Estimation of Performance Benchmark
Based on the Tensor Space

Zhi Zhang, Li-Sheng Hu

Department of Automation, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, 200240, China. (Corresponding author e-mail: lshu@sjtu.edu.cn)

Abstract: During the last two decades, performance assessment for control systems has been receiving wide attention. The key step is to describe and estimate the benchmark. However, the issue for more complex control systems still remains open. This paper is concerned primarily with the estimation of the performance benchmark in the tensor space, and explores to extend traditional methods to the complex system. An estimation procedure based on higher-order singular value decomposition is proposed. Numerical examples show the effectiveness of the proposed method.

1. INTRODUCTION

By virtue of valuable contributions from Harris [1989], research on control loops performance assessment has been increasing in the last two decades. There are many researches on performance assessment for linear control loops, see [Huang and Shah, 1999, Harris et al., 1999, Jelali, 2006, Qin, 1998], and the references therein. The main motivation of this issue is to provide an online automated procedure for determining whether specified performance targets and response characteristics are being met by the controlled process variables or not [Jelali, 2006]. The key step for performance assessment procedure is to establish the appropriate performance benchmark in order to obtain reasonable assessment results. The popular performance benchmark are the minimum variance performance and the prediction error variance.

Due to inevitable effects of process delay and disturbance, the closed-loop measured output is considered as the sum of the true output and the additive disturbance, where the disturbance keeps d-step ahead than the control action and is driven by the Gaussian white noise. In this framework, Huang and Shah [1999] designed the interactor-filter to estimate the performance benchmark, which extend the works of Harris [1989] to the multivariable case. The length of the interactor-filter, or delay, can be obtained by singular value decomposition (SVD), which can also be considered as the maximum number of multiple infinite zeros or the maximum iterative number of 1-step prediction process. Then the filter parameters can be obtained by the iterative QR factorization algorithm for these 1-step prediction processes. Hence, the factorization algorithm requires only the first d Markov parameter matrices. Alternatively, the estimation of benchmark can be achieved by orthogonal decomposition in the vector space. The measurement, i.e., the square of 2-norm, of the orthogonal complement is the prediction error variance, which can also be considered as the performance benchmark.

For reducing the complexity of the a priori knowledge requirement, subspace identification is used for performance assessment recently [Huang and R. Kadali, 2008, Ljung and McKelvey, 1996, Overschee and Moor, 1996]. The greatest advantage of this method is to avoid estimating the process delay. It only needs the continuous measurements of the input and the output of the past and the future data block in the time series form. In terms of causality, the future data block space is projected onto the past data block space. The orthogonal complement can be considered as the 1-step prediction error of the true closed-loop system in the data block form. Unlike estimating interactor matrices, this method can directly obtain the filter parameters by only one QR factorization of the data block. This method can be considered as orthogonal decomposition in the matrix space.

Interestingly, the subspace method is more advanced than the interactor-filter method. Because the subspace method can transform complex dynamics into the simpler ones in higher-order dimensions, and then the advanced linear operator such as matrix operator can be used to estimate the performance benchmark. The transformed dynamics can be considered as the complex relationship between the delay and the size of the lower triangular block-Toeplitz matrix. This transformation can be seemed as a class of nonlinear operator. In fact, some complex systems can be considered as the result of a complex transformation action which is a collection of the simpler linear transformation actions twisting around each other. If one wants to describe the performance benchmark for the complex control system as good as possible, on the one hand, the whole complex system can be projected onto the different subspaces, and then the synthetical information can be used; on the other hand, the whole complex system can be analytically described in the upper space of higher-order dimensions.

* This work was partially supported by Natural Science Foundation of China under Grants 60474020 and 60774006.
In terms of higher-order dimensions, the description based on tensor space can offer a powerful mathematical approach for obtaining performance benchmark. In the tensor space, the complex system can be transformed to linear systems by multiliner mappings, which can illustrate how the performance measurement is influenced by multiple underlying causal factors. Moreover, the tenor operation can give an uniform description of the multifactor structure. In fact, it is the natural development for estimation of performance benchmark. Grimble [2005], Harris and Yu [2007] discussed the problem of estimating performance benchmark for nonlinear control systems in the polynomial ring framework. In high-order form, it can be considered as a class of Volterra kernel function operation. Choudhury et al. [2008] used the properties of higher-order statistics (HOS) to detect and quantify the effects of the nonlinear action such as valve stiction. Yu et al. [2010] established multi-way PLS methods for monitoring batch processes, MacGregor [1995] have also reported multi-way PCA and multi-way PLS methods for monitoring batch processes, which can be virtually considered as the tensor operation.

Based on this idea, this paper will explore a feasible description on performance benchmark in the tensor space. The orthogonal complement of the closed-loop system can be obtained based on the higher-order singular value decomposition (HOSVD). The measurement of this orthogonal complement is the performance benchmark. A major contribution of this paper is the application of the idea of tensor operation and its usage to establish the performance benchmark and explore to extend traditional methods to more complex control systems.

The remainder of this paper is organized as follows. Symbol definitions are presented in Section 2. The preliminaries of tensor and HOSVD are presented in Section 3. Problem formulation on estimation of performance benchmark for the complex system is presented in Section 4. Section 5 presents the assessment procedure based on the tensor space. In Section 6, numerical examples show the effectiveness of the proposed method. Finally, the conclusion remarks are given in Section 7.

2. NOTATIONS

Throughout this paper, we will denote scalars by lower-case letters (a, b, . . .), vectors by italic uppercase letters (A, B, . . .), matrices by bold uppercase letters (A, B, . . .), and tensors by calligraphic uppercase letters (A, B, . . .). \( \mathbb{R}^{i_1 \times i_2 \times \cdots \times i_N} \) denotes the vector space of real valued \((i_1 \times i_2 \times \cdots \times i_N)\)-tensors. \( \mathbf{A}_{(n)} \) denotes the \( n \)-mode vector of tensor \( \mathbf{A} \in \mathbb{R}^{i_1 \times i_2 \times \cdots \times i_N} \), where \( n = 1, 2, \ldots, N \). \( \mathbf{A} \times_n \mathbf{U} \) denotes \( n \)-mode matrix-tensor product. rank(\( \mathbf{A}_{(n)} \)) denotes \( n \)-mode rank of tensor \( \mathbf{A} \). \( N \) will be reserved to denote the index upper bounds, unless stated otherwise.

3. PRELIMINARIES

A tensor is a high-order generalization of a scalar (zeroth-order tensor), vector (first-order tensor), and matrix (second-order tensor), which is the generalization of a matrix of orders higher than two. Whereas matrices define linear mappings over a vector space, tensors define multilinear mappings over a set of vector spaces.

The unfolded matrix \( \mathbf{A}_{(n)} \) can be obtained by unfolding the Nth-order tensor \( \mathbf{A} \in \mathbb{R}^{i_1 \times i_2 \times \cdots \times i_N} \), as shown in Fig. 1, see [Lathauwer et al., 2000a] for more details.

Fig. 1. Unfolding of the \((i_1 \times i_2 \times i_3)\)-tensor \( \mathbf{A} \) to the \((i_1 \times i_2)i_3\)-matrix \( \mathbf{A}_{(1)} \), the \((i_2 \times i_3)i_1\)-matrix \( \mathbf{A}_{(2)} \) and the \((i_3 \times i_1)i_2\)-matrix \( \mathbf{A}_{(3)} \).

The \( n \)-rank of tensor \( \mathbf{A} \) is the rank of the \( n \)-mode unfolded matrix \( \mathbf{A}_{(n)} \). The \( n \)-mode vectors of \( \mathbf{A} \) are the column vectors of the unfolded matrix \( \mathbf{A}_{(n)} \).

The Frobenius-norm of a tensor \( \mathbf{A} \) is given by

\[ \|\mathbf{A}\|_F := \sqrt{\langle \mathbf{A}, \mathbf{A} \rangle}. \]  (1)

The scalar product \( \langle \mathbf{A}, \mathbf{B} \rangle \) of two tensors \( \mathbf{A}, \mathbf{B} \in \mathbb{R}^{i_1 \times i_2 \times \cdots \times i_N} \) is defined as

\[ \langle \mathbf{A}, \mathbf{B} \rangle := \sum_{i_1} \sum_{i_2} \cdots \sum_{i_N} b_{i_1i_2 \cdots i_N} a_{i_1i_2 \cdots i_N}. \]  (2)

The \( n \)-mode product of a tensor \( \mathbf{A} \in \mathbb{R}^{i_1 \times i_2 \times \cdots \times i_N} \) by a matrix \( \mathbf{U} \in \mathbb{R}^{i_N \times b} \) is denoted by \( \mathbf{A} \times_n \mathbf{U} \in \mathbb{R}^{i_1 \times \cdots \times i_{n-1} \times b \times i_{n+1} \times \cdots \times i_N} \)-tensor, where the entries are given by

\[ (\mathbf{A} \times_n \mathbf{U})_{i_1 \cdots i_{n-1} j_{n+1} \cdots i_N} := \sum_{i_n} a_{i_1 \cdots i_{n-1} i_n j_{n+1} \cdots i_N} u_{j_n i_n}. \]  (3)

In terms of the unfolded tensor, it can be expressed as \( \mathbf{UA}_{(n)} \).

The HOSVD was introduced by Lathauwer et al. [2000a]. A \( n \)-th-order tensor \( \mathbf{A} \) can be decomposed as

\[ \mathbf{A} := \mathbf{S} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \cdots \times_N \mathbf{U}_N, \]  (4)

where \( \mathbf{U}_n = [\mathbf{U}_{n,1}, \mathbf{U}_{n,2}, \ldots, \mathbf{U}_{n,b}] \in \mathbb{R}^{i_n \times b} \) is an orthogonal matrix, \( \mathbf{S} \in \mathbb{R}^{i_1 \times \cdots \times i_{n-1} \times i_{n+1} \times \cdots \times i_N} \) is a tensor. The \( \mathbf{S} \) is also called as the core tensor, as shown in Fig. 2. The orthogonal matrix \( \mathbf{U}_n \) of the unfolded matrix \( \mathbf{A}_{(n)} \) from \( \mathbf{A} \) can be obtained by the standard SVD, where \( \mathbf{A}_{(n)} = \mathbf{U}_n \mathbf{\Sigma}_n \mathbf{V}_n^T \). Then the core tensor can be obtained as

\[ \mathbf{S} = \mathbf{A} \times_1 \mathbf{U}_1^T \times_2 \mathbf{U}_2^T \cdots \times_N \mathbf{U}_N^T. \]  (5)

The core tensor can also be computed in a matrix format,

\[ \mathbf{S}_{(n)} = \mathbf{U}_n^T \mathbf{A}_{(n)} (\mathbf{U}_{n+1} \otimes \mathbf{U}_{n+2} \otimes \cdots \mathbf{U}_N \otimes \mathbf{U}_1 \otimes \mathbf{U}_2 \otimes \cdots \otimes \mathbf{U}_{n-1}), \]  (6)

Note that \( \otimes \) denotes the Kronecker product.
For reducing construction, a tensor $\hat{A}$ can be defined by discarding the smallest $n$-mode singular values, which can be considered as the similarly optimal approximation of $A$, as shown in Fig. 3, see [Lathauwer et al., 2000a,b] for more details. The approximated core tensor $\hat{S}$ can be obtained by extracting the first $s_1, s_2, \ldots, s_N$ components of $S$. The truncated matrices can also be obtained as $\hat{U}_1, \hat{U}_2, \ldots, \hat{U}_N$, where $\hat{U}_n \in \mathbb{R}^{n \times s_n}$, and $\hat{U}_n = [U_{n,1}, U_{n,2}, \ldots, U_{n,s_n}]$, for $n = 1, 2, \ldots, N$. Thus the dimension of $\hat{A}$ is reduced and the approximation $\hat{A}$ is

$$\hat{A} := \hat{S} \times_1 \hat{U}_1 \times_2 \hat{U}_2 \cdots \times_N \hat{U}_N$$

(7)

Fig. 3. Schematic representation of the approximated tensor $\hat{A}$ for a third-order tensor.

For visualization, the $k$th block of the approximated tensor $\hat{A}$ as shown in Fig 3, can be obtained by multiplying the $\hat{S}$ with the orthogonal matrices $\hat{U}_2, \hat{U}_3$, and the $k$th row of matrix $\hat{U}_1$.

4. PROBLEM FORMULATION

Consider a control system described by a multilinear transformation. The performance measurements can be influenced by multiple underlying factors. A multilinear transformation is a nonlinear mapping from a set of $N$ domain vector spaces $\mathbb{R}^{i_n}$, $1 \leq n \leq N$, to a range vector space $\mathbb{R}^{i_m}$, which can be described as

$$A : \{\mathbb{R}^{i_1} \times \mathbb{R}^{i_2} \times \cdots \times \mathbb{R}^{i_N}\} \rightarrow \mathbb{R}^{i_m}.$$  

(8)

The problem considered is formulated as follows.

**Problem 1:** Establish the description of assessment procedure for performance benchmark based on the tensor space.

5. MAIN RESULTS

The following is the estimation procedure of performance benchmark based on HOSVD.

**Step 1.** The $N$th-order tensor $A$ can be decomposed by HOSVD, the unfolded matrix along the $i_n$ axis of interest can be obtained as $U_n$. Thus,

$$A = S \times_1 U_1 \times_2 U_2 \cdots \times_N U_N.$$

Go to Step 5 for the explicit model, or go to Step 2 for the implicit model

**Step 2.** The original tensor $A$ can be approximated by

$$\hat{A} = \hat{S} \times_1 \hat{U}_1 \times_2 \hat{U}_2 \cdots \times_N \hat{U}_N.$$

**Step 3.** The $(k + 1)$th block of $\hat{A}$ for $i_n$, $\hat{A}_{k+1} \in \mathbb{R}^{i_1 \times i_2 \times \cdots \times i_{n-1} \times i_{n+1} \times i_N}$, can be obtained by multiplying $\hat{S}$ with the orthogonal matrices $\hat{U}_1, \ldots, \hat{U}_{n-1}, \hat{U}_{n+1}, \ldots, \hat{U}_N$ and the $(k + 1)$th row of matrix $U_n$, $Y_{k+1} \in \mathbb{R}^{i_n}$.

$$\hat{A}_{k+1} = \hat{S} \times_1 \hat{U}_1 \times_2 \hat{U}_2 \cdots \times_{n-1}Y_{k+1} \times_{n+1} \hat{U}_{n+1} \cdots \times_N \hat{U}_N.$$

**Step 4.** For implicit model, the matrix $Y_1 = [Y_1, Y_2, \ldots, Y_{i_n-1}]^T$ and matrix $Y_2 = [Y_2, Y_3, \ldots, Y_{i_n}]^T$ are selected. The estimation $\hat{Y}_{k+1} = \hat{L}_kY_k$ can be obtained by orthogonal projection of $Y_2$ onto $Y_1$, where parameters matrix $L = Y_2Y_1^T$. Note that $\{\}$ denotes Moore-Penrose pseudo-inverse, and $Y_1^T = Y_1(Y_1Y_1^T)^{-1}$. The prediction of the $(k + 1)$th block of tensor $\hat{A}$ for $i_n$, $\hat{A}_{k+1} \in \mathbb{R}^{i_1 \times i_2 \times \cdots \times i_{n-1} \times i_{n+1} \times i_N}$, can be obtained as

$$\hat{A}_{k+1} = \hat{S} \times_1 \hat{U}_1 \times_2 \hat{U}_2 \cdots \times_{n-1}\hat{Y}_{k+1} \times_{n+1} \hat{U}_{n+1} \cdots \times_N \hat{U}_N.$$  

Hence the measurement of prediction error for the $k$th block tensor $\hat{A}$ as the performance benchmark for the $i_n$, is $||\hat{A}_{k+1} - \hat{A}_{k+1}||^2$.

**Step 5.** For explicit model, the process model and disturbance model as the open-loop information are needed, where the input can be considered as a causal factor for the output. Then the causal dynamics between the future output and the past output can be extract by the $d$ steps iterative QR factorizations along the axis of interest. Hence the explicit performance benchmark can be described by the open-loop information modified by the causal dynamics. The subsequent procedure is the same as [Huang and Shah, 1999, Huang and R. Kadali, 2008], they are refereed as further details.

**Remark 1:** In Step 5, the process delay $D \in \mathbb{R}^{i_4}$, the function $f(D) \in \mathbb{R}^{i_8}$, input $U_l \in \mathbb{R}^{i_8}$ and output $Y_i \in \mathbb{R}^{i_m}$ are considered as the modes. Considering a fourth-order tensor $A$, the HOSVD algorithm is used to decompose this tensor as follows:

$$A = S \times_1 \times_4 U_q \times_1 U_l \times_2 \cdots \times_m U_m,$$

(9)

where a core tensor $S$ and the product of four orthogonal mode matrices can be obtained. $S$ governs the interaction between the different modes. The column vectors of $U_q$ span the $\mathbb{R}^{i_q}$ space, while its rows make the delay, input and output invariant representation for each of the different $f(D)$. The relationship between $D$ and $f(D)$ is
\[ D = [0 \ 0 \ \cdots \ 0]^{\tau-1}, \]
\[ f(D) = \begin{bmatrix} q^0 & q^{-1} & \cdots & q^{-\tau+1} \end{bmatrix}^T, \]
where \( 1 \leq \tau < +\infty \), \( q^{-1} \) is the backward shift operator. \( f(D) \) can be considered as a class of nonlinear functions or filter structures with respect to \( D \). A quasi-input \( U_i \in \mathbb{R}^{iv} \) can be obtained if \( f(D) \) acting on \( U_i \). For visualization, (9) can be transformed to a third-order tensor,

\[ A = S \times_d U_d \times_j U_j \times m U_m. \]  

**Remark 2:** Alternatively, Step 3 and Step 4 can be achieved by subspace identification technology. If another axis \( i_j \) is also considered in Step 3,

\[
\begin{align*}
Y_1 &= \begin{bmatrix} Y_1^T & Y_2^T & \cdots & Y_{\tau}^T \\ Y_1^{T+1} & Y_2^{T+1} & \cdots & Y_{\tau+1}^{T+1} \\ \vdots & \vdots & \ddots & \vdots \\ Y_{2\tau-1}^{T} & Y_{2\tau}^{T} & \cdots & Y_{2\tau-1}\tau \end{bmatrix}, \\
Y_2 &= \begin{bmatrix} Y_1^{T+1} & Y_2^{T+2} & \cdots & Y_{\tau+1}^{T+1} \\ Y_1^{T+2} & Y_2^{T+3} & \cdots & Y_{\tau+1}^{T+2} \\ \vdots & \vdots & \ddots & \vdots \\ Y_{2\tau+n-1}^{T} & Y_{2\tau+n}^{T} & \cdots & Y_{2\tau+n-1}\tau \end{bmatrix}
\end{align*}
\]
can be obtained. In Step 4, \( \hat{Y}_2 = \hat{L}X_1 \) can be obtained by the orthogonal projection of row space of \( Y_1 \) onto the row space of \( Y_1 \). Thus, the 1-step prediction of \( \hat{U}_n = [Y_1^{T+1} Y_2^{T+2} \cdots Y_{\tau+1}^{T+1}]^T \) is obtained and the second row of \( \hat{U}_j = E_2 \) \in \mathbb{R}^{iv} \). The 1-step prediction of tensor \( \hat{A}_j \) for \( i_j \) of interest, \( \hat{A}_j \in \mathbb{R}^{i_1 \times i_2 \times \cdots \times i_j-1 \times 1 \times i_{j+1} \times \cdots \times i_N} \) can be obtained as

\[ \hat{A}_j = \hat{S} \times_1 \hat{U}_1 \cdots \times_n \hat{U}_n \cdots \times_j E_2 \cdots \times_N U_N. \]

It can be considered to transform the axis of interest from \( i_n \) to \( i_j \).

**Remark 3:** In subspace form of performance assessment [Huang and R. Kadali, 2008], the past and future output are defined as block-Hankel matrices \( Y_1, Y_2 \in \mathbb{R}^{i_1 \times n \times \tau} \), whose structures are mentioned in Remark 2. The tensor can be described as

\[ A = S \times_d U_d \times_m U_m \times_j U_j, \]

where \( D' \in \mathbb{R}^{iv}, Y_1, Y_2 \in \mathbb{R}^{i_1} \). Note that \( Y_1 \) denotes the past output, \( Y_2 \) denotes the future output, \( D' \) denotes the lagged time between \( Y_1 \) and \( Y_2 \), \( i_n \) and \( i_w \) are the time series axes. If \( D' \) is longer than the system sampling interval, it can be scaled to the unit-length by lifting or multirate sampling technology [Chen and Francis, 1995, Hu and Huang, 2005, Hu et al., 2004, and \( \text{rank}(S_i), \text{rank}(S_{i_1}), \text{rank}(S_{i_2}) \) and \( \text{rank}(S_{i_j}) \) are all full rank, (11) is just the subspace identification technology for the performance assessment. Thus, the prediction error can be obtained by the orthogonal projection of the row space of \( Y_2 \) onto the row space of \( Y_1 \).

**Remark 4:** As shown in Fig. 4, in a sense, time series analysis can be considered to deal with the points series along the time axis \( i_t \); subspace method can be considered to deal with the block matrices series along the block axis \( i_m \). Thus tensor method can also be considered to deal with the tensor series along the axis of interest \( i_w \).
The unfolded matrices of the core tensor of the HOSVD can also be obtained,

\[
\mathbf{S}(\nu) = \begin{bmatrix}
-3.6803 & 0.0514 & 0.0090 & 0.0642 \\
-0.0087 & -0.0601 & -2.7306 & -0.0689 \\
-0.0014 & 0.0480 & 0.0021 & -0.0405 \\
\end{bmatrix},
\]

\[
\mathbf{S}(\nu) = \begin{bmatrix}
-3.6803 & -0.0087 & -0.0014 & 0.0514 & -0.0601 & -0.0480 \\
0.0090 & -2.7306 & 0.0221 & 0.0642 & -0.0689 & -0.0405 \\
\end{bmatrix},
\]

\[
\mathbf{S}(m) = \begin{bmatrix}
-3.6803 & -0.0087 & -0.0014 & 0.0514 & -0.0601 & 0.0642 & -0.0480 & -0.0405 \\
0.0090 & -2.7306 & 0.0221 & 0.0642 & -0.0689 & -0.0405 \\
\end{bmatrix},
\]

We can see that rank(\(\mathbf{S}(\nu)\)) = 2 and rank(\(\mathbf{S}(m)\)) = 2 are full rank, however, rank(\(\mathbf{S}(\nu)\)) = 3 is not full rank. Thus, in the first \(d\) steps, there are unchangeable, or all-pass, dynamics between the output and the input. In terms of causality, corresponding to the last step \(d = 2\), the block-vector

\[
\Lambda = \begin{bmatrix} 1 & 2 & 0.1 & 0.6 \end{bmatrix}^T,
\]

is used to factorize dynamics by QR factorization along the axis of \(\nu\), which can be used to estimate the performance benchmark, see Step 5.

### 6.2 Example 2

This example is given to extend the traditional method to the complex case in the tensor framework. Consider a switched system including two 2 × 2 open-loop transfer function matrices:

\[
\mathbf{T}^1 = \begin{bmatrix}
q^{-1} & 1 & 0 & q^{-1} \\
2q^{-1} & 1 & 0 & 2q^{-1} \\
1 & 1 & 0.3 & 1 \\
1 & 1 & 0.3 & 1
\end{bmatrix},
\]

\[
\mathbf{T}^2 = \begin{bmatrix}
1 & 1 & 0 & q^{-1} \\
1 & 1 & 0 & 2q^{-1} \\
1 & 0 & 1 & q^{-1} \\
1 & 0 & 1 & 2q^{-1}
\end{bmatrix},
\]

The two block matrices of the first 2 Markov parameters gives:

\[
\mathbf{G}^1 = \begin{bmatrix}
\mathbf{G}_1 & 0 \\
\mathbf{G}_1 & \mathbf{G}_0
\end{bmatrix} = \begin{bmatrix}
1 & 1 & 0 & 0 \\
2 & 2 & 0 & 0 \\
0.1 & 0.1 & 1 & 1 \\
0.6 & 0.8 & 2 & 2
\end{bmatrix},
\]

\[
\mathbf{G}^2 = \begin{bmatrix}
\mathbf{G}_0 & 0 \\
\mathbf{G}_2 & \mathbf{G}_0
\end{bmatrix} = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0.3 & 0 & 0 & 0 \\
0.4 & 1 & 0 & 0 \\
0.03 & 1 & 0 & 3
\end{bmatrix},
\]

In this case, \(i_w\) denotes the switching axis, and the switched systems evolve along the \(i_w\). The Example 1 can be considered as the special case of the switched system where \(i_w = 1\). To some extent, the switching sequences can be considered as a class of time series as mentioned in Remark 4. Thus, in addition to the operator \(f(D)\), there is a operator \(f(W)\). They have the same structure as mentioned in Remark 1. The action of operator \(f(D)\) and \(f(W)\) will act on the input, respectively. Hence, \(\mathcal{A} \in \mathbb{R}^{12 \times 12 \times 12 \times 12 \times 12\times 12\times 12}\) is considered, where \(i\nu = 8\), \(i_d = 2\), \(i_{m} = 2\), \(i_w = 2\). The unfolded matrices are shown in Fig. 6.

![Fig. 6. The unfolded matrices of the (8 × 2 × 2 × 2)-tensor \(\mathcal{A}\).](image-url)
\[
S_{(w)} = \begin{bmatrix}
5.0560 & -0.0997 & -0.7059 & 0.9302 & -0.0887 & 0.0137 \\
-1.0188 & 0.4949 & -1.0119 & 0.7937 & -0.9491 & -0.0695 \\
-0.3661 & -3.2158 & 0.0365 & 0.9197 & -0.2707 & 0.0285 \\
\end{bmatrix},
\]

In this case, rank\((S_{(d)}) = 2\), rank\((S_{(m)}) = 2\) and rank\((S_{(w)}) = 2\) are full rank, however, rank\((S_{(w)}) = 6\) is not full rank. Thus, corresponding to the last step \(d = 2\) and \(w = 2\), the block-vector

\[
A = \begin{bmatrix}
1 & 2 & 0.1 & 0.6 & 1 & 0.3 & 0.4 & 0.03 \\
1 & 2 & 0.1 & 0.8 & 0 & 0 & 4 & 1 \\
\end{bmatrix}^T,
\]

is used to factorize dynamics by QR factorization along the axes of \(i_d\) and \(i_w\), respectively. The unitary interactor matrices along \(i_d\) can be obtained as

\[
D_d = \begin{bmatrix}
-0.4472q & -0.8944q \\
-0.8944q^2 & 0.4472q^2 \\
-0.9578q & -0.2873q^2 \\
-0.2873q^2 & -0.9578q^2 \\
\end{bmatrix},
\]

and the unitary interactor matrices along \(i_w\) can also be obtained as

\[
D_w = \begin{bmatrix}
-0.4472q & -0.8944q \\
-0.8944q^2 & 0.4472q^2 \\
-0.1387q & -0.9903q^2 \\
-0.9903q^2 & 0.1387q^2 \\
\end{bmatrix}.
\]

Hence, the underlying causal factors along the axes of \(i_d\) and \(i_w\) should be considered to estimate the performance benchmark, and the traditional interactor-filter is extended with respect to the axes of \(i_d\) and \(i_w\) in this case.

7. CONCLUSION
This paper considers the estimation of the performance benchmark from the tensorial point of view. The corresponding estimation procedure is proposed. The examples illustrate the effectiveness of the proposed method.

REFERENCES


