

Supplementary materials to “Adaptive cointegration analysis and principal component analysis with continual learning ability for monitoring multimode nonstationary processes”

1. NUMERICAL CASE TO EXPLAIN FIG. 1

The data for Fig. 1 in Section II.B are generated by:

Mode \mathcal{M}_1 :

$$\begin{cases} z_1 = 1.5t + 3 + e_1 \\ z_2 = t + 2.5 + e_2 \\ z_3 = -1.2t + 4.5 + e_3 \\ z_4 = 1.5 + e_4 \\ z_5 = 2.2 + e_5 \\ z_6 = 1.6t^2 - 0.1t + 0.2 \sin t + 0.8 + e_6 \\ z_7 = 0.1\sqrt{t} + 0.2t + 0.3 + e_7 \end{cases}$$

Mode \mathcal{M}_2 :

$$\begin{cases} z_1 = 1.5t + 3 + e_1 \\ z_2 = 2t + 2.5 + e_2 \\ z_3 = -0.7t + 4.5 + e_3 \\ z_4 = 1.7 + e_4 \\ z_5 = 0.9 + e_5 \\ z_6 = 0.6t^2 - 0.1t + 0.2 \sin t + 0.8 + e_6 \\ z_7 = 0.3\sqrt{t} + 0.3t + 0.3 + e_7 \end{cases}$$

where the noise $e_i \sim N(0, 0.01)$, $i = 1, \dots, 7$. t is piecewise linear and follows uniform distribution with $t \sim U([0, 4.99])$.

2. COMPUTATION OF ΔA_{k+1} AND ΔB_{k+1}

We illustrate the computation of ΔA_{k+1} and ΔB_{k+1} . The component $D_{k+1}^T E_{1,k}$ of $\Delta A_{1,k+1}$ is taken as an example to illustrate the recursion calculation manner.

$$D_{k+1}^T E_{1,k} = d_{k+1}^T \Delta x_{k+1}^p J_k^T E_{1,k} \quad (\text{S1})$$

If we compute (S1) directly, the computational complexity is $O(kp^2m_1^2)$ and increases continually. We should compute $J_k^T E_{1,k}$ recursively.

$$\begin{aligned} J_k^T E_{1,k} &= \begin{bmatrix} J_{k-1} \tilde{J}_k \\ \Delta x_k^p R_{k-1} \tilde{J}_k \end{bmatrix}^T \begin{bmatrix} E_{1,k-1} - J_{k-1} (\Delta x_k^p)^T h_k \\ h_k \end{bmatrix} \\ &= \tilde{J}_k^T J_{k-1}^T (E_{1,k-1} - J_{k-1} (\Delta x_k^p)^T h_k) + (\Delta x_k^p R_{k-1} \tilde{J}_k)^T h_k \\ &= \tilde{J}_k^T J_{k-1}^T E_{1,k-1} - \tilde{J}_k^T J_{k-1}^T J_{k-1} (\Delta x_k^p)^T h_k + \tilde{J}_k^T R_{k-1}^T (\Delta x_k^p)^T h_k \end{aligned} \quad (\text{S2})$$

The recursion of $J_k^T J_k$ is:

$$J_k^T J_k = \tilde{J}_k^T J_{k-1}^T J_{k-1} \tilde{J}_k + \tilde{J}_k^T R_{k-1}^T (\Delta x_k^p)^T \Delta x_k^p R_{k-1} \tilde{J}_k \quad (\text{S3})$$

As $\tilde{J}_k = I - \frac{(\Delta x_k^p)^T \Delta x_k^p R_{k-1}}{1+c_k}$, $D_{k+1}^T E_{1,k}$ can be calculated recursively based on (S1)-(S3). Similarly, other components of ΔA_{k+1} and ΔB_{k+1} can also be computed recursively.

The computational complexity of $\tilde{J}_k^T J_{k-1}^T J_{k-1} \tilde{J}_k$ is $O(p^2m_1^2)$ based on the associative law of multiplication. As Δx_k^p is a vector, the computational complexity of (S1) is $O(p^2m_1^2)$ by reasonable arrangement of matrix calculation order. It is understandable because each component contains at least one vector. Similarly, other components of ΔA_{k+1} and ΔB_{k+1} need $O(p^2m_1^2)$. In conclusion, the computational complexity of ΔA_{k+1} and ΔB_{k+1} is $O(p^2m_1^2)$ per update, which is independent of the number of existing samples.

3. PROOF OF THEOREM 1

Theorem 1. Matrix \mathbf{K} can be calculated recursively in the form of $\mathbf{K}_{k+1} = \begin{bmatrix} \mathbf{G}_{1,k+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{2,k+1} \end{bmatrix} \mathbf{K}_k$, where $\mathbf{G}_{1,k+1}$ and $\mathbf{G}_{2,k+1}$ are adaptively determined.

Proof:

$$\mathbf{K}_{k+1} = \mathbf{B}_{k+1}^{-\frac{1}{2}} = (\alpha_{k+1} \mathbf{B}_k + (1 - \alpha_{k+1}) \Delta \mathbf{B}_{k+1})^{-\frac{1}{2}} \quad (\text{S4})$$

As \mathbf{B} is a block diagonal matrix with $\mathbf{B} = \begin{bmatrix} \mathbf{B}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_2 \end{bmatrix}$, \mathbf{K} can be written in the form of $\mathbf{K} = \begin{bmatrix} \mathbf{K}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2 \end{bmatrix}$. We mainly introduce the recursion of \mathbf{K}_1 , and \mathbf{K}_2 can be updated similarly.

$$\begin{aligned} \mathbf{K}_{1,k+1} &= \mathbf{B}_{1,k+1}^{-\frac{1}{2}} \\ &= (\alpha_{k+1} \mathbf{B}_{1,k} + (1 - \alpha_{k+1}) \Delta \mathbf{B}_{1,k+1})^{-\frac{1}{2}} \\ &= \left(\left(\sqrt{\alpha_{k+1}} \mathbf{B}_{1,k}^{\frac{1}{2}} \right) \left(\mathbf{I} + \frac{1 - \alpha_{k+1}}{\alpha_{k+1}} \mathbf{B}_{1,k}^{-\frac{1}{2}} \Delta \mathbf{B}_{1,k+1} \left(\mathbf{B}_{1,k}^{-\frac{1}{2}} \right)^T \right) \left(\sqrt{\alpha_{k+1}} \mathbf{B}_{1,k}^{\frac{1}{2}} \right)^T \right)^{-\frac{1}{2}} \\ &= \frac{1}{\sqrt{\alpha_{k+1}}} \left(\mathbf{I} + \frac{1 - \alpha_{k+1}}{\alpha_{k+1}} \mathbf{B}_{1,k}^{-\frac{1}{2}} \Delta \mathbf{B}_{1,k+1} \left(\mathbf{B}_{1,k}^{-\frac{1}{2}} \right)^T \right)^{-\frac{1}{2}} \mathbf{B}_{1,k}^{-\frac{1}{2}} \\ &= \frac{1}{\sqrt{\alpha_{k+1}}} \left(\mathbf{I} + \frac{1 - \alpha_{k+1}}{\alpha_{k+1}} \mathbf{K}_{1,k} \Delta \mathbf{B}_{1,k+1} \mathbf{K}_{1,k}^T \right)^{-\frac{1}{2}} \mathbf{K}_{1,k} \\ &= \frac{1}{\sqrt{\alpha_{k+1}}} \tilde{\mathbf{K}}_{1,k+1}^{-\frac{1}{2}} \mathbf{K}_{1,k} \end{aligned} \quad (\text{S5})$$

where $\tilde{\mathbf{K}}_{1,k+1} = \mathbf{I} + \frac{1 - \alpha_{k+1}}{\alpha_{k+1}} \mathbf{K}_{1,k} \Delta \mathbf{B}_{1,k+1} \mathbf{K}_{1,k}^T$. $\Delta \mathbf{B}_{1,k+1}$ is symmetric and the rank is no more than 2. Thus, it can be reformulated into

$$\begin{aligned} \Delta \mathbf{B}_{1,k+1} &= [\mathbf{q}_{1,k+1} \quad \mathbf{q}_{2,k+1}] \begin{bmatrix} \beta_{1,k+1} & \\ & \beta_{2,k+1} \end{bmatrix} [\mathbf{q}_{1,k+1} \quad \mathbf{q}_{2,k+1}]^T \\ &= \mathbf{Q}_{1,k+1} \Xi_{1,k+1} \mathbf{Q}_{1,k+1}^T \end{aligned}$$

where $\beta_{1,k+1}$ and $\beta_{2,k+1}$ are non-zero eigenvalues if $\text{rank}(\Delta \mathbf{B}_{1,k+1}) = 2$. $\mathbf{q}_{1,k+1}$ and $\mathbf{q}_{2,k+1}$ are the eigenvectors. If the rank is 1, $\beta_{1,k+1}$ or $\beta_{2,k+1}$ is 0. Thus,

$$\begin{aligned} \tilde{\mathbf{K}}_{1,k+1} &= \mathbf{I} + \frac{1 - \alpha_{k+1}}{\alpha_{k+1}} \mathbf{K}_{1,k} \mathbf{Q}_{1,k+1} \Xi_{1,k+1} \mathbf{Q}_{1,k+1}^T \mathbf{K}_{1,k}^T \\ &= \mathbf{I} + \frac{1 - \alpha_{k+1}}{\alpha_{k+1}} (\mathbf{K}_{1,k} \mathbf{Q}_{1,k+1}) \Xi_{1,k+1} (\mathbf{K}_{1,k} \mathbf{Q}_{1,k+1})^T \\ &= \mathbf{I} + \tilde{\mathbf{Q}}_{1,k+1} \tilde{\Xi}_{1,k+1} \tilde{\mathbf{Q}}_{1,k+1}^T \end{aligned} \quad (\text{S6})$$

where $\tilde{\mathbf{Q}}_{1,k+1} = \mathbf{K}_{1,k} \mathbf{Q}_{1,k+1} \in \mathbb{R}^{m_1 \times 2}$, $\tilde{\Xi}_{1,k+1} = \frac{1 - \alpha_{k+1}}{\alpha_{k+1}} \Xi_{1,k+1}$ and $\text{rank}(\tilde{\Xi}_{1,k+1}) \leq 2$. As $\mathbf{B}_{1,k+1}$ is positive definite, $\mathbf{I} + \tilde{\mathbf{Q}}_{1,k+1} \tilde{\Xi}_{1,k+1} \tilde{\mathbf{Q}}_{1,k+1}^T$ is also positive definite. For brevity, let $\tilde{\mathbf{Q}}_{1,k+1} = [\tilde{\mathbf{q}}_{1,k+1} \quad \tilde{\mathbf{q}}_{2,k+1}]$, $\tilde{\Xi}_{1,k+1} = \begin{bmatrix} \tilde{\beta}_{1,k+1} & \\ & \tilde{\beta}_{2,k+1} \end{bmatrix}$. We select the calculation way of $\tilde{\mathbf{K}}_{1,k+1}^{-\frac{1}{2}}$ based on the rank of $\tilde{\Xi}_{1,k+1}$.

1) $\text{rank}(\tilde{\Xi}_{1,k+1}) = 1$.

Let $\beta_{1,k+1} \neq 0$ and $\beta_{2,k+1} = 0$, here $\tilde{\boldsymbol{\Xi}}_{1,k+1} = \tilde{\beta}_{1,k+1}$, $\tilde{\mathbf{Q}}_{1,k+1} = \tilde{\mathbf{q}}_{1,k+1}$. Then, according to [1]

$$\begin{aligned}\tilde{\mathbf{K}}_{1,k+1}^{-\frac{1}{2}} &= \left(\mathbf{I} + \tilde{\mathbf{Q}}_{1,k+1} \tilde{\boldsymbol{\Xi}}_{1,k+1} \tilde{\mathbf{Q}}_{1,k+1}^T \right)^{-\frac{1}{2}} \\ &= \left(\mathbf{I} + \tilde{\beta}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1}^T \right)^{-\frac{1}{2}} \\ &= \mathbf{I} + \frac{\tilde{\mathbf{q}}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1}^T}{\tilde{\mathbf{q}}_{1,k+1}^T \tilde{\mathbf{q}}_{1,k+1}} \left(\frac{1}{\sqrt{1 + \tilde{\beta}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1}^T \tilde{\mathbf{q}}_{1,k+1}}} - 1 \right) \\ &= \mathbf{I} + \gamma_{1,k+1} \tilde{\mathbf{q}}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1}^T\end{aligned}$$

where $\gamma_{1,k+1} = \frac{1}{\tilde{\mathbf{q}}_{1,k+1}^T \tilde{\mathbf{q}}_{1,k+1}} \left(\frac{1}{\sqrt{1 + \tilde{\beta}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1}^T \tilde{\mathbf{q}}_{1,k+1}}} - 1 \right)$. Thus, the recursion of \mathbf{K}_1 is

$$\mathbf{K}_{1,k+1} = \frac{1}{\sqrt{\alpha_{k+1}}} (\mathbf{I} + \gamma_{1,k+1} \tilde{\mathbf{q}}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1}^T) \mathbf{K}_{1,k} \quad (S7)$$

Here, $\mathbf{G}_{1,k+1} = \frac{1}{\sqrt{\alpha_{k+1}}} (\mathbf{I} + \gamma_{1,k+1} \tilde{\mathbf{q}}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1}^T)$.

2) $\text{rank}(\tilde{\boldsymbol{\Xi}}_{1,k+1}) = 2$.

The formula (S6) can be further reformulated into

$$\begin{aligned}\mathbf{I} + \tilde{\mathbf{Q}}_{1,k+1} \tilde{\boldsymbol{\Xi}}_{1,k+1} \tilde{\mathbf{Q}}_{1,k+1}^T &= \mathbf{I} + \tilde{\beta}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1}^T + \tilde{\beta}_{2,k+1} \tilde{\mathbf{q}}_{2,k+1} \tilde{\mathbf{q}}_{2,k+1}^T \\ &= \check{\mathbf{Q}}_{1,k+1} \check{\boldsymbol{\Lambda}}_{1,k+1} \check{\mathbf{Q}}_{1,k+1}^T\end{aligned}$$

where $\check{\mathbf{Q}}_{1,k+1}$ is the eigen matrix with $\check{\mathbf{Q}}_{1,k+1}^T \check{\mathbf{Q}}_{1,k+1} = \mathbf{I}$, $\check{\boldsymbol{\Lambda}}_{1,k+1}$ contains the eigenvalues. (S6) is realized by two successive rank-1 modification with first-order perturbation (FOP) [2]. Thus, (S5) is further calculated:

$$\mathbf{K}_{1,k+1} = \frac{1}{\sqrt{\alpha_{k+1}}} \check{\boldsymbol{\Lambda}}_{1,k+1}^{-\frac{1}{2}} \check{\mathbf{Q}}_{1,k+1}^T \mathbf{K}_{1,k} \quad (S8)$$

Here, $\mathbf{G}_{1,k+1} = \frac{1}{\sqrt{\alpha_{k+1}}} \check{\boldsymbol{\Lambda}}_{1,k+1}^{-\frac{1}{2}} \check{\mathbf{Q}}_{1,k+1}^T$.

In conclusion, $\mathbf{K}_{1,k+1}$ can be calculated adaptively in the form of $\mathbf{K}_{1,k+1} = \mathbf{G}_{1,k+1} \mathbf{K}_{1,k}$, where

$$\mathbf{G}_{1,k+1} = \begin{cases} \frac{1}{\sqrt{\alpha_{k+1}}} (\mathbf{I} + \gamma_{1,k+1} \tilde{\mathbf{q}}_{1,k+1} \tilde{\mathbf{q}}_{1,k+1}^T), & \text{rank}(\Delta \mathbf{B}_{1,k+1}) = 1 \\ \frac{1}{\sqrt{\alpha_{k+1}}} \check{\boldsymbol{\Lambda}}_{1,k+1}^{-\frac{1}{2}} \check{\mathbf{Q}}_{1,k+1}^T, & \text{rank}(\Delta \mathbf{B}_{1,k+1}) = 2 \end{cases}$$

As $\text{rank}(\Delta \mathbf{B}_{2,k+1}) \leq 2$ and $\mathbf{K}_{2,k+1} = \mathbf{G}_{2,k+1} \mathbf{K}_{2,k}$, $\mathbf{G}_{2,k+1}$ can be calculated recursively in the similar way. Thus, $\mathbf{K}_{k+1} = \begin{bmatrix} \mathbf{G}_{1,k+1} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{2,k+1} \end{bmatrix} \mathbf{K}_k$, where $\mathbf{G}_{1,k+1}$ and $\mathbf{G}_{2,k+1}$ are determined adaptively.

4. SOLUTION OF RPCA–EWC

For convenience, the subscripts of parameters are ignored. Specifically, let $\mathbf{P}_0^* = \mathbf{P}_{\mathcal{M}_{K-1}}$, $\mathbf{X}_2 = \tilde{\mathbf{X}}_{2,K}$, $\Omega = \zeta_K \Omega_{\mathcal{M}_{K-1}}$. The formula (12) in the paper can be described as follows:

$$\begin{aligned}\mathcal{J}(\mathbf{P}) &= \mathcal{J}_K(\mathbf{P}) + \zeta_K \mathcal{J}_{loss}(\mathbf{P}, \mathbf{P}_0^*) \\ &= \|\mathbf{X}_2 - \mathbf{X}_2 \mathbf{P} \mathbf{P}^T\|_F^2 + \|\mathbf{P} - \mathbf{P}_0^*\|_\Omega^2\end{aligned}\tag{S9}$$

The formula (S9) can be reformulated as

$$\mathcal{J}(\mathbf{P}) = \text{tr}(\mathbf{P}^T \Omega \mathbf{P}) - \text{tr}(\mathbf{P}^T \mathbf{X}_2^T \mathbf{X}_2 \mathbf{P}) - 2\text{tr}(\mathbf{P}^T \Omega \mathbf{P}_0^*) + \underbrace{\{\text{tr}(\mathbf{X}_2^T \mathbf{X}_2) + \text{tr}(\mathbf{P}_0^{*T} \Omega \mathbf{P}_0^*)\}}_{\text{constant}}\tag{S10}$$

Let $G(\mathbf{P}) = \text{tr}(\mathbf{P}^T \Omega \mathbf{P}) - 2\text{tr}(\mathbf{P}^T \Omega \mathbf{P}_0^*)$, $H(\mathbf{P}) = \text{tr}(\mathbf{P}^T \mathbf{X}_2^T \mathbf{X}_2 \mathbf{P})$. Thus, $\mathcal{J}(\mathbf{P}) = G(\mathbf{P}) - H(\mathbf{P}) + \text{constant}$. The minimization of (S10) is equivalent to

$$\begin{aligned}\min_{\mathbf{P}} \quad & G(\mathbf{P}) - H(\mathbf{P}) \\ \text{s.t.} \quad & \mathbf{P}^T \mathbf{P} = \mathbf{I} \in \mathbb{R}^{l \times l}\end{aligned}\tag{S11}$$

Because $G(\mathbf{P})$ and $H(\mathbf{P})$ are convex, the objective function (S11) is actually DC programming problem [3], [4]. DC programming includes linearizing and solving the convex functions as follows [5].

Assume that \mathbf{P}_i is the solution at i th iteration, we approximate the second part $H(\mathbf{P})$ by linearizing

$$H_l(\mathbf{P}) = H(\mathbf{P}_i) + \langle \mathbf{P} - \mathbf{P}_i, \mathbf{U}_i \rangle$$

As the subgradient $\mathbf{U} \in \partial H(\mathbf{P}) = 2\mathbf{X}_2^T \mathbf{X}_2 \mathbf{P}$, let $\mathbf{U}_i = 2\mathbf{X}_2^T \mathbf{X}_2 \mathbf{P}_i$. Then, (S11) is approximated by

$$\mathbf{P}_{i+1} \doteq \arg \min_{\substack{\mathbf{P}^T \mathbf{P} = \mathbf{I}}} G(\mathbf{P}) - \langle \mathbf{P}, \mathbf{U}_i \rangle$$

Because Ω is semidefinite, let $\Omega = \mathbf{L}^T \mathbf{L}$ and \mathbf{L} is the triangle matrix [5]. Thus, we can get

$$\begin{aligned}G(\mathbf{P}) - \langle \mathbf{P}, \mathbf{U}_i \rangle &= \text{tr}(\mathbf{P}^T \Omega \mathbf{P}) - 2\text{tr}(\mathbf{P}^T \Omega \mathbf{P}_0^*) - 2\text{tr}(\mathbf{P}^T \mathbf{X}_2^T \mathbf{X}_2 \mathbf{P}_i) \\ &= \langle \mathbf{L} \mathbf{P}, \mathbf{L} \mathbf{P} \rangle - 2\langle \mathbf{L} \mathbf{P}, \mathbf{L} \mathbf{P}_0^* + (\mathbf{L}^T)^{-1} \mathbf{X}_2^T \mathbf{X}_2 \mathbf{P}_i \rangle \\ &= \|\mathbf{Z}_i - \mathbf{L} \mathbf{P}\|_F^2 - \|\mathbf{Z}_i\|_F^2\end{aligned}\tag{S12}$$

where $\mathbf{Z}_i = \mathbf{L} \mathbf{P}_0^* + (\mathbf{L}^T)^{-1} \mathbf{X}_2^T \mathbf{X}_2 \mathbf{P}_i$ is constant at $(i+1)$ th iteration [5]. Then, (S12) is equivalent to [5], [6]

$$\mathbf{P}_{i+1} = \arg \min_{\substack{\mathbf{P}^T \mathbf{P} = \mathbf{I}}} \|\mathbf{P} - \mathbf{L}^T \mathbf{Z}_i\|_F^2\tag{S13}$$

Let $\mathbf{Y}_i = \mathbf{L}^T \mathbf{Z}_i = \Omega \mathbf{P}_0^* + \mathbf{X}_2^T \mathbf{X}_2 \mathbf{P}_i$. According to the lemma in [6], we can obtain that $\mathbf{P}_{i+1} = \mathbf{W}_i \mathbf{I}_{m_2, l} \mathbf{V}_i^T$, where $\mathbf{W}_i \in \mathbb{R}^{m_2 \times m_2}$ and $\mathbf{V}_i \in \mathbb{R}^{l \times l}$ are left and right singular values of the singular vector decomposition of \mathbf{Y}_i [5]. m_2 is the dimension of \mathbf{X}_2 and l is the number of principal components. The procedure is summarized in Algorithm 2 in the paper.

5. COEFFICIENTS OF THREE MODES FOR NUMERICAL CASE

In this paper, the coefficients are set differently for different modes. The detailed setting is described in (S14)-(S16).

Mode \mathcal{M}_1 :

$$\left\{ \begin{array}{l} a_1 = 1.5, b_1 = 3; \\ a_2 = 1, b_2 = 2.5; \\ a_3 = 0.3, b_3 = -0.6, c_3 = 2; \\ a_4 = -0.3, b_4 = 0.6, c_4 = 4; \\ a_5 = 2.2; \\ a_6 = 0.4, b_6 = -0.1, c_6 = 0.2, d_6 = 0.8; \\ a_7 = 0.6, b_7 = 0.1, c_7 = 0.6, d_7 = 0.4; \end{array} \right. \quad (\text{S14})$$

Mode \mathcal{M}_2 :

$$\left\{ \begin{array}{l} a_1 = 1.5, b_1 = 3.5; \\ a_2 = 2, b_2 = 2; \\ a_3 = 0.4, b_3 = -0.8, c_3 = 3; \\ a_4 = -0.2, b_4 = 0.4, c_4 = 3; \\ a_5 = 1.9; \\ a_6 = 0.8, b_6 = -0.1, c_6 = 0.4, d_6 = 0.8; \\ a_7 = 0.8, b_7 = 0.3, c_7 = 0.4, d_7 = 0.4; \end{array} \right. \quad (\text{S15})$$

Mode \mathcal{M}_3 :

$$\left\{ \begin{array}{l} a_1 = 1.2, b_1 = 3; \\ a_2 = 2, b_2 = 2.5; \\ a_3 = 0.4, b_3 = -0.8, c_3 = 2; \\ a_4 = -0.3, b_4 = 0.6, c_4 = 4; \\ a_5 = 1.6; \\ a_6 = 0.4, b_6 = -0.1, c_6 = 0.3, d_6 = 0.6; \\ a_7 = 0.5, b_7 = 0.2, c_7 = 0.5, d_7 = 0.8; \end{array} \right. \quad (\text{S16})$$

where t follows uniform distribution with $t \sim U([0, 1])$.

TABLE S1
VARIABLE DESCRIPTION OF PULVERIZING SYSTEM

Variables	Test results	Block number
Rotary separator speed	Nonstationary	Block 1
Coal mill seal air pressure	Nonstationary	Block 1
Differential pressure between seal air and grinding bowl	Nonstationary	Block 1
Upper and lower differential pressure of coal mill bowl	Nonstationary	Block 1
Instantaneous coal feeding capacity	Nonstationary	Block 1
Motor speed of coal feeder	Nonstationary	Block 1
Generator active power	Nonstationary	Block 1
Air powder mixture pressure	Nonstationary	Block 1
Cold primary air electric regulating baffle position feedback	Nonstationary	Block 1
Primary air flow	Nonstationary	Block 1
Primary air pressure	Nonstationary	Block 1
Primary air temperature	Stationary	Block 2
Feed coal current	Stationary	Block 2
Outlet temperature	Stationary	Block 2
Planetary gear box input bearing temperature	Nonstationary	Block 3
Inlet air temperature of forced draft fan	Nonstationary	Block 3
Bearing temperature of rotary separator	Nonstationary	Block 3

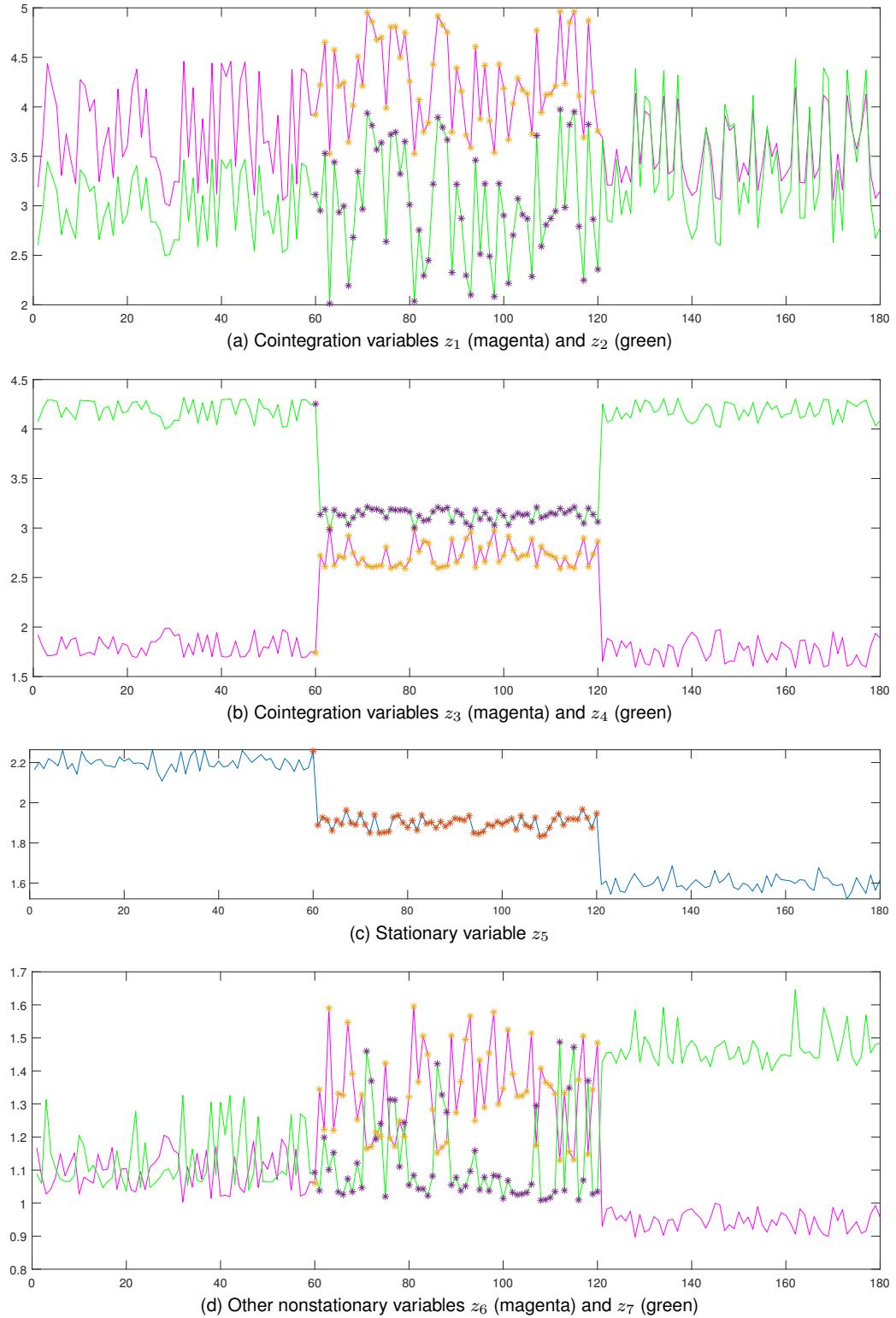


Fig. S1. Variables of the numerical case: the first 60 samples are from the mode \mathcal{M}_1 , the following 60 samples are from the mode \mathcal{M}_2 and the remaining data are from the mode \mathcal{M}_3 .

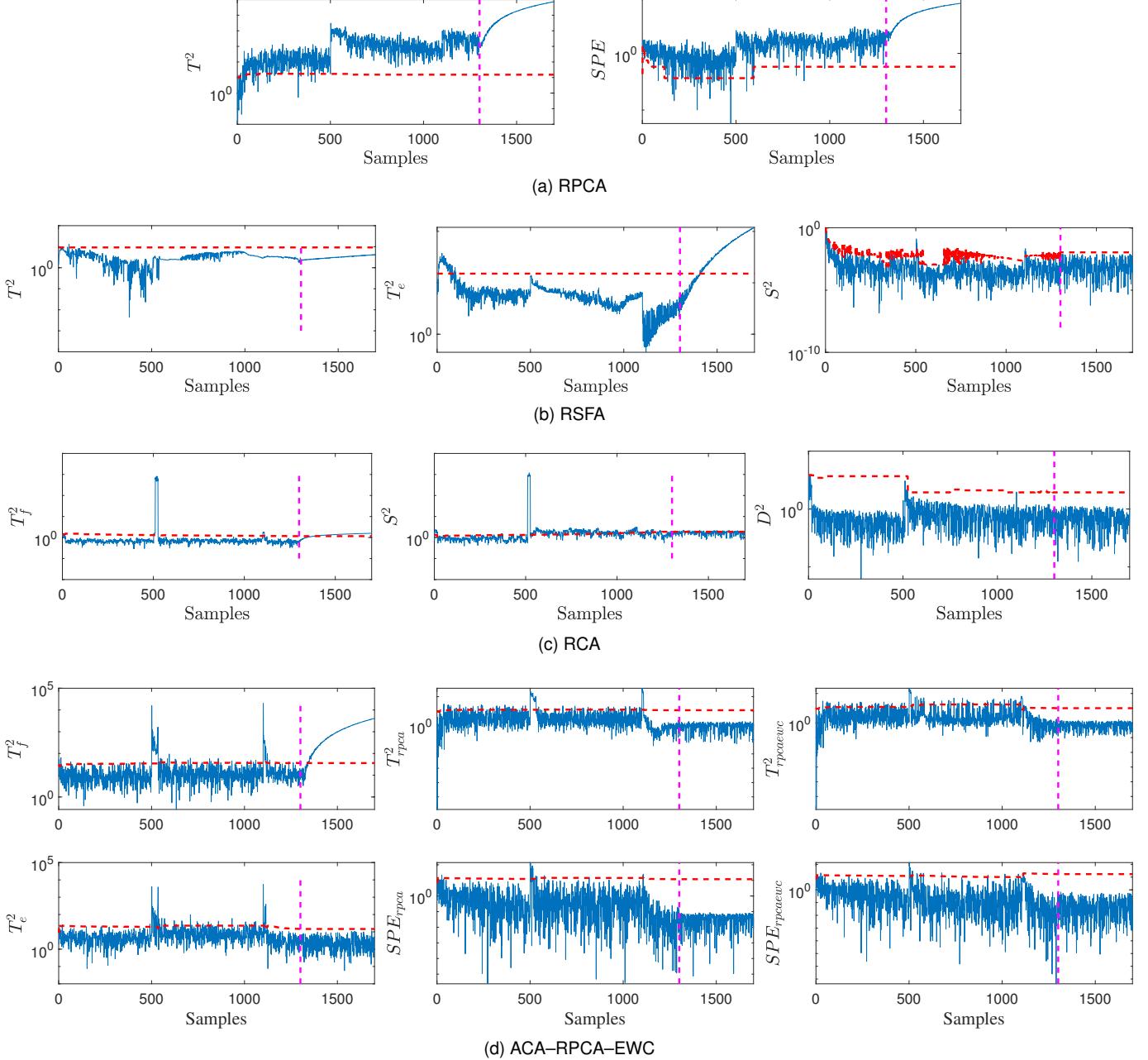


Fig. S2. Monitoring charts of Case 2: (a) RPCA fails to distinguish normal changes from real faults, and the FARs approach 100%; (b) RSFA can detect the fault if the amplitude is large enough, but the FDR is 72.75% and not satisfactory; (c) For RCA, the FDR of T^2 is 79.50%, but the FAR of S^2 is 43.46%. Normal changes are mistaken for the fault; (d) For the proposed ACA–RPCA–EWC method, z_5 is contained in T^2_f and the fault amplitude is small at the initial stage. Thus, the FDR of T^2_f is 89.25% and the other statistics are lower than thresholds. Besides, there are two time periods where the T^2_f and T^2_e are over the thresholds, it indicates that a new mode is encountered.

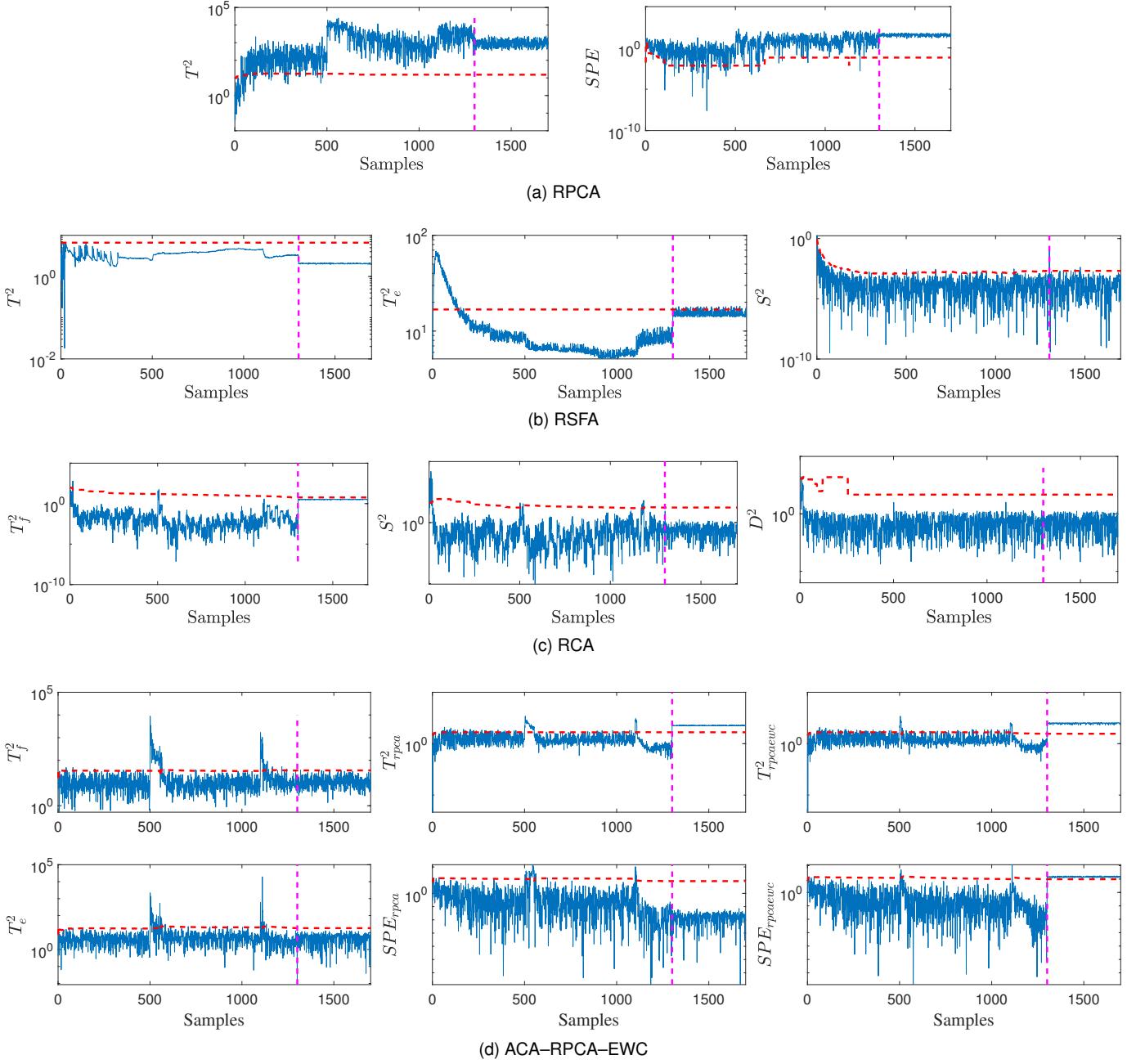


Fig. S3. Monitoring charts of Case 3: (a) RPCA is unable to track the normal changes and monitor the system accurately; (b) RSFA cannot track the rapid changes in the entire dataset and the fault is misjudged as normal variations. The FDR is less than 20% and the FAR is 10.69%; (c) RCA is not able to detect this abnormality and the FDR is less than 20%; (d) z_6 is faulty and it is irrelevant to cointegration relationship. Thus, the FDRs of T_f^2 and T_e^2 approach 0 for the proposed ACA-RPCA-EWC method. The FDRs of $T_{rpcaewc}^2$ and $SPE_{rpcaewc}$ are 100%, and it indicates that the fault is detected accurately. However, the FDR of SPE_{rpca} is 0, which means that RPCA-EWC performs better than RPCA and the learned knowledge retained by EWC is beneficial for building an accurate RPCA model.

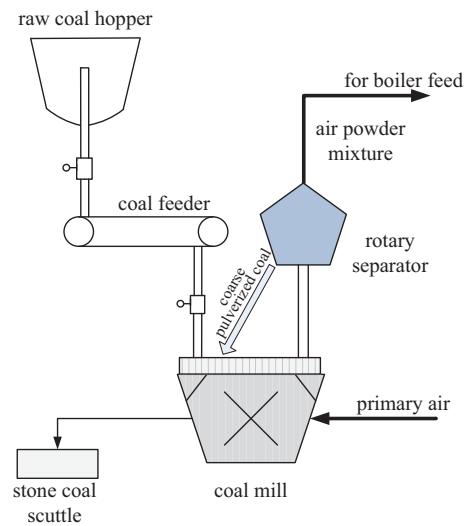


Fig. S4. Schematic diagram of coal pulverizing system

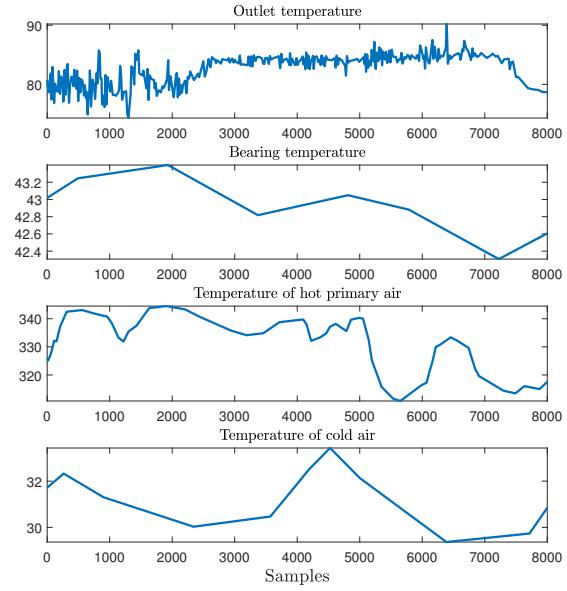
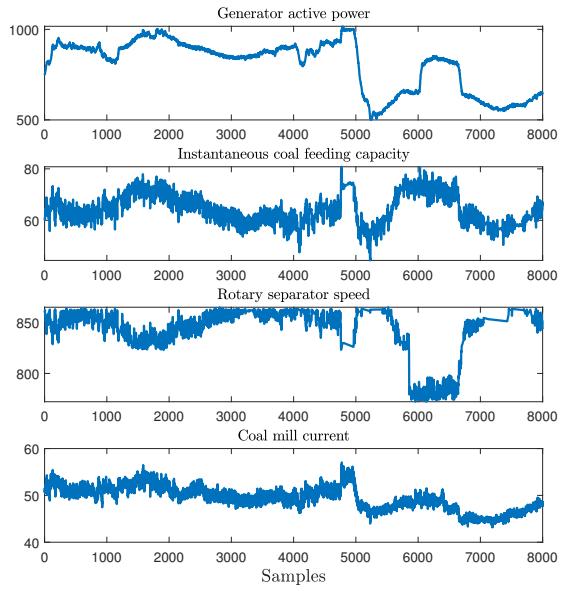


Fig. S5. Practical data from coal pulverizing system

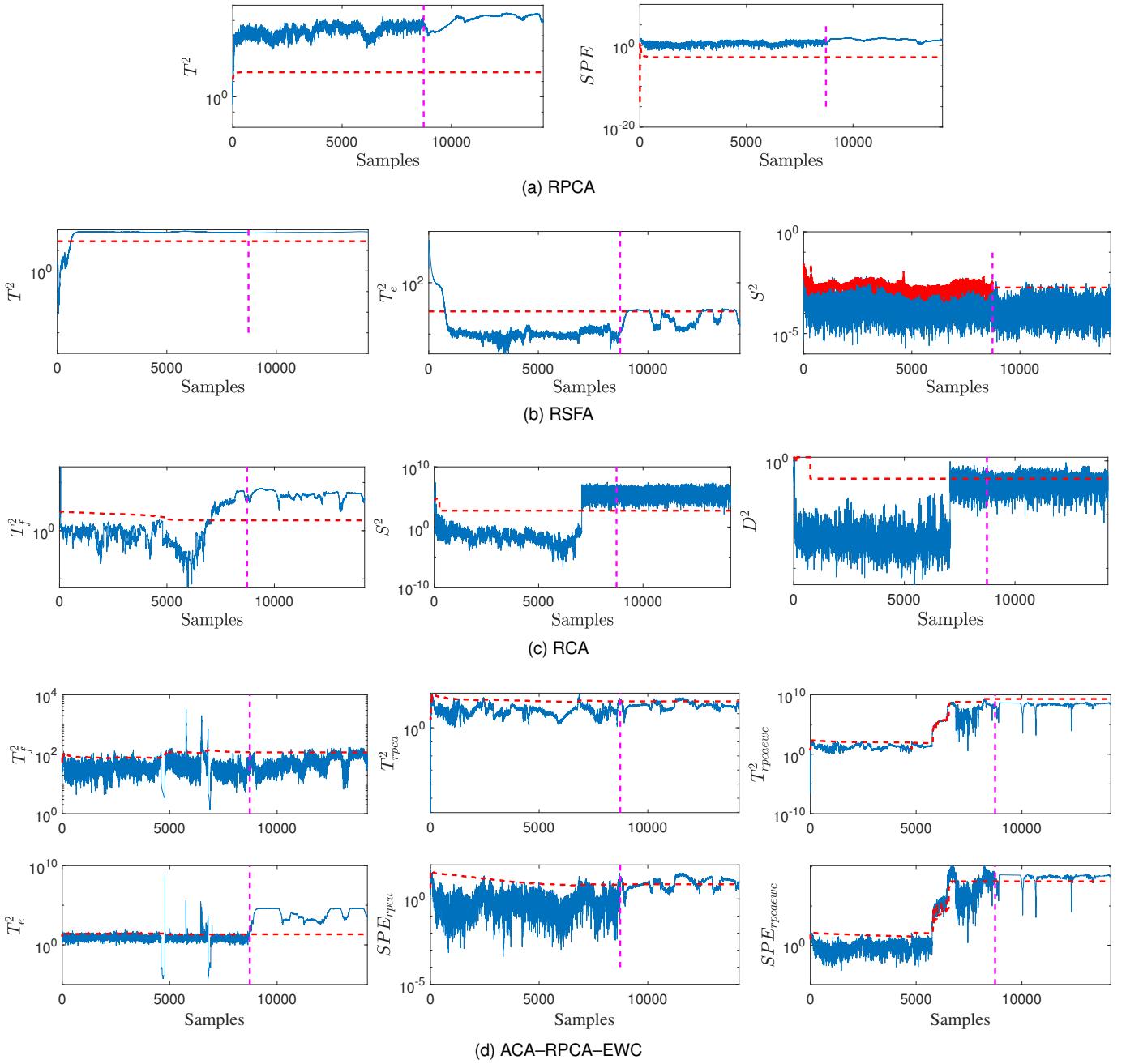


Fig. S6. Monitoring charts of Case 5: (a) RPCA cannot detect the fault and the FAR of SPE is 100%; (b) RSFA fails to distinguish normal changes from the real fault, and the FAR of T^2 is 92.84%; (c) Although the FDR of RCA is 100%, the FARs of T^2 and S^2 approximate 20%. According to the data record, the change of coal type is mistaken for the fault; (d) For the proposed ACA-RPCA-EWC method, T^2_e can detect the fault accurately and the FDR is 99.98%. T^2_e and T^2_f change sharply twice. According to the coal record and original data analysis, the first sudden change of two statistics originates from the switch of coal type, while the second abrupt change is attributed to the critical parameters adjusted artificially. $SPE_{rpcaewc}$ statistic enables to detect the fault precisely and the FDR is 93.86%. The short-term dynamic of two types of coal has a certain degree of similarity. The significant information of previous coal is preserved and beneficial for monitoring other coal. When the mode starts to switch from one to another, the FARs of SPE_{rpca} and $SPE_{rpcaewc}$ are relatively high because RPCA and RPCA-EWC are difficult to track the rapid normal changes in the initial stage. The system is judged as normal because all statistics return to normal quickly. Regardless of the false alarms caused by this situation, the FARs of SPE_{rpca} and $SPE_{rpcaewc}$ are 5.20% and 7.62%, respectively.

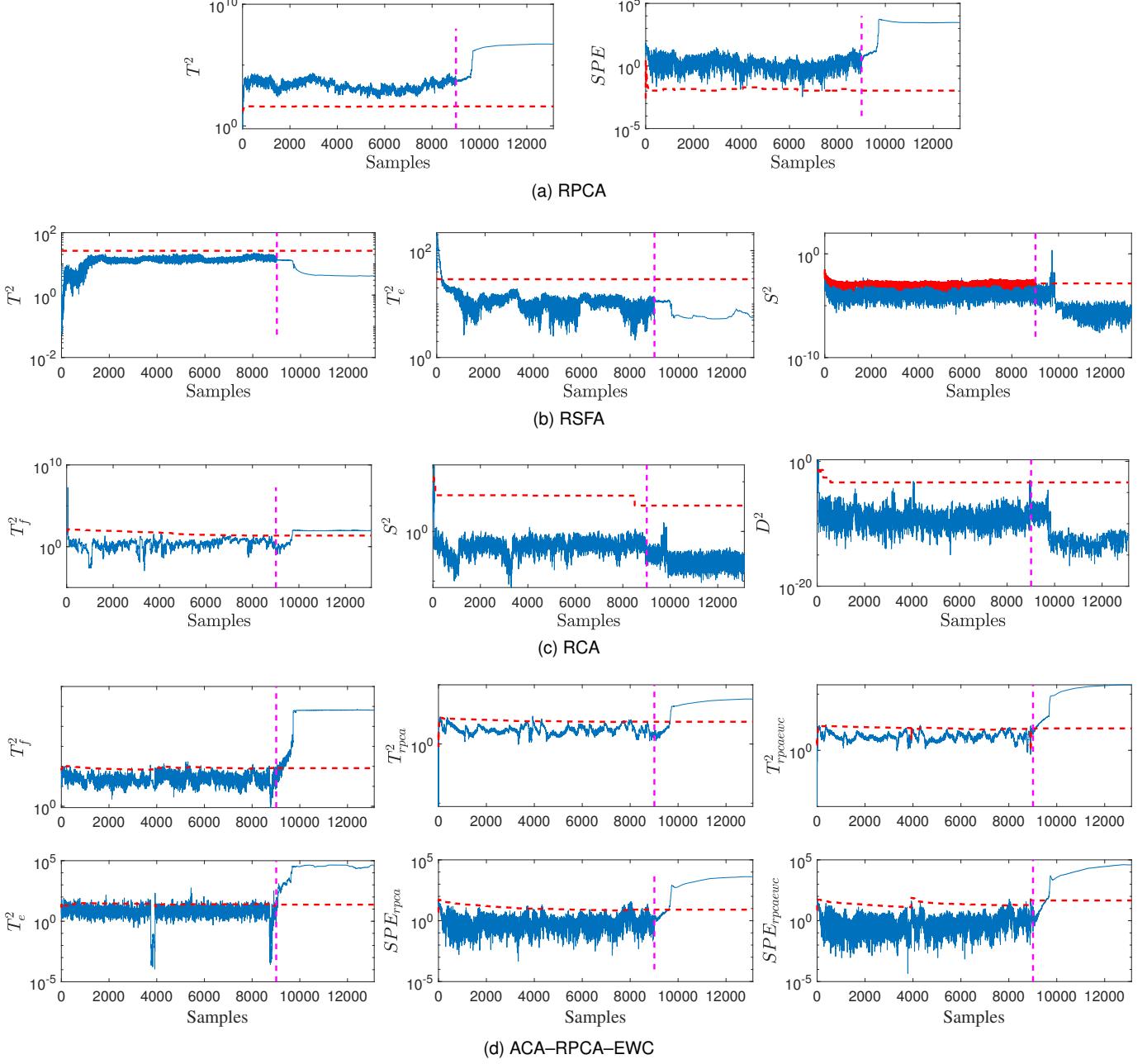


Fig. S7. Monitoring charts of Case 6: (a) The FAR of RPCA approaches 100% and RPCA is not appropriate for multimode nonstationary processes; (b) SFA cannot separate the normal variations and the real fault, and the FDR is close to 0; (c) There exists detection delay for RCA, and the FDR of T^2 is 83%; (d) The proposed ACA–RPCA–EWC algorithm can detect the fault accurately. The FDR of $T_{rpcaeuc}^2$ is 97.20%, which indicates that the significant information of the previous coal is preserved by EWC and beneficial for delivering excellent monitoring performance for other similar modes. However, the FDRs of T_{rpca}^2 and SPE_{rpca} are less than 86%.

REFERENCES

- [1] T. Tanaka, “Fast generalized eigenvector tracking based on the power method,” *IEEE Signal Proc. Let.*, vol. 16, no. 11, pp. 969–972, 2009.
- [2] L. M. Elshenawy, S. Yin, A. S. Naik, and S. X. Ding, “Efficient recursive principal component analysis algorithms for process monitoring,” *Ind. Eng. Chem. Res.*, vol. 49, no. 1, pp. 252–259, 2010.
- [3] J. C. O. Souza, P. R. Oliveira, and A. Soubeyran, “Global convergence of a proximal linearized algorithm for difference of convex functions,” *Optim. Lett.*, vol. 10, no. 7, pp. 1–11, 2015.

-
- [4] T. V. Voorhis and F. A. Al-Khayyal, "Difference of convex solution of quadratically constrained optimization problems," *Eur. J. Oper. Res.*, vol. 148, no. 2, pp. 349–362, 2003.
 - [5] J. Zhang, D. Zhou, and M. Chen, "Monitoring multimode processes: a modified PCA algorithm with continual learning ability," *J. Process Contr.*, vol. 103, pp. 76–86, 2021.
 - [6] J. Huang, F. Nie, H. Huang, and C. Ding, "Robust manifold nonnegative matrix factorization," *ACM T Knowl. Discov. D.*, vol. 8, no. 3, pp. 1–21, 2014.