUMERICAL EXPERIMENTS

First, we would like to check the theoretical derivations in Section III on practical examples.

Example 1. *Consider the pupil tracking problem where the corresponding state-space model is given in linear Gaussian PMM representation* (6), (7) *as follows [24]:*

$$F = \begin{bmatrix} 1.78 & -0.01 & 0.38 \\ -0.04 & 1.52 & 0.16 \\ -0.01 & 0.01 & 0.66 \end{bmatrix}, B = \begin{bmatrix} -0.83 & 0.04 & -0.33 \\ 0.04 & -0.58 & -0.15 \\ -0.02 & -0.01 & 0.34 \end{bmatrix}$$

where $H = I_3$, $\Theta = R = I_3$, $D = 0_3$ *and* $S = 0_3$. *The matrices* I_3 *and* 0_3 *denote the identity and zero matrices of size* 3×3, *respectively. Following [24], the initial values are* $\bar{x}_0 = [180, 80, 20]^T$ *and* $\Pi_0 = 0.1 I_3$.

To recover hidden state process, $\{x_k\}$, the PKF estimator is applied. We examine the conventional PKF method (Algorithm 3) and its robust *factored-form* implementations:

- `SR-PKF` (the Cholesky-based method) designed in [6];
- `UD-PKF` (the UD-based method) developed in [17];
- the newly proposed `SVD-PKF` (Algorithm 5);

The following set of numerical experiments is performed. The system is simulated for $k = 1, \ldots, N$ with $N = 50$ discrete time points. Then, the "true" trajectory of the dynamic state x_k^{exact} and the related measurements y_k are generated, $k = 1, \ldots, N$. Next, the examined filtering methods are applied for solving the inverse problem: given the simulated measurements, each filter under assessment yields the state vector estimate $\hat{x}_{k|k}$, $k = 1, \ldots, N$. For a fair comparative study, the same filtering initial values, the same system matrices and the same measurements are passed to all PKF estimators listed above. The outlined experiment is repeated for $M = 1000$ Monte-Carlo trials. All codes are written in MATLAB. The root mean square error (RMSE) is computed for each component of the state vector. Together with the RMSE_x, the accumulated CPU time (s) for each estimator is reported in Table I.

Having analyzed the obtained numerical results presented in Table I, we conclude that all estimators work with the same accuracy. Hence, the theoretical derivations presented in Section III and algebraic equivalence between four examined PKF implementations are confirmed in practice. The accumulated CPU time is higher for the *factored-form* implementations, i.e. for the `SR-PKF`, `UD-PKF` and new `SVD-PKF`, compared to the original PKF algorithm. This is in line with the classical KF theoretical results. Comparing the *factored-form* PKF implementations, we observe that their accumulated CPU time consumptions are almost of the same value for this low dimension problem.

TABLE I
THE RMSE AND CPU TIME (S) IN EXAMPLE 1, $M = 1000$ RUNS.

Method	RMSE_{x_i}			$\|RMSE_x\|_2$	CPU
	x_1	x_2	x_3		
PKF	2.0932	0.9077	0.7386	2.3981	3.79
SR-PKF	2.0932	0.9077	0.7386	2.3981	6.31
UD-PKF	2.0932	0.9077	0.7386	2.3981	6.22
SVD-PKF	2.0932	0.9077	0.7386	2.3981	6.34

TABLE II
IMPACT OF ROUNDOFF ON PKF IMPLEMENTATIONS IN [17, EXAMPLE 2].

Method	growing ill-conditioning $\delta \to \epsilon_{roundoff}$					
	10^{-3}	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}
PKF	0.1878	0.2633	0.1978	0.2941	0.1902	NaN
SR-PKF	0.1878	0.2633	0.1978	0.2941	0.1902	0.1703
UD-PKF	0.1878	0.2633	0.1978	0.2941	0.1902	0.1702
SVD-PKF	0.1878	0.2633	0.1978	0.2941	0.1902	0.1702

Next, to explore the effect of roundoff errors on the examined PKF estimators, we consider the ill-conditioned test problem proposed in [17, Example 2, p. 1623]. We repeat the numerical experiments outlined above for $M = 100$ Monte Carlo runs and various values of ill-conditioning parameter δ such that $\delta \to \epsilon_{roundoff}$, where $\epsilon_{roundoff}$ is the machine precision limit. The source of numerical instability in this example is in the residual covariance matrix inversion, because $R_{e,k}$ tends to a singular matrix; see the third reason of ill-conditioning of the classical KF discussed in [12, p. 288]. For $\delta \to \epsilon_{roundoff}$, we report $\|RMSE_x\|_2$ in Table II.

Our analysis suggests that the *factored-form* PKF implementations, i.e. the SR-PKF, UD-PKF and new SVD-PKF, provide a better estimation quality and accuracy in ill-conditioned situations, compared to the conventional PKF. In fact, the conventional PKF algorithm degrades faster than any other SR implementation under examination, and fails at $\delta = 10^{-8}$. Among the *factored-form* PKF implementations, the previously published UD- and the newly-developed SVD-based techniques slightly outperform the Cholesky-based counterpart, i.e. they provide the best estimation quality when solving ill-conditioned state estimation problem.

V. APPLICATION IN ECONOMETRICS

In this section the new robust SVD factorization-based estimation strategy is applied to the Russia Trading System Index (RTSI) for observing the weak-form market efficiency from 1 October, 1997 to 1 October 2017, i.e. for the last 20 years. Following [25], a market is weak-form efficient when there is no predictable profit opportunity based on the past movement of asset prices (an efficient market is unpredictable). One of the most common and simple tests for the presence of weak-form efficiency is to see if the returns process follows a random walk. This means that historical price information cannot provide profit opportunities. A more sophisticated tests are also exist in the econometrics literature. Here, we follow the Bayesian methodology and utilize the test for evolving efficiency (TEE) developed in [18]. This econometric model integrates time-varying regression coefficients with the underlying time-varying variance (volatility) process given by GARCH-in-Mean(1,1) specification. For this complicated model structure the classical state-space representation (1), (2) might be restrictive, while the LTI MIMO and PMMs may provide a better fit. The TEE is given as follows [18]:

$$y_k = \beta_{0,k} + \beta_{1,k} y_{k-1} + \delta h_k + \varepsilon_k, \quad \varepsilon_k \sim \mathcal{N}(0, h_k), \quad (8)$$

$$h_k = a_0 + a_1 \varepsilon_{k-1}^2 + b_1 h_{k-1}, \quad (9)$$

$$\beta_{i,k} = \beta_{i,k-1} + w_{i,k}, \quad i = 0, 1, \quad w_{i,k} \sim \mathcal{N}(0, \sigma_i^2) \quad (10)$$

where $a_0 > 0$, $a_1 \geq 0$, $b_1 \geq 0$ and $a_1 + b_1 < 1$ ensures that the GARCH(1,1) process in equation (9) is stationary. The process $\{y_k\}_{k=0}^N$ is the returns process and ε_k denotes the error term (i.e. the return residuals). The process $\{h_k\}_{k=0}^N$ is the hidden volatility process, which needs to be estimated from the available data $\{y_k\}_{k=0}^N$. The core result of the TEE methodology is the time-varying slope coefficient $\beta_{1,k}$. More precisely, the *evolution* of $\beta_{1,k}$ reflects the time-varying change in weak-form market efficiency. If $\beta_{1,k}$ equals to zero (within its confidence interval), then the TEE yields the conclusion that the market is weak-form efficient.

Motivated by [26], [27], the TEE estimation is done by the state-space approach as discussed in [18]. In model (8) – (10), the term ε_k is the return residual, i.e. $\varepsilon_k = y_k - \beta_{0,k} - \beta_{1,k} y_{k-1} - \delta h_k$; see equation (8). As discussed in [27], at time instance t_k the return residual ε_{k-1} is known. Hence, the state-space approach studied in [27] suggests to consider the value ε_{k-1} in equation (9) for computing h_k as a known control input. In summary, the TEE specification can be written in convenient LTI MIMO form as follows:

$$\begin{bmatrix} h_{k+1} \\ \beta_{0,k+1} \\ \beta_{1,k+1} \\ y_k \end{bmatrix} = \begin{bmatrix} \underbrace{\begin{bmatrix} b_1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{F} & \underbrace{\begin{bmatrix} a_0 & a_1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}}_{B} \\ \underbrace{\begin{bmatrix} \delta & 1 & y_{k-1} \end{bmatrix}}_{H} & \underbrace{\begin{bmatrix} 0 & 0 \end{bmatrix}}_{D} \end{bmatrix} \begin{bmatrix} h_k \\ \beta_{0,k} \\ \beta_{1,k} \\ 1 \\ \hat{\varepsilon}^2_{k|k-1} \end{bmatrix} + \begin{bmatrix} w_k \\ \varepsilon_k \end{bmatrix}$$

where $\hat{\varepsilon}_{k|k-1} = y_k - \hat{\beta}_{0,k|k-1} - \hat{\beta}_{1,k|k-1} y_{k-1} - \delta \hat{h}_{k|k-1}$ is the known return residual at time t_k. The state and measurement uncertainty processes $\{w_k\}$ and $\{\varepsilon_k\}$ are independent Gaussian zero-mean white-noise processes with covariance matrices $\Theta = \text{diag}\{0, \sigma_0^2, \sigma_1^2\}$ and $R_k = \hat{h}_{k|k-1}$, respectively.

When the econometric model is casted into the state-space form, the corresponding filtering method is used for estimating the unknown dynamic state. For the examined model the SVD-based LTI MIMO KF (Algorithm 4) is applied for estimating $x_k = [h_k, \beta_{0,k}, \beta_{1,k}]$ and for calculating the likelihood function to obtain the maximum likelihood estimate of unknown system parameters $\theta = [a_0, a_1, b_1, \delta, \sigma_0^2, \sigma_1^2]$.

Recall, the RTSI is a free-float capitalization-weighted index of 50 Russian stocks traded on the Moscow Exchange, calculated in the US dollars. The index was introduced on September 1, 1995. We use daily log-returns calculated on a continuously compounded basis and expressed in percentages, i.e. $y_k = 100(\ln S_k - \ln S_{k-1})$, where $\{S_k\}$ are the closing prices[1] within the examined period. The estimated TEE model is given as follows:

$$\begin{aligned} y_k &= \beta_{0,k} + \beta_{1,k} y_{k-1} - 0.0041 h_k + \varepsilon_k \\ h_k &= 0.0978 + 0.1204 \varepsilon_{k-1}^2 + 0.8657 h_{k-1} \\ \beta_{0,k} &= \beta_{0,k-1} + w_{0,k} \text{ where } \sigma_0^2 = 0.000001 \\ \beta_{1,k} &= \beta_{1,k-1} + w_{1,k} \text{ where } \sigma_1^2 = 0.000008 \end{aligned}$$

The time evolution of β_1 coefficient reflects the changes in weak-form market efficiency. The unknown state vector $x_k = [h_k, \beta_{0,k}, \beta_{1,k}]$ is estimated by the proposed SVD-based filtering technique. The obtained results for $\beta_{1,k}$ coefficient (with 95% confidence interval) and the daily log-returns are illustrated by Fig. 1.

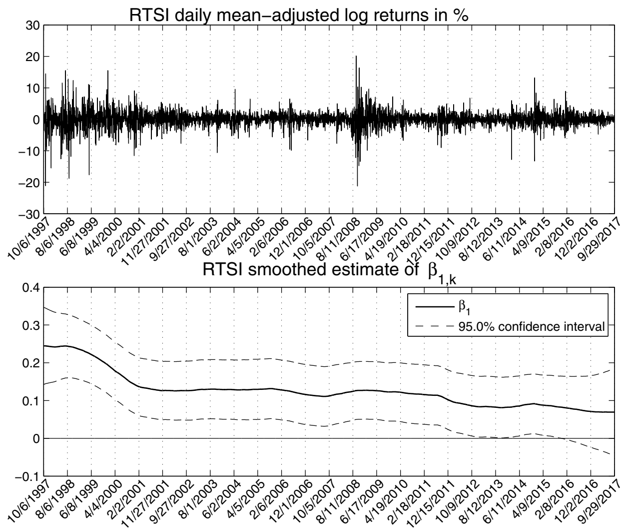

Fig. 1. Daily log-returns and results of the TEE for the RTSI

Having analyzed the bottom graph on Fig. 1, we conclude that the Russian market is *inefficient* at 95% confidence level for almost entire time period to be explored. Since the beginning of the examined period and until February 2016, the 95% confidence interval for $\beta_{1,k}$ does not contain the critical value of zero, implying that the market is weak-form inefficient. However, it is readily seen that the market gradually tends to weak-form efficiency throughout the time period under examination. The 95% confidence interval may indicate some degree of efficiency in the period from October 2012 to June 2014. At that period of time, the lower bound of the 95% confidence interval touches the critical level of zero. It is clear that the market becomes weak-form efficient only since February 2016, when the estimated time-path of $\beta_{1,k}$ for the RTSI is strongly within the 95% confidence interval.

References

[1] S. Gibson and B. Ninness, "Robust maximum-likelihood estimation of multivariable dynamic systems," *Automatica*, vol. 41, no. 10, pp. 1667–1682, Oct. 2005.

[2] A. Wills and B. Ninness, "On gradient-based search for multivariable system estimates," *IEEE Transactions on Automatic Control*, vol. 53, no. 1, pp. 298–306, Jan. 2008.

[3] S. Gibson, A. Wills, and B. Ninness, "Maximum-likelihood parameter estimation of bilinear systems," *IEEE Transactions on Automatic Control*, vol. 50, no. 10, pp. 1581–1596, Oct 2005.

[4] W. Pieczynski and F. Desbouvries, "Kalman filtering using pairwise Gaussian models," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Hong-Kong*, vol. VI, 2003, pp. 57–60.

[5] A. el Fquih B. and F. Desbouvries, "Unsupervised signal restoration in partially observed Markov chains," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Toulouse, France*, vol. 3, 2006, p. III.

[6] V. Némesin and S. Derrode, "Robust blind pairwise Kalman algorithms using QR decompositions," *IEEE Transactions on Signal Processing*, vol. 61, no. 1, pp. 5–9, Jan. 2013.

[7] ——, "Robust partial-learning in linear Gaussian systems," *IEEE Transactions on Automatic Control*, vol. 60, no. 9, pp. 2518–2523, Sep. 2015.

[8] I. Gorynin, L. Crelier, H. Gangloff, E. Monfrini, and W. Pieczynski, "Performance comparison across hidden, pairwise and triplet Markov models' estimators," *International Journal of Mathematical and Computational Methods*, vol. 1, pp. 253–258, 2016.

[9] P. Lanchantin, J. Lapuyade-Lahorgue, and W. Pieczynski, "Unsupervised segmentation of randomly switching data hidden with non-Gaussian correlated noise," *Signal Processing*, vol. 91, no. 2, pp. 163–175, 2011.

[10] M. Verhaegen and P. Van Dooren, "Numerical aspects of different Kalman filter implementations," *IEEE Trans. Automat. Contr.*, vol. AC-31, no. 10, pp. 907–917, Oct. 1986.

[11] M. S. Grewal and J. Kain, "Kalman filter implementation with improved numerical properties," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2058–2068, 2010.

[12] M. S. Grewal and A. P. Andrews, *Kalman filtering: theory and practice using MATLAB*, 4th ed. New Jersey: Prentice Hall, 2015.

[13] P. G. Kaminski, A. E. Bryson, and S. F. Schmidt, "Discrete square-root filtering: a survey of current techniques," *IEEE Trans. Automat. Contr.*, vol. AC-16, no. 6, pp. 727–735, Dec. 1971.

[14] G. J. Bierman, *Factorization Methods For Discrete Sequential Estimation*. New York: Academic Press, 1977.

[15] P. Park and T. Kailath, "New square-root algorithms for Kalman filtering," *IEEE Transactions on Automatic Control*, vol. 40, no. 5, pp. 895–899, May 1995.

[16] M. V. Kulikova and J. V. Tsyganova, "Improved discrete-time Kalman filtering within singular value decomposition," *IET Control Theory & Applications*, vol. 11, no. 15, pp. 2412–2418, 2017.

[17] M. V. Kulikova and J. V. Tsyganova, "The UD-based approach for designing pairwise Kalman filtering algorithms," *IFAC-PapersOnLine*, vol. 50, no. 1, 2017, pp. 1619–1624.

[18] A. Zalewska-Mitura and S. Hall, "Examining the first stages of market performance: A test for evolving market efficiency," *Economics Letters*, vol. 64, no. 1, pp. 1–12, 1999.

[19] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*. New Jersey: Prentice Hall, 2000.

[20] F. C. Schweppe, "Evaluation of likelihood functions for Gaussian signals," *IEEE Trans. Inf. Theory*, vol. IT-11, pp. 61–70, Jan. 1965.

[21] A. Harvey, *Forecasting Structural Time Series Models and the Kalman Filter*. Cambridge: Cambridge University Press, 1989.

[22] A. Björck, *Numerical methods in matrix computations*. Springer, 2015.

[23] D. Simon, *Optimal state estimation: Kalman, H-infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

[24] V. Némesin and S. Derrode, "Inferring segmental pairwise Kalman filter with application to pupil tracking," in *Traitement et Analyse de l'Information - Méthodes et Applications (TAIMA), Hammamet (Tunisie), 8-13 May*, 2013.

[25] E. Fama, "The behavior of stock-market prices," *The Journal of Business*, vol. 38, no. 1, pp. 34–105, 1965.

[26] S. Hall, "A note on the estimation of GARCH-M models using the Kalman filter," *Bank of England Discussion Paper*, no. 32, 1990.

[27] ——, "An application of the stochastic GARCH-in-Mean model to risk premia in the London Metal Exchange," *The Manchester School*, vol. 59, no. S1, pp. 57–71, 1991.

[1]The data set can be freely downloaded from https://www.finam.ru.